IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

QOMPLX LLC Plaintiff, v. PALO ALTO NETWORKS, INC. Defendant.	Civil Action No. 2:25-cv-00913-JRG

PALO ALTO NETWORK'S AMENDED INVALIDITY AND SUBJECT MATTER CONTENTIONS

I. INTRODUCTION

Defendant Palo Alto Networks, Inc. ("PANW") provides these Invalidity Contentions to Plaintiff Qomplx LLC ("Qomplx") for the following patents (collectively, "Asserted Patents") and claims (collectively, "Asserted Claims") identified as asserted in Plaintiff's Infringement Contentions served on November 10, 2025 ("Infringement Contentions"):

- Claim 1 of U.S. Patent No. 11,539,663 (the "'663 Patent");
- Claims 1-13 of U.S. Patent No. 12,143,424 (the "'424 Patent");
- Claims 1 and 3-5 of U.S. Patent No. 12,143,425 (the "'425 Patent"); and
- Claims 1, 5-6, 9-22, 24, and 28 of U.S. Patent No. 12,301,627 (the "'627 Patent").[1]

PANW addresses the invalidity of the Asserted Claims and concludes with a description of its document production and identification of additional reservations and explanations.

[1] Qomplx invoked paragraph 3(a)(i) of the Sample Discovery Order for Patent Cases Assigned to Judge Rodney Gilstrap and Judge Roy Payne" ("Sample Discovery Order") for '424 Patent claims 2-6 and 10-12, and '627 Patent claim 24. The entered Discovery Order (Dkt. No. 34) contains a materially identical paragraph 3.

II. RESERVATIONS AND EXPLANATIONS

These Invalidity Contentions and accompanying document productions are subject to further revision and supplementation. Nothing in these contentions constitutes an admission concerning the priority date, conception date, or date of reduction to practice of the Asserted Claims. PANW reserves the right to modify or supplement these Invalidity Contentions, including in response to any positions taken or information disclosed regarding the priority date, conception date, or date of reduction to practice of the Asserted Claims. This disclosure is directed to invalidity issues only and does not address claim construction, non-infringement, patent misuse, inequitable conduct, or other non-statutory defenses. PANW reserves all rights with respect to such issues, including but not limited to their position that the Asserted Claims are to be construed in a particular manner and are not infringed.

These Invalidity Contentions are based on PANW's current knowledge, understanding, and belief as to the facts and information available as of the date of these contentions. PANW has not yet completed its investigation, discovery, or analysis of information related to this action, and additional discovery may require PANW to supplement or amend its Invalidity Contentions. PANW reserves the right to amend or supplement its contentions once it gains access to relevant materials Qomplx has not yet produced. While PANW has made a good-faith effort to provide a comprehensive list of prior art relevant to this case, PANW reserves the right to modify or further supplement its prior art list and contentions on invalidity of the Asserted Claims at a later time with or based upon pertinent information that may be subsequently discovered from Qomplx or third parties. Moreover, discovery is ongoing and PANW reserves the right to pursue all other defenses that may be available to it, including but not limited to defenses that the Asserted Patents are unenforceable based on patent misuse, inequitable conduct, laches, estoppel, waiver,

acquiescence, patent exhaustion, unfair competition, unclean hands, express or implied license, or any other grounds.

Any invalidity analysis depends on claim construction, which is a question of law reserved for the Court. PANW reserves the right to amend, supplement, or materially modify its Invalidity Contentions in response to any claim construction positions that Qomplx may take in this case or any claim construction the Court may adopt in this case. PANW also reserves the right to assert that a claim is indefinite, not enabled, or fails to meet the written description requirement, in response to any additional claim construction positions Qomplx may take in this case or any further claim construction the Court may adopt in this case. These disclosures are based on the apparent claim construction positions taken by the parties and the Court's constructions to date. These disclosures are not, and nothing in these disclosures should be seen as, an endorsement, acquiescence, or acceptance of any of Qomplx's claim constructions.

PANW's Invalidity Contentions are directed to the claims that are asserted and identified in Qomplx's Infringement Contentions. PANW reserves the right to modify, amend, supplement or otherwise alter its Invalidity Contentions in the event that Qomplx supplements or amends its Infringement Contentions, takes a claim construction position that is different from or in addition to those seemingly set forth in its Infringement Contentions, or for any other reason.

PANW further contends that Qomplx appears to be pursuing overly broad constructions of the Asserted Claims in an effort to piece together an infringement claim where none exists and to accuse products that do not practice the claims as properly construed. At the same time, Qomplx's Infringement Contentions are in many respects too general and vague to discern exactly how Qomplx contends the accused instrumentalities practice each element of the Asserted Claims. These Invalidity Contentions are not intended to be, and are not, an admission that the Asserted

Claims are infringed by any of PANW's products or technology, that any particular feature or aspect of the accused instrumentalities practices any element of the Asserted Claims, or that any of Qomplx's proposed constructions are supportable or proper. To the extent that any of the prior art references disclose the same functionality or feature of any of the accused instrumentalities, PANW reserves the right to argue that said feature or functionality does not practice the Asserted Claims, and to argue, in the alternative, that if said feature or functionality is found to practice the Asserted Claims, then the prior art reference demonstrates that that feature or functionality is not novel, is obvious, or is not patentable.

In addition, Qomplx's Infringement Contentions are impermissibly vague and thus deficient on what aspects of the accused instrumentalities allegedly meet numerous limitations of the Asserted Claims. To the extent Qomplx is later permitted to modify or supplement its theories on infringement for those limitations, Palo Alto Networks reserves the right to amend or supplement its invalidity contentions based on any modified or new infringement theories by Qomplx.

Attached hereto as Exhibit Nos. A01-A16 ("A Charts"), B01-B17 ("B Charts"), C01-C17 ("C Charts"), D01-D26 ("D Charts"), and I-L ("Invalidity Charts") are representative claim charts that demonstrate how the Asserted Claims are invalid in view of certain prior art. The Invalidity Charts provide examples of prior art that discloses, either expressly or inherently, or renders obvious every limitation of certain claims and/or teachings, suggestions and motivations through which a person of ordinary skill in the art ("POSITA") at the time of the alleged inventions would have considered the limitations obvious in view of the state of the art at the time, the differences between the claimed invention and the state of the art, and the foreseeability from a technical perspective and/or marketing and/or natural and expected evolution of the art. Where PANW cites

a particular figure in a reference, the citation should be understood to encompass the caption and description of the figure and any text relating to the figure. Conversely, where PANW cites a particular text referring to a figure, the citation should be understood to include the figure as well. The references cited in the attached claim charts may disclose the limitations of the Asserted Claims expressly and/or inherently. Moreover, the suggested obviousness combinations in these cover pleadings and attached charts are in the alternative to PANW's contentions regarding anticipation. These obviousness combinations should not be construed to suggest that any reference included in any combination is not anticipatory in its own right.

For prior art patents and prior art publications identified in these Invalidity Contentions, PANW reserves the right to rely on the public use, offer for sale, sale, and/or actual products embodying the methods and systems described therein uncovered during discovery. PANW also reserves the right to rely on any related patents and patent applications, foreign patent counterparts and foreign patent applications of U.S. patents identified in these Invalidity Contentions, and U.S. counterparts of foreign patents and foreign patent applications identified in these Invalidity Contentions.

PANW further reserves the right to rely on uncited portions of the prior art references and in other publications and testimony as aids in understanding and interpreting the cited portions, as providing context thereto, and as additional evidence that a claim limitation is known or disclosed, whether expressly or inherently. PANW further reserves the right to rely on uncited portions of the prior art references, other publications, and testimony to establish bases for combinations of certain cited references that render the Asserted Claims obvious.

Prior art not included in this disclosure, whether known or not known to PANW, may be relevant to PANW's invalidity arguments in this case. In particular, Qomplx has not provided its

contentions on validity of the Asserted Claims, and PANW is currently unaware of the extent to which Qomplx will contend that limitations of the Asserted Claims are not disclosed in the prior art identified herein. PANW reserves the right to identify, in response to Qomplx's contentions, additional teachings in the same references or in other references that disclose or would have made the addition of the allegedly missing limitation obvious. Moreover, PANW's search for prior art is ongoing, and PANW has and will continue to subpoena third parties believed to have information relevant to this disclosure. PANW expressly reserves the right to amend, supplement, or modify this disclosure as additional information is obtained from third parties, or from Qomplx itself.

PANW reserves the right to modify, amend, or supplement these disclosures as additional information becomes available, and as its discovery and investigation proceeds.

III. PRIORITY DATE OF THE ASSERTED PATENTS AND CLAIMS

In its Infringement Contentions, Qomplx asserts that the patents are entitled to certain priority dates. It is Qomplx's burden to show entitlement to its asserted priority dates, and Qomplx has failed to meet that burden. *See Tech. Licensing Corp. v. Videotek, Inc.*, 545 F.3d 1316, 1327 (Fed. Cir. 2008); *see also, In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1376 (Fed. Cir. 2016) ("[A] patentee bears the burden of establishing that its claimed invention is entitled to an earlier priority date than an asserted prior art reference."). For example, Qomplx has failed to show that any of the Asserted Claims are entitled to a priority date earlier than the filing of the applications that issued as the Asserted Patents. PANW reserves the right to amend its invalidity contentions with additional prior art and/or with additional charts of the art if Qomplx changes its position and alleges earlier or later priority dates or if additional evidence demonstrates that the Asserted Patents are not entitled to their claimed priority dates.

IV. PRIOR ART REFERENCES (P.R. 3-3(A))

PANW identifies the following prior art now known to PANW to anticipate or render obvious the Asserted Claims, either expressly or inherently as understood by a POSITA, either alone or in combination. At this time, PANW contends that the prior art references described herein anticipate or render obvious, either alone or in combination, one or more of the Asserted Claims of the Asserted Patents. These prior art references also provide a description of the level of skill in the art and provide background information showing the state of the art and knowledge of a person of skill in the art. PANW reserves the right to rely on these references for any purpose.

PANW additionally incorporates by reference all prior art references cited on the face of the Asserted Patents, related patents, and foreign counterparts. PANW further incorporates by reference all prior art references identified in the file history of the same.

For each prior art reference cited herein or incorporated by reference, PANW reserves the right to rely upon any foreign counterpart of any U.S. patent or patent application publication and vice versa, any issued U.S. patent for any U.S. patent application publication and vice versa, any other patent family member of any foreign or U.S. patent or patent application publication, any systems, products, or prior inventions described in any patent, patent application publication, article or other publication, and any patent, patent application publication, article or other publication describing any systems, products, or prior inventions. In these invalidity contentions, including the exhibits, any citation to a printed publication or other reference describing a prior art system should also be construed to include a reference to the prior art system itself and vice versa.

A. Prior Art Patents and Publications[2]

Prior Art Reference[3]	Effective Filing Date	Publication Date
U.S. Patent App. Pub. No. 2013/0212214 ("Lawson")	9/13/2012	8/15/2013
U.S. Patent No. 11,416,587 ("Hartog")	3/26/2019	8/16/2022
U.S. Patent App. Pub. No. 2015/0026343 ("Borges")	7/22/2013	1/22/2015
U.S. Patent App. Pub. No. 2014/0108665 ("Arora")	10/7/2013	4/17/2014
U.S. Patent App. Pub. No. 2017/0214660 ("Shah")	1/22/2016	7/27/2017
U.S. Patent 9,930,103 ("Thompson")	4/8/2015	3/27/2018
U.S. Patent No. 9,311,161 ("Jagtap")	9/14/2012	3/20/2014
U.S. Patent No. 9,647,989 ("Woloszyn")	4/27/2011	11/1/2012
U.S. Pat. App. No. 2014/0208316 ("Ciano")	11/6/2013	7/24/2014
China Patent App. No. CN106254425 ("CN425")	7/22/2016	12/21/2016
U.S. Patent App. Pub. No. 2018/0198765 ("Maybee")	1/6/2017	7/12/2018
U.S. Patent 9,536,103 ("Redberg")	12/2/2013	12/17/2015
U.S. Patent No. 9,996,374 ("Sieffert")	6/16/2015	12/22/2016
U.S. Patent No. 9,146,727 ("Arcese")	8/23/2011	8/23/2012
U.S. Patent No. 9,075,690 ("Natchadalingam")	1/4/2011	1/30/2014
U.S. Patent No. 9,225,730 ("Brezinski")	3/19/2014	12/29/2015
U.S. Patent No. 10,810,210 ("Choudhury")	5/12/2017	11/15/2018
U.S. Patent No. 10,462,169 ("Durairaj")	4/29/2017	11/1/2018
U.S. Patent No. 9,503,469 ("Lin")	6/25/2012	12/31/2015
U.S. Patent No. 9,699,205 ("Muddu")	8/31/2015	3/2/2017
U.S. Patent No. 11,601,442 ("Sekar")	8/17/2018	2/20/2020
U.S. Patent No. 9,749,336 ("Zhang")	2/26/2013	8/29/2017
U.S. Patent No. 10,235,521 ("Zhang2")	2/26/2013	11/30/2017
U.S. Patent No. 10,205,735 ("Apostolopoulos")	1/30/2017	8/2/2018
U.S. Patent No. 11,201,890 ("Coull")	3/29/2019	12/14/2021
U.S. Patent No. 9,967,265 ("Peer")	9/29/2015	5/8/2018
U.S. Patent No. 9,596,254 ("Muddu2")	8/31/2015	3/14/2017
U.S. Patent No. 11,575,693 ("Muddu3")	8/31/2015	2/7/2023
U.S. Patent No. 10,944,795 ("Crabtree")	11/27/2017	8/13/2020
U.S. Patent App. Pub. No. 2020/0004905 ("Crabtree2")	2/5/2019	1/2/2020
U.S. Patent App. Pub. No. 2020/0296137 ("Crabtree3")	1/30/2020	9/17/2020
U.S. Patent App. Pub. No. 2018/0219894 ("Crabtree4")	11/29/2017	8/2/2018

[2] The A, B, C, and D exhibits each include exhibits charting system art. Those exhibits list and cite to non-limiting examples of evidence of the charted systems. Those individual pieces of evidence are themselves prior art publications and, to the extent not expressly listed here, are incorporated by reference into the tables in this section.

[3] PANW may rely on the applications from which patents listed herein issued and patents that issued from applications listed herein. The two types of documents have the same disclosures (other than, perhaps, abstract, summary, and claims) and the same effective filing dates. PANW may rely on either document as evidence of the common disclosures.

Prior Art Reference[3]	Effective Filing Date	Publication Date
U.S. Patent App. Pub. No. 2018/0295154 ("Crabtree5")	10/4/2017	10/11/2018
U.S. Patent App. Pub. No. 2018/0159881 ("Crabtree6")	10/23/2017	6/7/2018
U.S. Patent App. Pub. No. 2018/0013771 ("Crabtree7")	9/6/2017	1/11/2018
U.S. Patent App. Pub. No. 2017/0324768 ("Crabtree8")	7/20/2017	11/9/2017
U.S. Patent App. Pub. No. 2019/0222597 ("Crabtree9")	11/14/2018	7/18/2019
U.S. Patent No. 10,956,566 ("Shu")	10/12/2018	6/25/2020
U.S. Patent No. 11,323,463 ("Lee")	6/14/2019	12/17/2020
U.S. Patent App. Pub. No. 2007/0209074 ("Coffman")	3/4/2006	9/6/2007
U.S. Patent No. 9,898,604 ("Fang")	9/26/2013	2/18/2016
U.S. Patent App. Pub. No. 2015/0356301 ("Diehl")	6/6/2014	12/10/2015
U.S. Patent No. 8,898,784 ("Alexander")	5/29/2013	11/25/2014
U.S. Patent App. Pub. No. 2014/0259095 ("Bryant")	3/6/2013	9/11/2014
U.S. Patent No. 9,450,972 ("Mota")	7/23/2014	1/28/2016
WO 2017/127850 ("Hasan")	1/24/2016	7/27/2017
U.S. Patent No. 8,788,556 ("Zhang 556")	5/12/2011	11/15/2012
U.S. Patent Pub. No. 2014/0278337 ("Branson 337")	3/15/2013	9/18/2014
U.S. Patent Pub. No. 2013/0198318 ("Branson 318")	1/30/2012	8/1/2013
U.S. Patent Pub. No. 2013/0179586 ("Branson 586")	1/11/2012	7/11/2013
U.S. Patent No. 9,588,812 ("Branson 812")	7/26/2011	6/30/2016
U.S. Patent Pub. No. 2012/0102029 ("Larson")	10/25/2010	4/26/2012
U.S. Patent No. 7,870,556 ("Wholey")	5/16/2006	11/22/2007
U.S. Patent Pub. No. 2014/0365575 ("Spaven")	6/27/2014	12/11/2014
U.S. Patent No. 8,825,710 ("Gillen 710")	05/24/2012	09/02/2014
U.S. Patent Pub. No. 2012/0303670 ("Gillen 670")	05/24/2012	11/29/2012
U.S. Patent Pub. No. 2016/0188376 ("Rosas 376")	12/26/2014	6/30/2016
U.S. Patent Pub. No. 2015/0379153 ("Mital")	6/30/2014	12/31/2015
U.S. Patent Pub. No. 2014/0108489 ("Glines 489")	10/15/2013	4/17/2014
U.S. Patent No. 8,601,178 ("Amini 178")	6/13/2006	12/03/2013
U.S. Patent Pub. No. 2013/0031282 ("Amini 282")	6/13/2006	1/31/2013
U.S. Patent No. 9,229,952 ("Meacham")	11/5/2014	1/5/2016
U.S. Patent No. 9,699,205 ("Muddu")	8/31/2015	3/2/2017
U.S. Patent Pub. No. 2012/0259983 ("Nakadai")	12/18/2009	10/11/2012
U.S. Patent No. 8,875,145 ("Atterbury")	6/15/2010	12/15/2011

Article or Publication	Publication Date
Middleboxes: Taxonomy and Issues, B. Carpenter ("Middleboxes")	2/2002
SCADA Statistics Monitoring Using The Elastic Stack, J. Hamilton et al. ("SCADA")	2017
Exploring Splunk: Search Processing Language (SPL) Primer and Cookbook, D. Carasso ("Exploring Splunk")	2012

Article or Publication	Publication Date
Best Practices for MID Server Setup and Tuning" available at https://www.servicenow.com/community/itom-blog/best-practices-for-mid-server-setup-and-tuning/ba-p/2274590	11/8/2017
https://www.youtube.com/watch?v=-Y7zULTiyAc&t=76s	11/6/2017
https://www.youtube.com/watch?v=Pgi3WZAqmq0&t=211s	11/6/2017
https://www.atomosnetworks.com/wp-content/uploads/2016/04/citrix-cloudbridge-data-sheet.pdf	2015
https://docs.netscaler.com/en-us/citrix-adc/current-release/system/cloudbridge-connector-introduction.html	9/27/2025
https://docs.netscaler.com/en-us/citrix-adc/current-release/system/cloudbridge-connector-introduction/cloudbridge-connector-aws-introduction.html	9/27/2025
https://medium.com/keylogged/from-splunk-5-x-to-7-0-just-what-have-you-been-missing-out-on-48d37065d60e	1/12/2018
https://www.splunk.com/en_us/download/universal-forwarder.html	2005-2026
https://www.splunk.com/en_us/blog/learn/splunk-universal-forwarder.html	1/3/2023
https://help.splunk.com/en/splunk-cloud-platform/forward-and-process-data/universal-forwarder-manual/9.0/deploy-the-universal-forwarder/universal-forwarder-deployment-prerequisites	2005-2026
https://help.splunk.com/en/splunk-cloud-platform/forward-and-process-data/universal-forwarder-manual/9.0/about-the-universal-forwarder/about-the-universal-forwarder	2005-2026
https://help.splunk.com/en/splunk-cloud-platform/forward-and-process-data/universal-forwarder-manual/9.0/install-the-universal-forwarder/install-a-windows-universal-forwarder	2005-2026
https://help.splunk.com/en/splunk-cloud-platform/forward-and-process-data/universal-forwarder-manual/9.0/configure-the-universal-forwarder/enable-a-receiver-for-splunk-enterprise	2005-2026
Oasis MQTT Version 3.1.1 standard (available at https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html)	10/29/2014
www.hivemq.com/hivemq/ (available at https://web.archive.org/web/20170701184911/http://www.hivemq.com/hivemq/)	7/1/2017
HiveMQ Datasheet (available at https://web.archive.org/web/20170703061421/http://www.hivemq.com/wp-content/uploads/hivemq-product-sheet.pdf)	7/3/2017
MQTT Security Fundamentals: TLS / SSL (available at https://web.archive.org/web/20170422044156/http://www.hivemq.com/blog/mqtt-security-fundamentals-tls-ssl)	4/22/2017
What is Mule ESB? (available at https://web.archive.org/web/20181028231044/https://www.mulesoft.com/resources/esb/what-mule-esb)	10/28/2018
Hybrid Cloud Integration (available at https://web.archive.org/web/20190227192002/https://www.mulesoft.com/resources/esb/hybrid-cloud-integration-solutions)	2/27/2019

Article or Publication	Publication Date
Systems Integration Solutions (available at https://web.archive.org/web/20151010161453/https://www.mulesoft.com/resources/esb/systems-integration-solutions)	10/10/2015
MuleSoft Tutorialspoint (available at https://www.tutorialspoint.com/mulesoft/mulesoft_tutorial.pdf)	2019
Dell Boomi Security Overview Datasheet	6/2011
Dell Boomi SAP Integration Made Easy Whitepaper	2011
Dell Boomi, Connect Once – Integrate Everywhere datasheet	2011
Dell Boomi, Enterprise Integration Made Easy datasheet	2011
Huistra, David, "Automated Generation of Attack Trees by Unfolding Graph Transformation Systems," University of Twente, Faculty of Electrical Engineering, Mathematics & Computer Science ("Huistra")	3/2016
Docker Enterprise Edition (available at https://web.archive.org/web/20180315233420/https://www.docker.com/enterprise-edition)	3/15/2018
See, e.g., Docker – What is a Container? (available at https://web.archive.org/web/20180319033305/https://www.docker.com/what-container)	3/19/2018
SPADE: The System S Declarative Stream Processing Engine, Gedik, Andrade, Wu, Yu, and Doo, SIGMOD'08, Copyright 2008 ACM 978-1-60558-102-6/08/0 ("Spade-Gedik")	6/2008
Storm @Twitter, Toshniwal et al., SIGMOD'14, June 22–27, 2014, Snowbird, Utah, USA. Copyright © 2014 ACM 978-1-4503-2376-5/14/06 ("Toshniwal")	6/2014
Pipeline in BizTalk Server, Turruellas, C-SharpCorner.com	12/2009
The Design of the Borealis Stream Processing Engine, Abadi et al., 2005 CIDR Conference	2005
Storm: Real-Time Data Processing I HadoopABCD, Bikash Sen, (https://hadoopabcd .wordpress.com/2015/04/25/storm-real-ti me-data-processing/)	5/1/2015
Questioning the Lambda Architecture, Jay Kreps, available at https://www.oreilly.com/radar/questioning-the-lambda-architecture/	7/2/2014
How to beat the CAP theorem , Nathan Marz, available at http://nathanmarz.com/blog/how-to-beat-the-cap-theorem.html	10/13/2011
Apache Samza, LinkedIn's Framework for Stream Processing, Navina Ramesh, available at https://thenewstack.io/apache-samza-linkedins-framework-for-stream-processing/	01/07/2015
Big Data: Principles and best practices of scalable real-time data systems, Nathan Marz and James Warren, Manning Publications, Shelter Island, NY, see also https://www.manning.com/books/big-data.	4/2015
Apache Tez: A Unifying Framework for Modeling and Building Data Processing Applications, Saha et al., SIGMOD'15, May 31–June 4, 2015, Melbourne, Victoria, Australia.	06/04/2015

Article or Publication	Publication Date
Amazon Kinesis Product Details, Amazon, available at https://web.archive.org/web/20150905074423/https:/aws.amazon.com/kinesis/details/	Prior to 09/05/2015
AbInitio Blogspot Post, November 27, 2014, available at https://abinitio-introduction.blogspot.com/2014/11/abinitio.html	11/27/2014
"Addressing Data Volume, Velocity, and Variety with IBM InfoSphere Streams V3.0," Mike Ebbers et al., (Mar. 2013), available at: https://www.redbooks.ibm.com/redbooks/pdfs/sg248108.pdf ("Ebbers")	03/2013
"IBM InfoSphere Streams Harnessing Data in Motion," Chuck Ballard et al., (Sept. 2010) available at: https://www.redbooks.ibm.com/redbooks/pdfs/sg247865.pdf	09/2010
"Apache Kafka Documentation," Apache Kafka, available at: https://web.archive.org/web/20151013161047/http://kafka.apache.org/documentation.html	Prior to 10/20/2015
"Apache Kafka 0.8 basic training," Michael Noll (Aug. 18, 2014), available at: https://web.archive.org/web/20150906162333/http://www.slideshare.net/miguno/apache-kafka-08-basic-training-verisign	Prior to 10/20/2015
"Unmeltable Infrastructure at Scale: Using Apache Kafka, Twitter Storm and ElasticSearch on AWS," Jim Nisbet, Loggly (November 2013), available at:https://www.slideshare.net/slideshow/aws-reinvent-28504650/28504650?from_search=0, https://www.youtube.com/watch?v=LpNbjXFPyZ0 ("Loggly")	11/2013
"Kafka: a Distributed Messaging System for Log Processing," Jay Kreps, et al., NetDB'11 (Jun. 12 2011), available at: https://notes.stephenholiday.com/Kafka.pdf ("Kreps")	06/12/2011
"Announcing Suro: Backbone of Netflix's Data Pipeline," Jae Hyeon Bae et al., Netflix (Dec. 9, 2013), available at: https://web.archive.org/web/20150729145607/http://techblog.netflix.com/2013/12/announcing-suro-backbone-of-netflixs.html ("Suro")	12/09/2013
"Airflow: a workflow management platform - Airbnb Engineering," AirBnB (Jun. 2, 2015), available at: https://web.archive.org/web/20160216044343/http://nerds.airbnb.com/airflow/ ("Beauchemin")	06/02/2015
Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center, Hindman et al. In Proceedings of the 8th USENIX conference on Networked systems design and implementation (NSDI'11). USENIX Association, USA, 295–308. ("Hindman")	03/30/2011
Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center, audio/video of Matei Zaharia presentation at (NSDI'11), available at https://www.usenix.org/conference/nsdi11/mesos-platform-fine-grained-resource-sharing-data-center ("Zaharia Audio/Video")	03/2011
Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center, presentation at (NSDI'11), available at http://www.usenix.org/events/nsdi11/tech/slides/hindman.pdf ("Mesos Slides")	03/2011

Article or Publication	Publication Date
Amazon Kinesis Developer Guide, API Version 2013-12-02, Amazon ("Kinesis Developer Guide")	2014
Amazon Kinesis Service API Reference API Reference API Version 2013-12-02, Amazon.	2013
Amazon Kinesis Scaling Utility Github, available at https://github.com/awslabs/amazon-kinesis-scaling-utils/tree/5812a682fd7f37b4b2f43f12023fbba1cd88e5f1	Prior to 10/2015
Amazon Kinesis Client Github, at least as archived on https://web.archive.org/web/20150811095722/https://github.com/awslabs/amazon-kinesis-client	Prior to 08/11/2015
Amazon Kinesis on amazon.com, as archived at https://web.archive.org/web/20150905052709/https://aws.amazon.com/kinesis/?nc2=h_l3_al	Prior to 09/05/2015
Amazon Kinesis Product Details on amazon.com, as archived at https://web.archive.org/web/20150905074423/https://aws.amazon.com/kinesis/details/	Prior to 09/05/2015
Amazon Kinesis Developer Libraries on amazon.com, as archived at https://web.archive.org/web/20150905065309/https://aws.amazon.com/kinesis/developer-resources/	Prior to 09/05/2015
Scryer: Netflix's Predictive Auto Scaling Engine, Jacobson, Yuan, and Joshi. Netflix Tech Blog, available at https://web.archive.org/web/20160202023738/http://techblog.netflix.com/2013/11/scryer-netflixs-predictive-auto-scaling.html. ("Scryer Part 1")	11/05/2013
Scryer: Netflix's Predictive Auto Scaling Engine, Yuan, Joshi, Jacobson, Oberai. Netflix Tech Blog, available at https://web.archive.org/web/20151130040112/http://techblog.netflix.com:80/2013/12/scryer-netflixs-predictive-auto-scaling.html. ("Scryer Part 2")	12/04/2013
MillWheel: Fault-Tolerant Stream Processing at Internet Scale, Akidau et al. The 39th International Conference on Very Large Data Bases, August 26th - 30th 2013, Riva del Garda, Trento, Italy. Proceedings of the VLDB Endowment, Vol. 6, No. 11 ("MillWheel")	2013
Twitter Heron: Stream Processing at Scale, Kulkarni et al., SIGMOD '15: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (May 2015) ("Heron")	May-June 2015
Discretized Streams: Fault-Tolerant Streaming Computation at Scale, Zaharia et al. SOSP'13, Nov. 3–6, 2013, Farmington, Pennsylvania, USA. ("Discretized Streams")	11/2013
"Dynamic Scaling – Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905080435/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-scale-based-on-demand.html	Prior to 09/05/2015
"Auto Scaling API Reference," Amazon Web Services, available at: https://web.archive.org/web/20150905202258/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/AutoScalingGroup.html	Prior to 09/05/2015

Article or Publication	Publication Date
"Auto Scaling Group Lifecycle," Amazon Web Services, available at: https://web.archive.org/web/20150905151232/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/AutoScalingGroupLifecycle.html	Prior to 09/05/2015
"EC2 FAQs," Amazon Web Services, available at: https://web.archive.org/web/20150905064019/https://aws.amazon.com/ec2/faqs/	Prior to 09/05/2015
Auto Scaling Getting Started," Amazon Web Services, available at: https://web.archive.org/web/20150905074709/https://aws.amazon.com/autoscaling/getting-started/	Prior to 09/05/2015
"Auto Scaling Details," Amazon Web Services, available at: https://web.archive.org/web/20150901221931/http://aws.amazon.com/autoscaling/details/	Prior to 09/05/2015
"Auto Scaling Developer's Guide – How Auto Scaling Works," available at: https://web.archive.org/web/20150905194027/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/how-as-works.html	Prior to 09/05/2015
"Auto Scaling Developer's Guide – Configuring Your Auto Scaling Groups," Amazon Web Services, available at: https://web.archive.org/web/20150905201054/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/WorkingWithASG.html	Prior to 09/05/2015
"Auto Scaling Developer's Guide," Amazon Web Services, available at: https://web.archive.org/web/20150905085552/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-dg.pdf	Prior to 09/05/2015
"Auto Scaling API Reference," Amazon Web Services, available at: https://web.archive.org/web/20150905092352/http://docs.aws.amazon.com/AutoScaling/latest/APIReference/as-api.pdf	Prior to 09/05/2015
"Load Balance Your Auto Scaling Group – Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905204232/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/US_SetUpASLBApp.html	Prior to 09/05/2015
"Monitoring Your Auto Scaling Instances – Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905192253/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-instance-monitoring.html	Prior to 09/05/2015
"Planning Your Auto Scaling Group," Amazon Web Services, available at: https://web.archive.org/web/20150905185750/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/planning-your-auto-scaling-group.html	Prior to 09/05/2015
"Scaling Based on Amazon SQS - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905201756/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-using-sqs-queue.html	Prior to 09/05/2015
"Scaling Based on Metrics - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905112437/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/policy_creating.html	Prior to 09/05/2015

Article or Publication	Publication Date
"Scaling Plans - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905192930/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/scaling_typesof.html	Prior to 09/05/2015
"Scaling the Size of Your Auto Scaling Group - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905115316/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/scaling_plan.html	Prior to 09/05/2015
"Tutorial: Set Up a Scaled and Load-Balanced Application - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905095527/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-register-lbs-with-asg.html	Prior to 09/05/2015
"What Is Auto Scaling? - Auto Scaling," Amazon Web Services, available at: https://web.archive.org/web/20150905145054/http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/WhatIsAutoScaling.html	Prior to 09/05/2015
"AWS | Elastic Load Balancing - Cloud Network Load Balancer," Amazon Web Services, available at: https://web.archive.org/web/20150905082332/https://aws.amazon.com/elasticloadbalancing/	Prior to 09/05/2015
"Elastic Load Balancing API Reference," Amazon Web Services, available at: https://web.archive.org/web/20150906003403/http://docs.aws.amazon.com/ElasticLoadBalancing/latest/APIReference/elb-api.pdf	Prior to 09/06/2015
"Elastic Load Balancing Developer Guide," Amazon Web Services, available at: https://web.archive.org/web/20150905080707/http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-dg.pdf	Prior to 09/05/2015
"Real-time Analytics at Massive Scale with Pinot," Praveen Neppalli Naga, LinkedIn Engineering (Sept. 29, 2014), available at: https://engineering.linkedin.com/analytics/real-time-analytics-massive-scale-pinot	2014
"Auto Scaling in the Amazon Cloud," Greg Orzell, Justin Becker, Netflix (Jan. 18, 2012), available at: https://web.archive.org/web/20160202030056/http://techblog.netflix.com/2012/01/auto-scaling-in-amazon-cloud.html	2012
"CLASP: collaborating, autonomous stream processing systems," Michael Branson et al., IFIP 2007, available at: https://dl.acm.org/doi/10.5555/1516124.1516148. ("Clasp").	2007
"Multi-site Cooperative Data Stream Analysis," Fred Douglis et al., available at: https://www.ece.stonybrook.edu/~fanye/papers/osr06-multi.pdf	2006
"Muppet: MapReduce-Style Processing of Fast Data," Wang Lam et al., VLDB Endowment Vol. 5, No. 12 (Aug. 31, 2012), available at: https://arxiv.org/pdf/1208.4175 ("Lam")	2012
Design, Implementation, and Evaluation of Network Monitoring Tasks for the Borealis Stream Processing Engine, Master's Thesis, Morten Lindeberg ("Lindeberg")	2007

PANW additionally identifies and relies on patent references and publication references that describe or are otherwise related to the prior art systems identified below. PANW's investigation into prior art patent and publication references is ongoing, and PANW reserves the right to identify and rely on additional patent or publication references that are identified through further investigation or discovery. PANW reserves the right to supplement as further prior art is identified through investigation or discovery.

B. Prior Art Systems and Products

PANW also contends that the Asserted Claims of the Asserted Patents are invalid based on public knowledge and uses and/or offers for sale or sales of products and services that are prior art. For example, the systems and products in the following table may constitute prior art.

The disclosures of the system and product references identified herein are not limited to the specific documents cited in the A Charts, B Charts, C Charts, D Charts, or Obviousness Charts or to documents cited in the preceding tables in Section A. As understood by a person of ordinary skill in the art, system and product disclosures are often evidenced through materials describing real-world implementations, deployments, demonstrations, and use of the disclosed systems and products. Such materials—including antedating technical presentations, blog posts, articles, documentation, videos, and similar materials—constitute evidence of how the disclosed systems operated and were understood in the art at the relevant time. PANW reserves the right to rely on such materials to illustrate or corroborate the disclosures of system and product references.

Prior Art	Entities	Date[4]
ServiceNow MID Server	ServiceNow	2017
Citrix CloudBridge	Citrix	2015
Splunk Forwarder	Splunk	2013

[4] The date for prior art systems may be the first announced release of that system. Palo Alto Networks reserves the right to rely on any version that qualifies as prior art to the relevant Asserted Claims.

Prior Art	Entities	Date[4]
MQTT Systems	HiveMQ	2017
Docker	Docker	2013
Dell Boomi Atomsphere	Boomi/Dell	2011
MuleSoft	MuleSoft	2015
Pathfinder VM	LightCyber and Palo Alto Networks	April 2019 and earlier
The MulVAL System	Princeton University	2005
The NetSPA System	MIT	2002
The Advanced Cyber Decision Platform System	Fractal Industries	2017
Implementations of Larson, Atterbuy, and/or Wholey patents	Ab Initio	Pre-Oct. 2015
Alibaba implementation of Flink	Alibaba	Pre-Oct. 2015
AWS (Autoscaling, Elastic Load Balancing, Elastic MapReduce)	Amazon	2009; Pre-Oct. 2015
Kinesis (including Scaling Utility)	Amazon	11/2013
Airflow	Apache	06/2015
Flink (including Flink on Yarn)	Apache	May 2011
Hadoop (incl. Yarn and MapReduce)	Apache	2006, 2012 (Yarn)
Heron	Apache / Twitter	Pre-Oct. 2015
Kafka	Apache	2011
Mesos (nee Nexus)	Apache	2011
Samza	Apache	2013
Spark Streaming	Apache	2014
Storm	Apache	2011
Tez	Apache	2014
Borealis	Brandeis/Brown/MIT	2005
Splunk stream processing	Cisco/Splunk	Pre-Oct. 2015
Implementation of stream processing	Crowd Strike	Pre-Oct. 2015
Implementation of stream processing	Fortinet	Pre-Oct. 2015
Dataflow	Google	06/2014
MapReduce	Google	2008
MillWheel	Google	2013
System S / SPADE	IBM	2009
Infosphere (including Streams)	IBM	2009
Implementation of Branson patents	IBM	Pre-Oct. 2015
Implementation of Amini patents	IBM	Pre-Oct. 2015
Implementation of stream processing	LinkedIn	Pre-Oct. 2015
BizTalk Server	Microsoft	2014
Implementation of Zhang patents	Microsoft	Pre-Oct. 2015
Implementation of Mittal Patents	Microsoft	Pre-Oct. 2015
Mantis	Netflix	2014
Scryer	Netflix	11/2013

Prior Art	Entities	Date[4]
Implementation of stream processing	Oracle	Pre-Oct. 2015
Implementation of Meacham patents	Palantir	Pre-Oct. 2015
Splunk security platform Implementation of patents and related	Splunk/Cisco	2015
Implementation of Storm and Mesos	Twitter / X	Pre-Oct. 2015
Muppet	Walmart Labs	2012

PANW intends to conduct third-party discovery from companies and sources that may have relevant prior art or evidence related to relevant prior art. PANW reserves the right to conduct such discovery and supplement its invalidity contentions to include any prior art systems and products that render the Asserted Claims invalid and/or any evidence that otherwise supports PANW's invalidity contentions. PANW may use physical samples, executable software, or source code as evidence of the relevant functionality of these prior art products or services. PANW will make available for inspection any physical samples of products, systems, or software listed above, and/or any source code therefor, that it has in its possession or that becomes available in the future during discovery.

V. INVALIDITY UNDER 35 U.S.C. §102 AND §103 (P.R. 3-3(B) AND (C))

A. Anticipation

PANW contends that the Asserted Claims of the '663 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references charted in Exhibits A01-A16:

Exhibit No.	Prior Art
A01	U.S. Patent App. Pub. No. 2013/0212214 to Lawson ("Lawson")
A02	U.S. Patent No. 11,416,587 to Hartog ("Hartog")
A03	U.S. Patent App. Pub. No. 2015/0026343 to Borges ("Borges")
A04	U.S. Patent App. Pub. No. 2014/0108665 to Arora ("Arora")
A05	U.S. Patent App. Pub. No. 2017/0214660 to Shah ("Shah")
A06	ServiceNow MID Server
A07	Citrix CloudBridge

Exhibit No.	Prior Art
A08	Splunk Forwarder
A09	U.S. Patent 9,930,103 to Thompson ("Thompson")
A10	U.S. Patent No. 9,647,989 to Woloszyn ("Woloszyn")
A11	U.S. Patent App. Pub. No. 2018/0198765 to Maybee ("Maybee")
A12	MQTT Systems
A13	MuleSoft
A14	Boomi Atomsphere
A15	U.S. Patent 9,536,103 to Redberg ("Redberg")
A16	Pathfinder VM

PANW contends that the Asserted Claims of the '424 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references charted in the B Charts:

Exhibit No.	Prior Art
B01	Netflix Mantis ("Mantis")
B02	U.S. Patent Pub. No. 2012/0303670 ("Gillen 670")
B03	U.S. Patent Pub. No. 2016/0188376 ("Rosas 376")
B04	U.S. Patent Pub. No. 2013/0198318 ("Branson 318")
B05	U.S. Patent Pub. No. 2013/0031282 ("Amini 282")
B06	U.S. Patent Pub. No. 2014/0108489 ("Glines 489")
B07	Borealis
B08	Apache Storm ("Storm")
B09	Apache Flink ("Flink")
B10	Apache Samza ("Samza")
B11	Apache Tez ("Tez")
B12	Apache Spark Streaming ("Spark" or "Spark Streaming")
B13	IBM Infosphere ("Infosphere")
B14	Apache Kafka ("Kafka")
B15	Apache Airflow ("Airflow")
B16	Big Data by Marz and Warren ("Big Data")

Exhibit No.	Prior Art
B17	Amazon Kinesis ("Kinesis")

PANW contends that the Asserted Claims of the '425 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references charted in the C Charts:

Exhibit No.	Prior Art
C01	Netflix Mantis ("Mantis")
C02	U.S. Patent Pub. No. 2012/0303670 ("Gillen 670")
C03	U.S. Patent Pub. No. 2016/0188376 ("Rosas 376")
C04	U.S. Patent Pub. No. 2013/0198318 ("Branson 318")
C05	U.S. Patent Pub. No. 2013/0031282 ("Amini 282")
C06	U.S. Patent Pub. No. 2014/0108489 ("Glines 489")
C07	Borealis
C08	Apache Storm ("Storm")
C09	Apache Flink ("Flink")
C10	Apache Samza ("Samza")
C11	Apache Tez ("Tez")
C12	Apache Spark Streaming ("Spark" or "Spark Streaming")
C13	IBM Infosphere ("Infosphere")
C14	Apache Kafka ("Kafka")
C15	Apache Airflow ("Airflow")
C16	Big Data by Marz and Warren ("Big Data")
C17	Amazon Kinesis ("Kinesis")

PANW contends that the Asserted Claims of the '627 Patent are invalid as anticipated under 35 U.S.C. § 102 in view of each of the prior art references charted in Exhibits D01-D26:

Exhibit No.	Prior Art
D01	U.S. Patent No. 9,225,730 ("Brezinski")
D02	U.S. Patent No. 10,810,210 ("Choudhury")
D03	U.S. Patent No. 10,462,169 ("Durairaj")

Exhibit No.	Prior Art
D04	Huistra, David, "Automated Generation of Attack Trees by Unfolding Graph Transformation Systems," University of Twente, Faculty of Electrical Engineering, Mathematics & Computer Science ("Huistra")
D05	U.S. Patent No. 9,503,469 ("Lin")
D06	U.S. Patent No. 9,699,205 ("Muddu")
D07	U.S. Patent No. 11,601,442 ("Sekar")
D08	U.S. Patent No. 9,749,336 ("Zhang")
D09	U.S. Patent No. 10,235,521 ("Zhang2")
D10	U.S. Patent No. 10,205,735 ("Apostolopoulos")
D11	U.S. Patent No. 11,201,890 ("Coull")
D12	U.S. Patent No. 9,967,265 ("Peer")
D13	U.S. Patent No. 9,596,254 ("Muddu2")
D14	U.S. Patent No. 11,575,693 ("Muddu3")
D15	U.S. Patent No. 10,944,795 ("Crabtree")
D16	The MulVAL System ("MulVAL")
D17	The NetSPA System ("NetSPA")
D18	The Advanced Cyber Decision Platform ("ACDP")
D19	U.S. Patent App. Pub. No. 2020/0004905 ("Crabtree2")
D20	U.S. Patent App. Pub. No. 2020/0296137 ("Crabtree3")
D21	U.S. Patent App. Pub. No. 2018/0219894 ("Crabtree4")
D22	U.S. Patent App. Pub. No. 2018/0295154 ("Crabtree5")
D23	U.S. Patent App. Pub. No. 2018/0159881 ("Crabtree6")
D24	U.S. Patent App. Pub. No. 2018/0013771 ("Crabtree7")
D25	U.S. Patent App. Pub. No. 2017/0324768 ("Crabtree8")
D26	U.S. Patent App. Pub. No. 2019/0222597 ("Crabtree9")

These charts identify specific examples of where each limitation of the Asserted Claims is found in the prior art references. The cited portions of prior art references identified in the attached claim charts are exemplary only and representative of the content and teaching of the prior art references, and should be understood in the context of the reference as a whole and as they would

be understood by a POSITA. Citations to a figure, chart, or table should be understood to include the corresponding description and vice versa. PANW may also rely on other publications that include the same disclosure, including other patent family members, foreign counterparts, issued U.S. patents, and published patent applications.

To the extent any item of prior art cited above is deemed not to disclose, explicitly or inherently, any limitation of an Asserted Claims of the Asserted Patents, PANW reserves the right to argue that any difference between that prior art and the corresponding patent claim would have been either inherent to the art, or obvious to a POSITA, even if PANW has not specifically denoted that the art is to be combined with the knowledge of a POSITA.

PANW reserves the right to amend these invalidity contentions to assert these or any subsequently identified references depending on the infringement positions Qomplx may take as the case proceeds and PANW's ongoing investigation.

B. Obviousness

Each anticipatory prior art reference disclosed in the preceding section, either alone or in combination with other prior art, also renders the Asserted Claims of the Asserted Patents invalid as obvious. To the extent any limitation is deemed not to be exactly met by an item of prior art then any purported differences are such that the claimed subject matter as a whole would have been obvious to one skilled in the art at the time of the alleged invention, in view of the state of the art, the prior art references cited herein, and knowledge of those skilled in the art.

In general, a claimed invention is unpatentable if the differences between it and the prior art "are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art." 35 U.S.C. § 103(a); *Graham v. John Deere Co.*, 383 U.S. 1, 13-14 (1966). In particular, each prior art reference may be combined with (1) information known to persons skilled in the art at the time of the alleged invention, (2) any of the

other anticipatory prior art references, and/or (3) any of the additional prior art identified above and in the charts, including in Exhibits I-L. To the extent any of the references in the Invalidity Charts is deemed to lack a particular claim limitation, that reference can be combined with the disclosure identified for that limitation in the Invalidity Chart for another reference, or in Exhibits I-L, to render that limitation and the claim as a whole obvious. PANW may also show that such an allegedly lacking limitation is rendered obvious by the disclosure of any of the other references identified in the Invalidity Charts.

In view of the Supreme Court's decision in *KSR International Co. v. Teleflex Inc.*, 550 U.S. 398, 127 S.Ct. 1727, 1739 (2007) ("KSR"), the USPTO issued a set of Examination Guidelines. *See* Examination Guidelines for Determining Obviousness Under 35 U.S.C. § 103 in View of the Supreme Court Decision in *KSR International Co. v. Teleflex Inc.*, 72 Fed. Reg. 57,526 (Oct. 10, 2007). Those Guidelines summarized the KSR decision, and identified various rationales for finding a claim obvious, including those based on other precedents. Those rationales include:

(A) Combining prior art elements according to known methods to yield predictable results;

(B) Simple substitution of one known element for another to obtain predictable results;

(C) Use of known technique to improve similar devices (methods, or products) in the same way;

(D) Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;

(E) "Obvious to try" – choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;

(F) Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art;

(G) Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

Id. at 57, 529. PANW contends that one or more of these rationales apply in considering the obviousness of the Asserted Claims.

PANW reserves the right to amend or supplement these contentions regarding anticipation or the obviousness of the Asserted Claims in view of further information from Qomplx, or information discovered during discovery. Qomplx has not identified what elements or combinations it alleges were not known to one of ordinary skill in the art at the time. Therefore, for any claim limitation that Qomplx alleges is not disclosed in a particular prior art reference, PANW reserves the right to assert that any such limitation is either inherent in the disclosed reference or obvious to one of ordinary skill in the art at the time in light of the same, or that the limitation is disclosed in another of the references disclosed above and in combination would have rendered the Asserted Claims obvious.

PANW intends to present expert evidence demonstrating a POSITA's motivation to modify or combine prior art references. Such facts and evidence is expert testimony and will be presented in the experts' reports according to the schedule of the Court. PANW incorporates by reference its forthcoming expert reports and any testimony of its experts presented according to the Court's schedule. The motivation to combine or modify the above items of prior art are present, for example, in the references themselves, the Asserted Patents, references cited on the face of the

Asserted Patents, the knowledge, skill, or creativity of one of ordinary skill in the art, the prior art as a whole, and/or the nature of the problems allegedly addressed by the Asserted Patents.

The law requires no showing of a specific motivation to combine or modify prior art references disclosed herein and in the Invalidity Charts, as each combination or modification of art would have no unexpected results and at most would simply represent a known alternative to one of skill in the art. *See KSR Int'l Co. v. Teleflex, Inc.*, 127 S.Ct. 1727, 1739-40 (2007) (rejecting the Federal Circuit's "rigid" application of the teaching, suggestion, or motivation to combine test, instead espousing an "expansive and flexible" approach). Indeed, the Supreme Court held that a POSITA is "a person of ordinary creativity, not an automaton" and "in many cases, a POSITA will be able to fit the teachings of multiple patents together like pieces of a puzzle." *Id*. at 1742. Thus, to a POSITA, the Asserted Claims represent solutions that would have been obvious to try with predictable results.

Nevertheless, should additional evidence of motivation to combine or modify be required to render the Asserted Claims obvious, PANW contends that there was a motivation to modify or combine the references identified herein. The teaching, suggestion, or motivation to combine these references, although not required, is found, explicitly or implicitly in one or more of at least the following: the knowledge or common sense of one of ordinary skill in the art; the prior art references themselves and/or the prior art as a whole, including interrelated teachings of multiple prior art references; the common fields of technology of the references; the subject matter acknowledged as prior art in the Asserted Patents; the nature of the problem to be solved and the existence of similar improvements in similar applications; design incentives and other market forces, including the advantages of creating a superior and more desirable product and the effects of demands known to the design community or present in the marketplace; the ability to implement

the alleged invention as a predictable variation of the prior art; improvements in similar devices; any needs or problems known in the fields addressed by the Asserted Patents; the teachings in the references directed to solving the problems that the Asserted Patents were allegedly directed to solving; and the number of identified, predictable solutions to the problems addressed by these patents. In addition, the simultaneous (and/or prior) inventions described above (and elsewhere in these contentions) are evidence that motivation to combine the concepts described in the various prior art references did, in fact, exist, and they were, in fact, combined.

In addition, it would have been obvious to try combining or modifying the prior art references identified above because there were only a finite number of predictable solutions and/or because known work in one field of endeavor prompted variations based on predictable design incentives and/or market forces either in the same field or a different one. In addition, the combination or modification of the prior art references would have been obvious because the combination or modification represents known potential options with a reasonable expectation of success. Specific examples of such motivations to combine or modify are outlined below, and are not meant to be exhaustive.

Additional evidence that there would have been a motivation to combine or modify the prior art references above includes the interrelated teachings of multiple prior art references; the effects of demands known to the design community or present in the marketplace; the existence of a known problem for which there was an obvious solution encompassed by the Asserted Claims; the existence of a known need or problem in the field of the endeavor at the time of the alleged invention(s); and the background knowledge, skill, or creativity that would have been possessed by a person having ordinary skill in the art. PANW may rely on uncited portions of the prior art references cited and produced, other publications and testimony, and the testimony of experts to

establish that a POSITA would have been motivated to modify or combine certain of the cited references so as to render the claims obvious. For example, where multiple publications describe the same system, multiple devices are intended to be used together, or are from the same author or publisher, one of ordinary skill would be motivated and find it obvious to read those publications together and combine their disclosure. As another example, where a product was available in another country, the availability of such a product would motivate a POSITA to arrive at the same or similar product in the United States. As yet another example, where the prior art identifies the desirability of or plans for future enhancements of a system, a POSITA would be motivated to incorporate such enhancements even if the original developer or author was unable to complete those enhancements for business reasons.

Exemplary combinations are listed in the below sections for the Asserted Patents for purposes of explaining and exemplifying possible combinations under the motivation discussed in a particular paragraph. These combinations are not intended to be limiting but rather intended to provide notice of the reasoning behind particular motivations to combine. In many instances, multiple different motivations would apply to a particular combination (for example, if multiple references were both authored by the same person and referred to the same product/system, then both of those motivations would apply) and the specific combinations were not duplicated.

Should any prior art be deemed not to disclose, explicitly or inherently, any limitation of a claim, PANW reserves the right to argue that any such difference between that prior art and the corresponding claim would have been obvious to one of ordinary skill in the art. To the extent that such an argument is deemed a "combination" analysis for purposes of obviousness, PANW discloses its present intention to rely on the separate combination of the knowledge of a person having ordinary skill in the art with each item of prior art identified. The knowledge of a POSITA

is demonstrated by, for example, the full list of references above, the prior art produced in this litigation, and forthcoming expert reports.

Because discovery is ongoing and PANW has not yet completed its investigation, discovery, or analysis of the issues raised by Plaintiff's claims, PANW reserves the right to supplement and amend its explanation of why the prior art renders the asserted claims obvious, including an identification of any combinations of prior art showing obviousness, as it receives additional information either through its own investigations or from Plaintiff or third parties. In particular, PANW's investigation and analysis is significantly impeded by the insufficiency and incompleteness of Plaintiff's infringement contentions.

1. '663 Patent Obviousness

One of the Following Primary References	In Combination With The Following Secondary References
Lawson	Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Hartog	Lawson, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Borges	Lawson, Hartog, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Arora	Lawson, Hartog, Borges, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Shah	Lawson, Hartog, Borges, Arora, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg,

	Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
ServiceNow MID Server	Lawson, Hartog, Borges, Arora, Shah, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Citrix CloudBridge	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Splunk Forwarder	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Thompson	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Woloszyn	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Maybee	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
MQTT Systems	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MuleSoft, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
MuleSoft	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, Boomi Atomsphere, Redberg, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Boomi Atomsphere	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Redberg, Pathfinder VM,

	Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Redberg	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Pathfinder VM, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam
Pathfinder VM	Lawson, Hartog, Borges, Arora, Shah, ServiceNow MID Server, Citrix CloudBridge, Splunk Forwarder, Thompson, Woloszyn, Maybee, MQTT Systems, MuleSoft, Boomi Atomsphere, Redberg, Jagtap, Ciano, CN425, Docker, Middleboxes, SCADA, Exploring Splunk, Sieffert, Arcese, Natchadalingam

PANW may assert obviousness based on the primary reference alone, or a combination of a primary reference and (1) a single one of the secondary references identified in the chart associated with that primary reference; (2) all secondary references identified in the chart associated with that primary reference; or (3) any combination of a subset of secondary references identified in the chart associated with that primary reference. In addition to the reasons identified above, one of ordinary skill in the art would be motivated to combine any of the references identified above because they are in the same field of network computer systems that provide for transfer and transformation of data between a local area network and a remote, e.g., cloud-based, network. Because the references are in the same field, address the same technology, and are intended to solve the same general problems, one of ordinary skill in the art would have found it obvious to combine the references, rendering the asserted patent obvious. *See, e.g.*, Lawson at Abstract ("A cloud gateway for coupling an industrial system to a cloud platform is provided. The cloud gateway collects data from one or more industrial controllers, meters, sensors, or other devices comprising an industrial automation system. The cloud gateway optionally performs additional transformations on the data to add context, Summarize, filter, reformat, and/or encrypt the data. The cloud gateway then sends data to a cloud platform for use by one or more cloud-

based applications or services."); Hartog at 3:46-58 ("FIG. 1 shows an illustrative example 100 of a data security server 112 associated with a site 102 in communication with a computing resource service provider 118 via a communication network, such the internet 122."); Borges at Abstract ("A computer system and associated method and software where communications between the cloud and a sensor device (for example, a temperature sensor) pass through and are processed by a middleware appliance."); Arora at Abstract ("The present disclosure is directed to multi-level application programming interface (API) integration of a distributed enterprise application architecture deployed across data center/private cloud and public cloud, such as end-to-end configuration of bridging two virtual or physical datacenters via a cloud bridge using simple/user-friendly configuration interfaces and API calls."); Shah at [0003] ("In some aspects, a system for optimizing network traffic is described. The system includes a plurality of appliances. An appliance can comprise one or more network interfaces and a secure session connection optimizer module. The one or more network interfaces are configured to facilitate secure communications between a client device and a server, wherein the secure communications involve a plurality of secure session connections comprising a first secure session connection between client device and appliance and a second secure session connection between appliance and another appliance."); ServiceNow MID Server at Best Practices Page ("A ServiceNow MID Server enables ServiceNow applications like Discovery and Event Management to communicate with external systems, or targets, on the enterprise's local network or in the public cloud. The MID Server software runs on a host computer on the enterprise's network or in the cloud."); Citrix CloudBridge at Understanding CloudBridge Connector ("The CloudBridge Connector feature of the NetScaler appliance connects enterprise datacenters to external clouds and hosting environments, making the cloud a secure extension of your enterprise network."); Splunk Forwarder at Splunk Universal

Forwarder Article ("A universal forwarder collects data from a variety of places — whether data sources or other forwarders — and then sends it to a forwarder or a Splunk deployment."); Thompson at 2:25-38 ("Specifically, systems and methods are disclosed which facilitate the handling of user requests to perform certain tasks on remote or third party systems."); Woloszyn at Abstract ("An intercepting proxy server processes traffic between an enterprise user and a cloud application which provides Software as a Service (SaaS). The intercepting proxy server provides interception of real data elements in communications from the enterprise to the cloud and replacing them with obfuscating information by encrypting individual real data elements without disturbing the validity of the application protocol."); Maybee at [0034] ("Certain embodiments may provide the ability to migrate data to - and - from the cloud and may provide for local data to co - exist with data in the cloud by way of a hybrid cloud storage system, which provides for storage elasticity and scale while layering ZFS file system functionality on the cloud storage."); MQTT Systems at HiveMQ Page ("HiveMQ empowers your organization to connect all your devices and services with minimal effort by using the de-facto IoT standard protocol MQTT. HiveMQ's natural habitat is in the frontier between devices and enterprise systems. It makes sure all your devices and services get the latest information via instant, bi-directional push messages and can send their current data to your backend systems in real time."); MuleSoft at Hybrid Cloud Integration ("Hybrid integration bridges the network divide between your existing enterprise (databases, warehouses, applications, and legacy systems) and SaaS/PaaS, B2B (partners & suppliers), B2C (customer engagement), BYOD (enterprise mobile), and big data. Hybrid integration solves the issue of taking existing information from on-premises systems."); Boomi Atomsphere at Boomi SAP Integration, p. 1 ("AtomSphere supports any combination of SaaS, Cloud, and On-Premise applications with integration processes that can be deployed securely behind the firewall or just as

securely in the Cloud. Through AtomSphere, large enterprises can now tap into the industry's largest open network of SaaS, Cloud and On-Premise applications with unprecedented ease."); Redberg at 2:18-26 ("Methods and systems are described for vendor independent and secure cloud storage distribution and aggregation. According to one embodiment, a generalized application programming interface (API) is provided by a cloud storage gateway device that is logically interposed between one or more third-party cloud storage platforms and users of an enterprise. The API facilitates storing of files, issuing of search requests against the files and retrieval of content of the files."); Jagtap at Abstract ("A system, apparatus, method, and computer program product for automatically configuring management service payloads for cloud IT service delivery are disclosed."); Ciano at Abstract ("A Solution is proposed for deploying a virtual appliance onto a data processing system."); CN425 at Abstract ("The invention discloses a method for a mobile device to transmit data to cloud, a system for a mobile device to transmit data to cloud, and a mobile terminal."); Docker at Docker for the Enterprise ("Docker Enterprise Edition (EE) is a Containers-as-a-Service platform for IT that manages and secures diverse applications across disparate infrastructure, both on-premises and in the cloud."); Middleboxes at 1.1 ("A middlebox is defined as any intermediary device performing functions other than the normal, standard functions of an IP router on the datagram path between a source host and destination host."); SCADA at p. 451 ("This paper describes the service that we have implemented in order to more easily obtain statistics and error logs from all applications through a centralised web application."); Exploring Splunk at p. 9 ("By indexing and aggregating log files from many sources to make them centrally searchable, Splunk has become popular among system administrators and other people who run technical operations for businesses around the world.").

The ’663 Patent purports to address complications associated with managing data transmissions between enterprises and cloud-based services. *See, e.g.*, ‘663 Patent at 7:13-16 (“As cloud-based computing services become more popular, management of the data collection from a business and transmission of that data to a cloud-based service become more complicated.”); 7:22-31 (“When a large business enterprise uses a cloud-based computing service, heterogeneous data transfer between cloud services and large offices or campuses for organizations presents numerous problems, including lack of reliable data collection methods, poor standardized support for connection-oriented protocols by network appliances, security concerns with unfiltered or poorly filtered data, and bandwidth concerns with constantly streaming data which may result in network slowdown due to unprioritized data trans-fer.”). The ’663 Patent purports to solve this problem with a “a system and method that uses midservers integrated with the business enterprise computer infrastructure and the cloud-based infrastructure 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.” *Id.* at 7:61-66. A POSITA would be motivated to combine the identified prior art references because the references address the same or similar issues in the same or similar ways. *See* Exs. A1-A16. Further, the prior art references teach complementary technical features, for example, an architecture with midserver that can receive data from multiple devices within an enterprise, aggregate and transform the data, and retransmit the data to an external system such as a cloud-based application. *See id.* Persons of ordinary skill in the art would have recognized that there were no significant impediments (technological or otherwise) to combining elements and features from the identified references. Indeed, the prior art references evidence a mixing-and-matching of the same basic features into different combinations.

Developments in the art would also provide motivation to combine prior art references in ways that would take advantage of those developments. For example, the emergence of cloud computing and software-as-a-service platforms in the late 1990's and early 2000's created a need for development of methods for efficiently managing enterprise data and interfacing with external networks and systems. The need for solutions to the problems associated with integrating enterprise architectures with cloud-based service providers was well-recognized. For example, Arora explained:

> The explosion of cloud computing domain has accelerated the growth in the 3 major cloud service models: IaaS, PaaS, SaaS. With Infrastructure as a Service (IaaS). there has been a large amount of development efforts but with less standardization as compared to PaaS and SaaS solutions. As a result, IaaS service providers provide various different architectures and technologies. This makes integration of a distributed enterprise application architecture with such IaaS service providers complex and challenging.

Arora at [0004]. Similarly, Hartog noted that the use of remote computing also posed security risks:

> As the use of remote computing increases, it is becoming more and more challenging to effectively manage information security. Systems are becoming more and more complex, often with multiple different subsystems (e.g., services) managed by different entities (e.g., different administrators in different business units). Clients may wish to access various resources through a variety of different networks, both trusted and untrusted. While data security solutions are adapting to different computing environments, visibility into network traffic, particularly for untrusted or remote networks, still poses many challenges for current data security solutions. These data security solutions are rapidly changing, making it increasingly more difficult to build solutions that can adapt to this changing environment.

Hartog at 1:7-20.

In light of the teachings of the prior art, a POSITA would have readily appreciated the benefits of using an intermediary device (i.e., a midserver) to facilitate the transfer and transformation of data between enterprise systems and cloud-based applications. As shown in the prior art references herein, many companies and individuals utilized this approach and recognized

its benefits. *See* Exhibits A1-A16. As taught in numerous prior art references, for example, use of a midserver facilitates integration of enterprise computing systems with cloud-based systems because the midserver can transform data coming from the enterprise to match the format required by the cloud-based system. Rather than requiring each separate computing device in the enterprise to communicate data in the required format of multiple different cloud-based systems, the midserver can manage this data formatting. A POSITA would have been motivated to provide this capability in order to make it easier to add and remove devices and applications from the system.

A POSITA further would have been motivated to use a midserver to reduce the processing load on enterprise devices. For example, Arora teaches that a midserver "relieves servers 106 of much of the processing load caused by repeatedly opening and closing transport layers connections to clients 102 by opening one or more transport layer connections with each server 106 and maintaining these connections to allow repeated data accesses by clients via the Internet." Arora at [0154].

A POSITA further would have been motivated to use a midserver to improve security of the communications between an enterprise and cloud-based system. Hartog, for example, discloses that the use of a midserver "may provide enhanced data security for a variety of networks, including untrusted networks. The described system and techniques may also provide for a faster roll out of information security systems while maintaining the security of data, thus decreasing downtime, reducing security risk, particularly in untrusted or remote networks." Hartog at 3:39-45.

These benefits and others disclosed in the cited prior art references would have motivated a POSITA to look to these references and combine their teachings. A POSITA would have been

motivated to combine the cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits A1-A16.

A POSITA would also be motivated to implement a midserver as a virtual appliance. The use of virtual appliances, including midservers operating as a virtual appliance, and their related benefits were well-known by the time of the claimed priority date of the '663 patent. Those benefits include, for example, easier deployment, pre-configuration, portability, hardware independence and ease of maintenance and management. Hartog, for example, discloses use of a midserver that "may be created on standalone hardware, or integrated with existing network hardware, through the use of a virtualized environment, such as a software container engine or service, such as Docker. The data security server may be loaded with a software container engine, such that it can support multiple different software containers that may share a host operating system, libraries, etc., but allow isolated execution of one or more applications in the separate software containers." 1:57-2:22; *see also* Arora at [0147] ("The appliance 200 comprises one or more virtual servers or virtual internet protocol servers, referred to as a vServer, VIP server, or just VIP 275a-275n (also referred herein as vServer 275). The vServer 275 receives, intercepts or other wise processes communications between a client 102 and a server 106 in accordance with the configuration and operations of the appliance 200."), [0200] ("Any of the modules of the appliance 200 as described in connection with FIG. 2A may be packaged, combined, designed or constructed in a form of the virtualized appliance delivery controller 450 deployable as one or more software modules or components executable in a virtualized environment 300 or non-virtualized environment on any server, such as an off the shelf server. For example, the virtual appliance may

be provided in the form of an installation package to install on a computing device."); Shah at [0032] ("In some embodiments, appliance 108 can be a virtual appliance. In some embodiments, appliance can be a physical device having multiple instances of virtual machines (e.g., virtual Branch Repeater)."); Thompson at 14:62-15:4 ("In other embodiments, the endpoint proxy system 132 is implemented by one or more virtual machines instances in a hosted computing environment."), 1:47-54 ("With virtualization, the single physical computing device can create, maintain, delete, or otherwise manage virtual machines in a dynamic manner."), 15:5-12 ("The host computing device 510 may host a virtual machine instance 114 by executing a software virtual machine manager 122, such as a hypervisor, that manages the virtual machine instance 114. The virtual machine instance 114 may execute an instance of an operating system and application software."); HiveMQ Datasheet p. 4 ("HiveMQ runs without any complex installation routines and works on Linux, Windows and OS X. Deploy HiveMQ with ease on server hardware, Virtual Machines or Containers (like Docker) in your private, hybrid or public cloud. We don't force you to use a (virtual) appliance, you can install HiveMQ virtually everywhere: In your own computing center, on your local machine or on IaaS providers like AWS, Azure, or Google Cloud Platform."). The references cited herein demonstrate that the benefits of using virtual applications, including the fact that they are easier to deploy, configure and update, were well-known and those benefits would have motivated a person of ordinary skill in the art.

A POSITA would also be motivated to utilize existing methods for automatically installing a virtual appliance software application that is configured to automatically load stored configurations on a midserver. Not only was the use of virtual appliances well-known as of the claimed priority date of the '663 Patent, but numerous techniques were also well-known for automatically installing a virtual appliance software application that is configured to automatically

load stored configurations, which is beneficial in that it provides improved security and allows system managers to easily ensure that all users have access to the most up-to-date applications and proper configurations. Lawson, for example, recognized the value in being able to automatically install and configure an intermediary gateway: "Industrial devices 108 and 110 and/or cloud gateways 106 having smart configuration capability can be configured to automatically detect and communicate with the cloud platform 102 upon installation at any facility, simplifying integration with existing cloud-based data storage, analysis, or reporting applications used by the enterprise." [0040] Likewise, the ServiceNow MID Server included processes to automatically configure the MID Server, including identifying subnets that the MID Serve can reach in order to interact with endpoints in those subnets. *See* Ex. A6; *see also* Arora at [0346] ("The appliance and/or virtual appliance may execute a script based on the cloud bridge layer configuration to instruct and configure an appliance or virtual appliance to establish a cloud bridge."). Another prior art reference, Arcese, disclosed a method for "a software application distributed as a virtual appliance in a virtual machine system [to be] automatically upgraded, comprising detecting, by an application programming interface of a virtual machine monitor, a change in a previously stored version of a list of virtual appliance virtual disks attached to a corresponding virtual machine." Arcese at 2:45-51. Similarly, Natchadalingam discloses methods to "provide secure updates of software and/or an operating system running on a virtual machine, where a hypervisor-based proxy server receives the updates via a network and pushes the updates to the virtual machine." Natchadalingam at 2:36-40. A POSITA would have been motivated to use known techniques for automatically installing a virtual appliance software application that is configured to automatically load stored configurations because it eases deployment and configuration, improves security, and ensures that software is up-to-date. A POSITA would have been motivated to combine these

teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits A1-A16.

A POSITA also would have been motivated to utilize secure network connections between an enterprise and an external network in order to ensure the protection of enterprise data. The need to secure, and techniques for securing, network connections within an enterprise and between an enterprise and an external network were well-known as of the claimed priority date of the '663 Patent. For example, Lawson taught that "The proxy industrial device can also optionally include an encryption component 1018 (similar to encryption components 210, 314, and 818 of FIGS. 2, 3, and 8, respectively) for encrypting the industrial data 1014 prior to submission to the cloud platform." Lawson at [0075]. Borges taught the use of a "hardware cryptography sub-mod 308 which encrypts and decrypts the substantive data of the cBDOs" and a "HTTPS (hypertext transfer protocol) proxy/VPN (virtual private network) sub-mod 312 which manages proxies for and virtual private network communications of cDOs for secure and reliable network communications between appliance server 104 and various cloud computing resources." Borges at [0032]. Arora taught that "the appliance 200 provides a secure virtual private network connection from a first network 104 of the client 102 to the second network 104" of the server 106, such as an SSL VPN connection." [0095]. Shah taught "Appliance 108 can comprise one or more network interfaces configured to facilitate secure communications between client device 102 and a server (e.g., public server 150 or enterprise server 122), wherein the secure communications involve a plurality of secure session connections comprising a first secure session connection between client device 102

and the appliance 108 and a second secure session connection between the appliance 108 and another appliance 108." Shah at [0097]. Furthermore, the use of SSL and TLS were well-known cryptographic protocols that provided secure communications over a network, such as the internet, and were used by multiple systems, including implementations of a midserver. *See, e.g.*, MID Server Best Practices Page, Youtube Video (https://www.youtube.com/watch?v=-Y7zULTiyAc&t=76s) at 2:39; HiveMQ Page ("Of course all state-of-the-art security standards like SSL/TLS are supported to keep your data safe while being transferred to and from HiveMQ."); Thompson at 4:52-62 ("The network 104A and/or 104B can use protocols and components for communicating via the Internet or any of the other aforementioned types of networks. For example, the protocols used by the network 104A and/or 104B may include Hypertext Transfer Protocol (HTTP), HTTP Secure (HTTPS), Message Queue Telemetry Transport (MQTT), Constrained Application Protocol (CoAP), and the like. Protocols and components for communicating via the Internet or any of the other aforementioned types of communication networks are well known to those skilled in the art and, thus, are not described in more detail herein."); Shah at [0002] ("A middlebox can establish secure session connections with a client device, a server, or another middle box, for example, using secure session layer (SSL) or a transport layer security (TLS) handshake protocol."); Boomi AtomSphere, Security Datasheet, p. 1 ("All communication from Atom to data center uses SSL 128 bit encryption. All outbound communication from Atom to data center is HTTPS. Atoms use a standard SSL Handshake to authenticate with platform.boomi.com."). A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such

features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits A1-A16.

A POSITA also would have been motivated to utilize a midserver to receive data over a local network from a plurality of computing devices, apply transformation to at least a portion of the data, and retransmit the data over a secure connection as a single data stream. One of the well-known benefits of using a midserver is that it simplifies complexities of integrating distributed enterprise applications with cloud based services by aggregating data from multiple devices, transforming it through encryption, protocol formatting, etc., and retransmitting it to the appropriate remote network location. For example, Arora '665 discloses that a midserver can include a "multi-protocol compression engine 238 [that] compresses any payload of or any communication using a modified transport control protocol," thereby "accelerat[ing] performance for users accessing applications via desktop clients." Arora at [0140]-[0141]; *see also* Arora at [0095] ("[T]he appliance 200 accelerates transport layer traffic between a client 102 and a server 106. The appliance 200 may provide acceleration techniques for accelerating any transport layer payload from a server 106 to a client 102. Such as: 1) transport layer connection pooling, 2) transport layer connection multiplexing, 3) transport control protocol buffering, 4) compression and 5) caching."). Lawson teaches that "The cloud gateway collects data from one or more industrial controllers, meters, sensors, or other devices comprising an industrial automation system. The cloud gateway optionally performs additional transformations on the data to add context, summarize, filter, reformat, and/or encrypt the data. The cloud gateway then sends data to a cloud platform for use by one or more cloud-based applications or services." Lawson at Abstract. Shah discloses that a "middlebox is a network appliance that manipulates Internet traffic by optimizing data flow across the network" by, for example, "establish[ing] secure session

connections with a client device, a server, or another middlebox" and "modify[ing] communications received from client device, server, or another appliance and send[ing] the modified communication to the next destination." Shah at [0001]-[0002]. Hartog discloses "a connection-oriented protocol is used to communicate between network endpoints such that the connection-oriented protocol (sometimes called a connection-based protocol) is capable of transmitting data in an ordered stream." 18:56-60." Hartog at 18:56-60. Borges discloses that "In this embodiment, the form, formal, protocol, communication layer, encapsulation, packetization, etc. of communication between appliance server 104 and cloud-based computer set 110 is largely, or entirely, by a data construct called a cEDO, which cEDOs are communicated over network 114." Borges at [0026]. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits A1-A16.

2. '424 and '425 Patents Obviousness

One of the Following Primary References	**In Combination With The Following Secondary References**
Mantis	Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury).
Gillen 670	Mantis, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS

	Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Rosas 376	Mantis, Gillen 670, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Branson	Mantis, Gillen 670, Rosas 376, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Amini 282	Mantis, Gillen 670, Rosas 376, Branson, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Glines 489	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)

Borealis	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Storm	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Flink	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Samza	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Tez	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google

	Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Spark Stream	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, InfoSphere, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
IBM InfoSphere	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, Kafka, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Kafka	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Airflow, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Apache Airflow	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Big Data, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Big Data	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere,

	Kafka, Airflow, Kinesis, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)
Kinesis	Mantis, Gillen 670, Rosas 376, Branson, Amini 282, Glines 489, Borealis, Storm, Flink, Samza, Tez, Spark Streaming, InfoSphere, Kafka, Airflow, Big Data, Apache Hadoop, Apache Mesos (and, e.g., Hindman, Zaharia Audio/Video, Mesos Slides), Amazon AWS Auto-Scaling, Amazon AWS Elastic Load Balancing, Amazon AWS Elastic MapReduce, Google Dataflow, MapReduce,Google Millwheel (including, e.g., MillWheel by Akidau), IBM System S (including, e.g., Spade by Gedik), Branson 337, Branson 586, Branson 812, Netflix Scryer (including, e.g., Scryer Part 1, Scryer Part 2), AbInitio (including, e.g., Larson, Wholey, and Atterbury)

PANW may assert obviousness based on (1) each primary reference alone, or a combination of a primary reference and (2) a single one of the secondary references identified in the above chart as associated with that primary reference; (3) all secondary references identified in the above chart as associated with that primary reference; (4) any combination of a subset of secondary references identified in the chart associated with that primary reference; (5) any of the above in further combination with any of the references identified below in this section or in Exhibits J (for the ’424 patent) and K (for the ’425 patent).

In addition to the reasons identified above and below, one of ordinary skill in the art would be motivated to combine as set forth in preceding paragraph for the reasons set forth in the respective B Charts and C Charts and in Exhibit J and Exhibit K..

One of ordinary skill in the art would also be motivated to combine any of the references identified above because they are in the same related fields of data processing or process management, especially as relates to near real-time distributed processing. Because the references are in the same field, address the same technology, and are intended to solve the same general

problems, one of ordinary skill in the art would have found it obvious to combine the references, rendering the asserted patent obvious. *See, e.g.*, Mantis: Netflix's Event Stream Processing System at pp. 9-17, 36-39, 81 (system needed to process millions of updates per second and hundreds of billions per day with a distributed architecture) and Stream-processing with Mantis at p. 2("Mantis is a platform for building low-latency, high throughput stream-processing apps"); Gillen 670 at [0003] ("This disclosure relates to load balancing in a computing system and more particularly to the distribution of processing tasks and memory pressure among a plurality of hardware and software systems in a computing environment"); Rosas 376 at [0001] and [0002] ("The present invention is in the field of systems, methods and computer program products for distributed stream processing applications that have to deal with changing traffic conditions and real time processing." "An embodiment of the invention is an elastic stream processing systems that can cope with changing traffic conditions alleviating load balancing issues due to unbalanced data distribution. The method is capable of adapting itself to different traffic conditions to avoid bottlenecks in the stream processing process."); Branson 318 at [0002] ("Embodiments of the present invention generally relate to stream computing applications. Specifically, the disclosed embodiments disclose distributing processing elements or operators to take advantage of underutilized hard ware resources in a stream computing environment."); Amini 282 at[0003] ("The present invention generally relates to the field of distributed stream processing systems, and more particularly relates to scheduling techniques for a distributed stream processing system."); Glines 489 at [0006] ("Various aspects of the present application may relate to techniques by which to address the above-mentioned limitations of existing large data analysis techniques, and/or to generally provide a high-performance computing environment. The techniques described herein, which may involve the use of something called a 'flowlet' which will be dis cussed in further detail

below, may be implemented using hardware, software, firmware and/or combinations thereof.”); Abadi-2005 at 2 (“Borealis is a distributed stream processing engine.”); Storm Homepage (“Apache Storm is a free and open source distributed realtime computation system. Storm makes it easy to reliably process unbounded streams of data, doing for realtime processing what Hadoop did for batch processing. … Storm has many use cases: realtime analytics, online machine learning, continuous computation, distributed RPC, ETL, and more. Storm is fast: a benchmark clocked it at over a million tuples processed per second per node. It is scalable, fault-tolerant, guarantees your data will be processed, and is easy to set up and operate…. A Storm topology consumes streams of data and processes those streams in arbitrarily complex ways, repartitioning the streams between each stage of the computation however needed.”); ApacheCon-Flink at 3 (Flink is a “Distributed Data Flow Processing System” “focused on large-scale data analytics,” “real-time stream and batch processing,” and with a “robust execution backend.”); Ramesh (“At LinkedIn, we created Apache Samza to solve various kinds of stream processing requirements in the company. The framework, originally open sourced by LinkedIn, helps you build applications to process feeds of messages…. Samza's goal is to provide a lightweight framework for continuous data processing.. Samza might help you to update databases, compute counts or other aggregations, transform messages, or any number of other operations.”); Tez Framework at 1357 (Tez is “an open-source framework designed to build data-flow driven processing runtimes” and “provides a scaffolding and library components that can be used to quickly build scalable and efficient data-flow centric engines.”); Spark Streaming Programming Guide (“Spark Streaming … enables scalable, high-throughput, fault-tolerant stream processing of live data streams. Data can be ingested from many sources like Kafka, Flume, Twitter, ZeroMQ, Kinesis, or TCP sockets, and can be processed using complex algorithms expressed with high-level functions like map, reduce,

join and window… processed data can be pushed out to filesystems, databases, and live dashboards."); Ebbers at 53 ("InfoSphere Streams provides a development platform and runtime environment that can ingest, filter, analyze, and correlate massive volumes of continuous data streams… InfoSphere Streams fuses these heterogeneous data types onto a powerful computing platform that enables complex data analysis… Stream processing requires resilient, memory-rich servers to ensure continuous delivery of insight…"); Kafka Documentation ("Kafka is a distributed, partitioned, replicated commit log service."); Beauchemin (Airflow addresses (1) "network[s] of computation jobs," which "are typically DAGs (directed acyclic graphs)" that need to be scheduled, are "mission critical," "evolving," and "heterogenous," and (2) workflow management for these computation jobs); *Big Data* at xvii ("*Big Data* teaches you to build these systems using an architecture that takes advantage of clustered hardware along with new tools designed specifically to capture and analyze web-scale data. It describes a scalable, easy-to-understand approach to Big Data systems that can be built and run by a small team. Following a realistic example, this book guides readers through the theory of Big Data systems and how to implement them in practice."); Hindman at Abstract and 1 ("We present Mesos, a platform for sharing commodity clusters between multiple diverse cluster computing frameworks, such as Hadoop and MPI.", "Clusters of commodity servers have become a major computing platform, powering both large Internet services and a growing number of data-intensive scientific applications. Driven by these applications, researchers and practitioners have been developing a diverse array of cluster computing frameworks to simplify programming the cluster. Prominent examples include MapReduce [18], Dryad [24], MapReduce Online [17] (which supports streaming jobs), Pregel [28] (a specialized framework for graph computations), and others [27, 19, 30]. It seems clear that new cluster computing frameworks will continue to emerge, and that no

framework will be optimal for all applications. Therefore, organizations will want to run multiple frameworks in the same cluster, picking the best one for each application. Multiplexing a cluster between frameworks improves utilization and allows applications to share access to large datasets that may be too costly to replicate across clusters.) Sharing improves cluster utilization and avoids per-framework data replication."); Kinesis Developer Guide at 1 ("Amazon Kinesis is a managed service that scales elastically for real-time processing of streaming big data. … Amazon Kinesis takes in large streams of data records that can then be consumed in real time by multiple data-processing applications that can be run on Amazon Elastic Compute Cloud (Amazon EC2) instances. … Amazon Kinesis applications read data from the Amazon Kinesis stream and process the read data in real time. The processed records can be emitted to dashboards, used to generate alerts, dynamically change pricing and advertising strategies, or emit data to a variety of other Amazon big data services such as Amazon Simple Storage Service (Amazon S3), Amazon Elastic MapReduce (Amazon EMR), or Amazon Redshift. Amazon Kinesis applications can also emit data into another Amazon Kinesis stream, enabling more complex data processing."); Scryer Part 1 ("To deliver the best possible experience to Netflix customers around the world, it is critical for us to maintain a robust, scalable, and resilient system. That is why we have built (and open sourced) applications ranging from Hystrix to Chaos Monkey. All of these tools better enable us to prevent or minimize outages, respond effectively to outages, and/or anticipate the kinds of operational gaps that may eventually result in outages. Recently we have built another such tool that has been helping us in this ongoing challenge: Scryer. Scryer is a new system that allows us to provision the right number of AWS instances needed to handle the traffic of our customers. But Scryer is different from Amazon Auto Scaling (AAS), which reacts to real-time metrics and adjusts instance

counts accordingly. Rather, Scryer predicts what the needs will be prior to the time of need and provisions the instances based on those predictions.

The MapReduce references, including Hadoop, "Google" MapReduce, and Amazon Elastic MapReduce, all relate to the underlying MapReduce process (documented at least as early as 2004) premised on distributing or parallelizing processing by first mapping a process across severs and then reducing the results to an output. Hadoop can be considered an open-source engine for running MapReduce, including a distributed file system (HDFS, analogous to Google's GFS), a processing layer for applying MapReduce, and (as of at least Hadoop 2.0) the YARN resource manager. Amazon Elastic MapReduce (EMR) can be considered as a cloud-hosted version of Hadoop, running on Amazon's AWS infrastructure. It introduced EMRFS as a file system.

See also below in this section regarding IBM System S, Branson 337, Branson 586, Branson 812, AbInitio, Larson, Wholey, and Atterbury.

The '424 and '425 patents[5] purport to improve our ability to "analyze" "vast amounts of information" "into knowledge or informed action" by allowing "the analysis of current, possibly complex and changing streaming data" with a system that can "self-assess its own operations and key intermediate factors in both the data stream and stored information and make changes to its own function to optimize function and maximize the probability of reliable conclusions." '424 patent at 4:11-30. To address deficiencies in linear or rigidly programmed data pipelines, the patents acknowledge a need for "a system that intelligently combines processing of a current data stream with the ability to retrieve relevant stored data…" and that has "the ability to monitor for both operational issues within its components and should be able to learn and react to intermediate

[5] The '424 patent and the '425 patent have a common specification. The cover pleadings generally cite only the '424 patent specification, but unless explicitly stated, identical language appears in the '425 patent specification.

determinations of the analyses it runs and also should be able to self-modify to maintain optimal operation." '424 patent at 4:46-54. A POSITA at the time would have been motivated to combine the identified prior art references because the references address the same or similar issues in the same or similar ways. *See* B Charts and C Charts.

As acknowledged or assumed in the background section of the patents, a POSITA would have been aware of various needs and opportunities associated with ever-increasing amounts of data, including real-time or stream data (see, e.g., '424 patent 3:39-4:30). A POSITA would also have understood that in data processing systems can be viewed as having a "data plane" and a "control plane." The data plane refers to how data is processed and moves through the system: e.g., its ingestion, its transformation into intermediate output, and the forward of that output to other stages of the system. The patents invoke data plane issues at, for example, '424 patent 4:31-45. The control plane concerns how the system is managed and orchestrated: what is monitored, what decisions are made based on that monitoring, and how those decisions cause other components to act. The patents invoke concerns about control plane issues at, for example, '424 patent 4:46-54. A POSITA would have understood that to a large extent, improvements to the data plane and improvements to the control plane can be both complementary and independent. Accordingly, a POSITA would have been motivated to combine references that disclose data processing systems with references that disclose data plane improvements or modifications—even if those references are nominally directed to systems with a different control plane architecture or are largely silent as to control plane—and with references that disclose control plane improvements or modifications—even if those references are nominally directed to systems with a different data plane architecture or are largely silent as to data plane.

As set forth in the book *Big Data* at pages 7-10 in a section entitled "Desired properties of a Big Data system," a POSITA would also have thought about "complexity" as well as "scalability" because systems were preferably high performant, resource efficient, but also "easy to reason about." A POSITA would have been concerned about, *inter alia*, robustness and fault tolerance, latency, scalability (including horizontal scalability, which relates to adding more resources), general applicability (e.g., to "financial management systems, social media analytics, scientific applications, social networking, or anything else"), and maintenance (including "anticipating when to add machines to scale, keeping processes up and running, and debugging anything that goes wrong in production"). *Big Data* pp. 7-10. This was well understood even before the book's publication in April 2015, and are also reasons why a POSITA would have been motivated to combine primary references with references that disclose improvements or features in any of these aspects.

Although collaboration among data scientists and practitioners is not new, open source systems both evidenced the practice of cross fertilization and combining various frameworks to implement comprehensive system and provided further encouragement for it. For example, many of the primary systems are open source community projects within the Apache Software Foundation ecosystem. That ecosystem also included additional frameworks and systems that addressed some of the concerns and issues set forth in the preceding paragraphs. As is reflected in the documents listed in the B and C exhibits for those systems, it was common for practitioners to deploy systems that combined multiple Apache frameworks or projects or that combined one or more Apache frameworks with other frameworks and systems, open source or otherwise. But the availability and cataloging of all of these projects on the Apache website particularly encouraged practitioners to combine Apache projects with other. This is even noted in some of the patent

literature. *See, e.g.*, Glines 489 at [0042] (referring to multiple Apache projects as well as other projects when discussing the implementation of its distributed data processing system).

By way of non-limiting example, Apache Flink (which originated from a University-based project called Stratosphere) leveraged and offered dataflow graphs, graph-informed monitoring, and control-plane orchestration across distinct worker systems. Apache Spark leveraged and offered DAG-based execution, graph-informed monitoring, and dynamic scheduling/re-scheduling of work across executors in response to monitored execution state. Google Dataflow provided support for monitoring-driven scaling and execution of pipeline stages across managed worker systems and included pipeline-as-graph representations and graph-derived stage structure at the control plane. Apache Storm leveraged and offered topology-based (graph/topology) stream processing and monitoring-driven reassignment of work across worker nodes. Even to the extent some of these frameworks or systems were nominal alternatives to each other, in practice they were applied together or aspects of them were combined together, because the collaborative community and the open source nature of the frameworks encouraged practitioners to adopt solutions that met their needs.

Non-limiting evidence of this can be found at pages for many of the above-referenced projects that set out other projects within the "ecosystem" or that are "related" or supplemental. See, e.g., the Apache Ecosystem page (a version antedating the priority date is available at https://web.archive.org/web/20150905080356/https:/cwiki.apache.org/confluence/display/KAFKA/Ecosystem), presenting a community-reported list of tools understood to integrate with Kafka, including Storm, Samza, Spark Streaming, Hadoop, Elastic Search, AWS, Flume, and Ganglia. The Spark project had a similar page, with an antedating version archived at https://web.archive.org/web/20150905133522/https:/cwiki.apache.org/confluence/display/SPAR

K/Supplemental+Spark+Projects. This Supplemental Spark Projects page refers to integration with Mesos, ElasticSearch, and IBM cluster management software, and identifies a handful of other projects and applications that use Spark. See also Ramesh ("Samza's architecture is composed of 3 components… The actual implementation of the streaming layer and execution layer is pluggable. Streaming implementation can be provided via any of the existing implementations: Kafka (topics) or Hadoop (a directory of files in HDFS) or a database (table). Similarly, systems like Apache YARN and Apache Mesos can be plugged-in for job execution systems. Samza has in-built support for Apache YARN and Apache Kafka.").

Similarly, the Amazon Kinesis Details page archived at https://web.archive.org/web/20150905074423/https:/aws.amazon.com/kinesis/details/ and the Kinesis Developer Resources page archived at https://web.archive.org/web/20150905065309/https:/aws.amazon.com/kinesis/developer-resources/ furtherer demonstrating that Kinesis, like most of the Apache systems and even the closed-source systems charted as primary references, had an API or APIs that facilitated integration with other systems, including other primary and secondary references disclosed in these limitations. These pages specifically refer to integration with ElasticSearch, Storm, Elastic MapReduce, and Spark Streaming.

Despite any differences in emphasis, the systems in the Apache Framework represented successive and overlapping developments in distributed data processing architectures. As stated above, Apache Hadoop and its MapReduce execution model represent an early and foundational distributed computing framework, derived from Google's MapReduce model and widely deployed by the mid-2000s. Hadoop helped popularize the core concepts of distributed task execution, centralized job coordination, and execution of data processing workloads across clusters of

commodity computing systems. These concepts form the baseline for many subsequent distributed data processing systems. Apache Spark was introduced in part as an evolution beyond MapReduce, retaining distributed execution across clusters while introducing in-memory processing and explicit representation of computation as directed acyclic graphs (DAGs) of stages and transformations. Spark's DAG-based execution model, centralized driver, and task scheduling mechanisms reflect a direct progression from earlier MapReduce-style systems toward more flexible, graph-oriented pipeline execution. Apache Storm and Apache Flink (including its predecessor, Stratosphere) were developed in part to address continuous and low-latency stream processing use cases. Both systems represent streaming computations as explicit topologies or dataflow graphs composed of operators connected by data streams, and both employ centralized control components that monitor execution and assign or reassign work across distributed worker nodes. These systems extend the same fundamental distributed pipeline concepts to unbounded data streams. These Apache systems are technically interrelated and were studied, developed, and evaluated by largely overlapping community of interested practitioners. They share common architectural principles, including distributed execution across clusters, representation of computation as multi-stage pipelines or graphs, centralized control components that monitor execution, and dynamic scheduling or orchestration of work across worker systems. By the priority date they were well established across multiple, interoperating open-source platforms and widely described in public documentation, academic literature, and industry practice. A POSITA would have been motivated to combine any of these with any other of these, and for similar reasons a POSITA would have been motivated to combine any non-Apache reference with any of the Apache references and any of the Apache references with any of the non-Apache references.

A POSITA also would have been motivated to combine references that processed data in similar ways. Some system and frameworks are oriented to batch processing, other may be more directed to one-at-a-time data processing, and others are directed to stream (sometimes called micro-batch) processing. A POSITA implementing a solution suited for a particular processing approach would have been motivated to combine systems that were well-suited for that type of approach. But a POSITA would not have been discouraged from combining references that were primarily pitched at different styles of data processing and would even have been motivated to do so for the at least the other reasons set forth in these contentions.

By 2015 certain systems were so foundational and well established that a POSITA would have combined them or the solutions they implemented with any of the primary references. For example, MapReduce and the Apache Hadoop implementation of it were foundational distributed job orchestration references that included monitoring of job/task execution and reassignment/re-execution of tasks on distinct worker machines. Indeed, MapReduce and Hadoop are explicitly mentioned by *Big Data* (throughout) and Glines 489 (at [0005] and [0029]).

A POSITA would also have understood that certain companies and entities were active in the field and been motivated to look to and combine references from those companies and entities. For example, IBM is the assignee of numerous patents in the field, including the Branson and Amini patents (which a POSITA would have been motivated to combine by dint of both having the same assignee). A POSITA would have been aware that by the time of the alleged invention, Microsoft, HP, Amazon, Netflix, Alibaba, Twitter (which originated Mesos), Google, LinkedIn (which originated Samza and Kafka), and Yahoo were all active in the field, either because they had their own needs or because they helped their customers in this space (or both). A POSITA

would have been motivated to combine references associated with these and with other entities active in the field with any of the primary references.

Further as to IBM, its InfoSphere product line included InfoSphere Streams, which itself was related to work on System S and the associated SPADE project. Branson, the first named inventor on a primary reference and three secondary referneces, and Amini, the first named inventor on a primary reference, were both involved with System S and with InfoSphere. See https://www.linkedin.com/in/michael-branson-24a34888/ and https://research.ibm.com/people/lisa-amini. These relationships, which were known at the time, would have motivated a POSITA to combine any and all of these references with each other and, for the reasons set forth elsewhere in this section, with any of the other primary references.

Another significant vendor or player in the relevant data processing space was AbInitio. *See, e.g.*, AbInitio BlogSpot Post. AbInitio was, and is, a vendor of "extract, transform, and load" (ETL) infrastructure that works at a very large scale. The Larson, Wholey, and Atterbury references were all invented by AbInitio employees and assigned to AbInitio. A POSITA would have understood that AbInitio's solutions and patent applications were a potentially fruitful source of information regarding fundamental approaches to issues that are common across all data processing platforms that need to operate at scale, and for that reason (as well as for other reasons set forth in this section) would have been motivated to combine those references with any of the primary references.

Just as many of the systems in use at the time had public lists of related projects, as mentioned above, deployers of those systems were also often known. In other words, a POSITA considering a particular prior art system would have been motivated and readily able to identify a list of companies deploying that system, and then been motivated to (1) locate presentations and

videos from that company or its employees discussing that deployment (such content was commonly made publicly available and presented via the company's website, third-party developer channels, and/or technical conferences and trade shows), (2) consider the combinations and modifications disclosed in such material, (3) locate and consider other documents associated with that company, such as relevant patents or technical articles. For example, the Apache Kafka community published a "Powered By" page at (a version antedating the priority date is available at https://web.archive.org/web/20150810144926/https:/cwiki.apache.org/confluence/display/KAFKA/Powered+By). It lists LinkedIn, Yahoo, Twitter (referencing Storm), Netflix, Cisco, Oracle, and dozens more. This list, like similar lists, includes links to documentation or announcements about some of those implementations, further facilitating combinations. See also Tez Framework at 1357 ("Tez is deployed at Yahoo!, Microsoft Azure, LinkedIn and numerous Hortonworks customer sites, and a growing number of engines are being integrated with it.").

Non-limiting examples of announcements or explanations that companies provided about their deployments include, e.g., https://web.archive.org/web/20150908120110/http://yahooeng.tumblr.com/post/109994930921/kafka-yahoo, https://web.archive.org/web/20150729145607/http://techblog.netflix.com/2013/12/announcing-suro-backbone-of-netflixs.html, https://web.archive.org/web/20150821085236/https://blog.twitter.com/2015/handling-five-billion-sessions-a-day-in-real-time, and https://web.archive.org/web/20150822083446/http://highscalability.com/blog/2012/2/13/tumblr-

architecture-15-billion-page-views-a-month-and-harder.html, (links are to archive.org versions, although in at least some instances the original posts are still available).

Consistent with the field being one of fertile collaboration, a POSITA would not only have been aware of Borealis, but a POSITA would have understood that one of the very goals of this university-originated research project was to explore ideas that would inform and be informed by other developments in the field. See, e.g., Lindeberg at Section 4.1 ("Borealis, as part of a research project, has as purpose to identify and address shortcomings of current stream processing techniques."). References to Borealis in the bodies or bibliographies of other projects would also motivate a POSITA to consider Borealis both as a starting point for further modification and combination and as a source of disclosures with which to modify other references. See, e.g., Toshniwal (re Storm) at bibliography; Discretized Streams (re Mesos) at 3, 6, 7, 13, and bibliography; Lam (re Muppet) at 1822 and bibliography; Clasp at 365 and bibliography Borealis is also cited on the face of several patents in the space, including Amini 178.

The title alone of *Big Data* ("Big Data: Principles and Best Practices of Scalable Real-Time Data Systems") would be sufficient to motivate a POSITA to consider it in combination with any of the other primary references, because the term "big data" was commonly used at the time to refer to the field of art. Moreover, one of the authors of the books, Nathan Marz, was a significant advocate for Storm and was himself well-known in the field. The book was also credited at the time for popularizing, if not creating, the "Lambda Architecture" approach to data processing, and a POSITA considering any of the other primary references would also have considered those references through the lens of the Lambda Architecture and therefore have looked to *Big Data*. The book describes computing systems that are expressly distributed across clusters of machines and designed to process and analyze large-scale data by executing computations in

parallel across multiple nodes. It explains that such systems "use many machines working in parallel to store and process data," and that scalability is achieved by adding additional machines to the system, with computation and data processing inherently distributed across the cluster. The book further teaches architectures in which computation is performed across distributed batch, serving, and streaming layers operating on clustered hardware, including the execution of distributed processing frameworks (such as MapReduce-based batch computation and distributed stream-processing topologies) on clusters of computer systems. *See, e.g.*, Big Data at Chapters 1 (Sections 1.2–1.5), 6 (Sections 6.4–6.5), and 14 (Sections 14.1–14.3). As set forth above, these disclosures all would have been relevant to a POSITA considering any of the other primary references, who would have been motivated to modify those references in light of the disclosures in *Big Data*. *See also* B16 and C16.

For reasons including performance, scalability, and robustness, a POSITA would have been motivated to combine any reference not disclosing a distributed computing environment with a reference that discloses such an environment.

For some of the reasons already set forth above, a POSITA would have been motivated to combine any reference not disclosing streaming data ingestion with a reference that does, not least because of the increasing need, referenced above, to process the streams that resulted from instrumented physical system (e.g., beacons or monitoring systems), social media, financial systems, and the like.

Because multi-stage transformation pipelines were already well known and provided advantages with respect to at least flexibility and intelligibility/maintainability, a POSITA would have been motivated to combine any reference that does not teach multi-stage transformation pipelines with references that did.

A POSITA would have understood that distributed computational graphs provide advantages with respect to representing data flow and transformational stages, and would have been motivated to combine any reference that did not disclose them with any reference that did.

A POSITA would have been motivated to modify any of the primary references to use distributed processing or to combine any of them with references that taught distributed processing. Engineers are motivated to use distributed processing when the volume, velocity, or computational complexity of data may approach or exceed the practical limits of a single computer system. Distributed processing allows work to be divided across multiple machines, enabling higher throughput, lower end-to-end latency, improved fault tolerance, and horizontal scalability as workloads grow or fluctuate. These motivations are further reinforced by the availability of cloud computing infrastructure, which allows users to rapidly provision, scale, and decommission computing resources on demand. Cloud platforms make it straightforward to bring new computers online for processing when load increases and to release them when demand subsides, making distributed architectures both technically and economically attractive. As a result, distributed processing became a standard and well-understood design choice for large-scale and real-time data processing systems.

A POSITA would have been motivated to modify any of the primary references to use processing pipelines or to combine any of them with references that taught using processing pipelines. Engineers are motivated to structure processing logic as pipelines in order to decompose complex tasks into a sequence of simpler, well-defined stages. This concept predates computing and is exemplified by Henry Ford's automobile assembly line, introduced in 1913, which organized manufacturing as a series of sequential, specialized operations to increase efficiency and throughput. In computing systems, pipelines similarly promote modularity, code reuse, and clarity

by allowing each stage to perform a specific transformation on the data. Pipeline architectures also support streaming execution, enabling data to be processed incrementally as it arrives rather than waiting for complete datasets. This staged approach improves efficiency, simplifies development and debugging, and allows individual stages to be optimized or replaced independently.

A POSITA would have been motivated to modify any of the primary references to use multiple different pipelines or to combine any of them with references that taught using multiple different pipelines. In many systems, different stages of processing serve fundamentally different purposes and therefore benefit from separate pipelines. For example, one pipeline may ingest and normalize raw input data, another may analyze or enrich that data using computationally intensive algorithms, and a third may evaluate results to determine whether some action should be taken, such as generating alerts, triggering downstream workflows, or updating system state. Using multiple distinct pipelines allows each to be tailored to its specific function, performance requirements, and resource needs. This separation improves flexibility, avoids overloading a single pipeline with incompatible responsibilities, and enables independent development and scaling of different processing functions within the same system. This type of functional separation is well understood outside of computing and mirrors how work is commonly divided in human organizations. For example, when a building is being planned and built, different individuals follow different defined sequences of steps: architects, engineers, planning officials, construction contractors, financiers, etc. Different actors perform different pipelines of tasks tailored to their roles. A POSITA designing computer systems was motivated to apply the same organizational principles by dividing functionality across multiple processing pipelines, each responsible for a specific stage of work, rather than forcing all tasks into a single processing flow.

A POSITA would have been motivated to modify any of the primary references to use back-to-back or chained pipelines or to combine any of them with references that taught using back-to-back or chained pipelines. Engineers are often motivated to connect pipelines in sequence, such that the output of one pipeline becomes the input to another, in order to perform multi-stage processing where each stage builds on the results of the previous one. Chaining pipelines allows complex transformations to be expressed as a flow of data through successive stages, such as moving from raw data ingestion, to intermediate feature extraction, to higher-level analysis or decision-making. This approach simplifies system design, supports incremental processing, and enables intermediate results to be reused or routed to multiple downstream operations. Back-to-back pipelines are therefore a natural and predictable extension of modular pipeline design.

A POSITA would have been motivated to modify any of the primary references to use control mechanisms or to combine any of them with references that taught using control mechanisms. As distributed and pipelined systems grow in scale and complexity or are subject to review in regard to the potential for that, engineers are motivated to include monitoring and control mechanisms to ensure safe, reliable, and efficient operation. Monitoring and control are fundamental components of many engineered systems beyond computing, including aircraft flight control systems, automotive engine and braking systems, nuclear power plant safety systems, and industrial process control systems. In each case, sensors monitor system state and controllers take corrective action to maintain stability and performance. Similarly, in distributed computing systems, monitoring provides visibility into execution state, resource utilization, throughput, and latency, while control mechanisms use that information to manage execution, respond to faults, and prevent system instability or collapse.

A POSITA would have been motivated to modify any of the primary references to place monitoring and control functions on a separate computer system, or to combine any of them with references that taught placing monitoring and control functions on a separate computer system. Engineers are motivated to implement monitoring and control functionality on a different computer system from the systems performing data processing in order to prevent interference between control operations and workload execution. Processing nodes are often subject to high load, variable latency, or partial failure, and allowing those conditions to impact control functionality can jeopardize the entire system. If control logic cannot execute in a timely and reliable manner, the system may fail to respond to overloads or faults, potentially leading to cascading failures or system collapse. Separating control from workers ensures that monitoring and coordination remain responsive and authoritative, even when individual processing nodes are stressed or unavailable.

A POSITA would have been motivated to modify any of the primary references to trigger or activate execution on other computers based on monitoring, or to combine any of them with references that disclosed the same. Monitoring is more valuable if the system can act on what it observes. Engineers are therefore motivated to trigger execution on other computer systems in response to monitored conditions such as increased input rates, processing backlogs, excessive latency, resource exhaustion, or component failures. For example, a system may activate additional processing nodes when input volume rises, reassign work when a node fails, or redistribute tasks to relieve hotspots or bottlenecks. These behaviors mirror real-world operational needs, such as scaling capacity to meet demand or responding to faults to maintain service continuity. Trigger-based execution enables systems to adapt dynamically rather than relying on static configuration or requiring human involvement.

A POSITA would have been motivated to modify any of the primary references to store information about a processing pipeline, or to combine any of them with references that did the same. Engineers are motivated to store information about processing pipelines in order to describe how the system is intended to operate, including how processing stages are configured, how they are connected, and how data is expected to flow between them. Having stored pipeline information gives a monitoring or control system context that it can use to understand system behavior, diagnose problems, and make informed decisions about execution. Pipeline metadata provides a representation of system structure that allows control logic to reason about dependencies, data flow, and execution responsibilities. Persisting this information is therefore desirable, if not essential, for coordinating distributed execution and ensuring that the system operates as designed.

A POSITA would have been motivated to modify any of the primary references to represent connections between processing pipelines as a computational graph, or to combine any of them with references that did the same. Engineers are motivated to represent the connections between processing pipelines using a computational graph because graphs provide a natural, well-understood abstraction for describing relationships, dependencies, and data flow between processing components. A computational graph allows the flow of data from the output of one pipeline to the input of another to be expressed in a formal and inspectable manner, making it easier to reason about connectivity, routing, and execution dependencies. When pipelines span multiple computer systems, representing pipeline-to-pipeline connections as a distributed graph allows different portions of that connectivity information to be stored, accessed, and managed across those systems in a scalable and fault-tolerant manner. Using a computational graph to describe how outputs from one pipeline flow into another was a conventional and predictable design choice for engineers building distributed data-processing systems.

A POSITA would have been motivated to modify any of the primary references to use stored graph metadata to route pipeline output data, or to combine any of them with references that did the same. Engineers are motivated to use stored pipeline or graph metadata to determine how output data should be routed between processing components because doing so improves flexibility, scalability, and adaptability. By consulting stored graph information that describes pipeline connectivity, a system can dynamically determine where pipeline output messages should be transmitted next, even as pipelines are reconfigured, extended, or chained together. This allows routing decisions to be made based on the system's defined structure rather than fixed assumptions embedded in processing logic. Engineers are further motivated to store relevant portions of graph metadata locally on the computer systems performing routing decisions in order to enable fast and reliable access to that information. Accessing locally stored graph data reduces or avoids the need to retrieve routing information from remote systems across a network, which would introduce additional latency, increase network load, and create additional points of failure. Local availability of graph metadata allows routing decisions to be made promptly and deterministically, even in the presence of network congestion, partial failures, or transient communication delays. Storing and using local graph information for routing therefore improves performance, robustness, and scalability, and was a conventional and predictable design choice in distributed computing systems.

A POSITA would have been motivated to modify any of the primary references to remotely trigger pipeline execution on other computer systems, or to combine any of them with references that disclosed remotely triggering pipeline execution on other computer systems. Engineers are motivated to allow one computer system to cause another computer system to execute processing logic in order to separate coordination decisions from workload execution and to enable multiple

computer systems to perform work in parallel. Centralizing execution control allows processing tasks to be distributed across multiple machines, increasing throughput, improving responsiveness, and avoiding bottlenecks that would arise if all work were performed by a single system. If execution decisions were confined to the same systems performing data processing, control functionality could be delayed or impaired by workload pressure or partial failures. Allowing one computer system to remotely trigger pipeline execution on other computer systems ensures that execution can be initiated, adjusted, or restarted in a controlled and timely manner, while enabling work to be shared among multiple computers rather than concentrated on one. Mechanisms for causing processes or procedures to be executed on remote computer systems were well known long before the priority date. For example, remote procedure call (RPC) mechanisms allow a program executing on one computer to invoke the execution of a procedure on another computer as if it were a local call, while the underlying system handles message transmission, parameter passing, and execution on the remote machine. RPC-based systems were standardized and widely deployed in distributed environments, including those described in Internet Engineering Task Force (IETF) publications such as RFC 5531, which specifies the Open Network Computing (ONC) Remote Procedure Call protocol. The existence and widespread use of RPC mechanisms demonstrates that engineers routinely designed systems in which one computer system causes other computer systems to execute software instructions so that multiple computers could actively participate in processing, making such remote triggering of execution a known, conventional, and predictable design practice.

A POSITA would have been motivated to modify any of the primary references to process data substantially in real time, or to combine any of them with references that disclosed processing data substantially in real time. Engineers are motivated to process data substantially in real time

in, e.g., scenarios where users, downstream systems, or automated decision processes are waiting on results, or where excessive latency would reduce the usefulness of the output. Real-time or near-real-time processing is particularly important in latency-sensitive contexts such as monitoring systems, alerting systems, operational analytics, and interactive or streaming applications. Processing data as it arrives avoids or reduces buffering delays, enables timely responses, and improves overall system responsiveness. As a result, applying processing pipelines to streaming data in substantially real time was a well-understood and routine design objective for engineers building distributed data-processing systems.

3. '627 Patent Obviousness

One of the Following Primary References	**In Combination With The Following Secondary References**
Brezinski	Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Choudhury	Brezinski, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Durairaj	Brezinski, Choudhury, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Huistra	Brezinski, Choudhury, Durairaj, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, , MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Lin	Brezinski, Choudhury, Durairaj, Huistra, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan

Muddu	Brezinski, Choudhury, Durairaj, Huistra, Lin, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Sekar	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Zhang	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Zhang2	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Apostolopoulos	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Coull	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Peer	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Muddu2	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Muddu3	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan

Crabtree	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
MulVAL	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
NetSPA	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
ACDP	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree2	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree3	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree4	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree5	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree6, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree6	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree7, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan

Crabtree7	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree8, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree8	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree9, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan
Crabtree9	Brezinski, Choudhury, Durairaj, Huistra, Lin, Muddu, Sekar, Zhang, Zhang2, Apostolopoulos, Coull, Peer, Muddu2, Muddu3, Crabtree, MulVAL, NetSPA, ACDP, Crabtree2, Crabtree3, Crabtree4, Crabtree5, Crabtree6, Crabtree7, Crabtree8, Shu, Lee, Coffman, Fang, Diehl, Alexander, Bryant, Mota, Hasan

PANW may assert obviousness based on the primary reference alone, or a combination of a primary reference and (1) a single one of the secondary references identified in the chart associated with that primary reference; (2) all secondary references identified in the chart associated with that primary reference; or (3) any combination of a subset of secondary references identified in the chart associated with that primary reference. In addition to the reasons identified above, one of ordinary skill in the art would be motivated to combine any of the references identified above because they are in the same field of network cybersecurity. Because the references are in the same field, address the same technology, and are intended to solve the same general problems, one of ordinary skill in the art would have found it obvious to combine the references, rendering the asserted patent obvious. *See, e.g.*, Choudhury at 1:17-23 ("The number and sophistication of cyber attacks on governments, private-sector companies, and individuals have grown dramatically in recent years. To counter these alarming trends, computer security personnel may attempt to detect cyber attacks in their early stages, so that defensive actions may be taken to avert, or at least mitigate, damage or loss."); Durairaj at Abstract ("A lateral movement application identifies lateral movement (LM) candidates that potentially represent a security threat.

Security platforms generate event data when performing security-related functions, such as authenticating a user account. The disclosed technology enables greatly increased accuracy identification of lateral movement (LM) candidates…"); Lin at 2:33-37 ("Detecting anomalous events that occur across multiple available sources is described. In some embodiments, an anomalous event comprises an incident and/or an entity associated with that incident that is suspected as being associated with a security threat."); Muddu at Abstract ("A security platform employs a variety techniques and mechanisms to detect security related anomalies and threats in a computer network environment."); Crabtree at Abstract ("A system for cybersecurity rating using active and passive external reconnaissance, that uses a web crawler that sends message prompts to external hosts and receives responses from external hosts, a time-series data store that produces time-series data from the message responses, and a directed computational graph module that analyzes the time-series data to produce a weighted score representing the overall cybersecurity state of an organization."); Coull at Abstract ("A method for performing cyber-security analysis includes generating a semantic graph in which each object is represented as a node, and each event associated with an object is represented as an edge.").

The '627 Patent purports to address the problems with managing cybersecurity for entities within a network infrastructure. *See, e.g.*, '627 Patent at 2:34-48. The '627 Patent purports to solve this problem with a "system and method for correlating network event anomalies using active and passive reconnaissance to identify attack information, that can identify attack patterns and points of origin based on observed anomalies and their relationships to other observed network events." *Id.* at 2:52-57. In particular, the '627 Patent describes and claims a "cyber-physical graph" of an organization "comprising nodes representing entities associated with the organization and edges representing relationships between entities associated with the organization" and

"identifying an anomalous event based on analysis of cyber-physical graph." *Id.* at 3:49-4:6. A POSITA would be motivated to combine the identified prior art references because the references address the same or similar issues in the same or similar ways. *See* Exs. D1-D26. Persons of ordinary skill in the art would have recognized that there were no significant impediments (technological or otherwise) to combining elements and features from the identified references. Indeed, the prior art references evidence a mixing-and-matching of the same basic features into different combinations.

Developments in the art would also provide motivation to combine prior art references in ways that would take advantage of those developments. For example, well before the claimed priority date of the '627 patent organizations were increasingly reliant on the use of a variety of different computing devices, used by a variety of users to connect with online services, which led to an increase in cybersecurity threats. The need for solutions to the problems associated with such a large number of connections, including the associated security risks, was well-recognized. For example, Brezinski explained that "Providers of online services or other computing services may deploy a large number of computing devices to execute processes and services within a computing environment. Such large deployments may enable an organization to maintain quality of service under a variety of operating conditions. However, a complex deployment of a large number of computing devices may create challenges when ensuring the security, confidentiality, and integrity of the information stored on the computing devices." Brezinski at 1:6-14. Similarly, Choudhury noted that "The number and sophistication of cyber attacks on governments, private-sector companies, and individuals have grown dramatically in recent years. . . . The problem of detecting cyber attacks can be viewed as a problem of identifying warning patterns in computer network traffic." Choudhury at 1:17-26; *see also* Durairaj at 1:7-8 ("Activity detection, both

friendly and malicious, has long been a priority for computer network administrators."); Crabtree at 2:49-64 ("IPv4 address space consists of 2^{32}, or almost 4.3 billion possible IP addresses, which is within the giga-order of magnitude. Today's computers have CPUs with gigahertz clock rates and gigabytes of RAM and storage. Today's networks allow band-widths exceeding 1 gigabit per second. This makes iterations over the entire IPv4 space possible within a comparatively short time period. Today's port scanning software is able to perform massive port scans up to the complete IPv4 space within minutes. Once a device is connected to the Internet, it will be almost immediately scanned for open ports and services. Most of these scans are done by anonymous individuals, in particular targeting at finding and exploiting system vulnerabilities. However, there is also a variety of individuals and organizations openly practicing massive port scanning and pursuing different objectives.").

In light of the teachings of the prior art, a POSITA would have readily appreciated the benefits of storing in memory a representation of a first graph, wherein the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises representations of a first plurality of edges, wherein the first graph is a directed graph. Use of such a directed computation graph to map the relationships between entities within an organization in order to manage those relationships and quickly identify potential security risks was well-known. As shown in the prior art references herein, many companies and individuals utilized this approach and recognized its benefits. *See* Exhibits D1-D26. For example, Brezinski discloses that "This disclosure describes implementations of systems, devices, methods, and computer-readable media for determining a graph that correlates a plurality of events detected on host devices, and for identifying anomalous activity by traversing the graph according to one or both of a rarity metric and a risk metric associated with elements of the graph." Brezinski at

2:8-13. Similarly, Choudhury discloses that "A warning pattern can be modeled as a directed graph (query graph) in which vertices represents machines, Internet Protocol ("IP") addresses or other network entities, and edges represent messages, communications, etc. between the network entities. A query graph can convey how an attack, intrusion, etc. propagates through the computers of a network." Choudhury at 1:26-32. Durairaj teaches that "One of the challenges in obtaining a 'big picture' view is the scale of the problem. As the number of network devices grows and the complexity of device operations increases, it is increasingly more difficult to correlate actions of attackers across network devices, especially when the attackers mask themselves behind different user names or system privileges." Durairaj at 3:9-14. To address this problem, Durairaj discloses that "A graph is created [that is] comprised of nodes which represent entities, and edges (also referred to as "connections") between nodes which represent login or other association activity between entities. Nodes or edges of the graph can be associated with various data, such as event(s), event segments, node(s), feature(s), feature vector(s), etc., and the nodes/edges/various data can have associated weights. A macro may be used to refine a population of LM candidates. A macro is data that can be used to identify various event-related data, e.g., specific tasks associated with a particular phase of an attack, among others." *Id.* at 4:20-31; *see also, e.g.*, Crabtree at 2:65-3:3 ("What is needed is a system and method for performing active and passive external reconnaissance using network scanning, that feeds information into a distributed computational graph data pipeline to produce hybrid graph/time-series data for use in producing cybersecurity ratings for organizations."); *id.* at 3:10-18 ("The aspects described herein provide a system and methods for cybersecurity rating using active and passive external reconnaissance, comprising a web crawler that send message prompts to external hosts and receives responses from external hosts, a time-series data store that produces time-series data from the message responses, and a

directed computational graph module that analyzes the time-series data to produce a weighted score representing the overall cybersecurity state of an organization.”).

As taught in these prior art references and others the use of such computational graphs to monitor security of a system and identify anomalies was beneficial for numerous reasons, including because “[b]y correlating event data in a graph, and traversing the graph according to risk and rarity of events, implementations may enable the detection of an attack that involves a complex chain of access in a computing environment.” Brezinski 4:34-38; *see also* Durairaj at 7:13-19 (“As a result of the more accurate identification of LM candidates enabled by the disclosed technology, threats can be more accurately identified. By more accurately identifying threats, resources, such as compute resources or humans in the form of IT security professionals, can be more productively utilized to identify those threats that exhibit actual malicious LM.”); Coull at 3:66-4:17 (“Some known cyber-security systems interfaces organize critical alerts, such as indicators of compromise (IOCs) into a global timeline or graph of associated alerts (e.g., within a “security operations center”). Such representations, however, are typically voluminous (e.g., requiring an analyst to scroll through many pages), and can include benign and/or duplicative information, making it cumbersome/difficult for an expert or analyst to readily discern/detect significant cyber-security threats. Without the ability to readily discern a cyber-security threat, the analyst often cannot timely mitigate or remediate that threat. Systems and methods of the present disclosure address the foregoing shortcomings by generating and dynamically updating a semantic graph and automatically propagating changes to threat scores (e.g., in response to newly-detected alerts) within the semantic graph, providing a more concise and efficient representation of the overall cyber-security condition of a system or network.”). A POSITA would have been motivated to provide this capability in order to facilitate the ability to efficiently monitor for and identify

potential security threats in network computing environments that would otherwise be difficult or impossible to detect. These benefits and others disclosed in the cited prior art references would have motivated a POSITA to look to these references and combine their teachings. A POSITA would have been motivated to combine the cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to include within a plurality of entities in the graph a plurality of accounts and a plurality of resources. The inclusion of accounts and resources in the entities of a computation graph and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, providing a visualization of the devices and resources in a security infrastructure. Brezinski, for example, discloses "As shown in FIG. 4, the graph 118 may include vertices 402 each corresponding to an entity such as a host device 102, a process 104, or a service 106." Brezinski at 8:29-42. Similarly, Crabtree discloses that "According to the aspect, a cyber-physical system graph may comprise a visualization of hierarchies and relationships between devices and resources in a security infrastructure, contextualizing security information with physical device relationships that are easily understandable for security personnel and users." Crabtree at 18:66-19:4. Crabtree further discloses that "time-series information … may be received and incorporated, updating CPG information as changes occur and events are logged. This information may then be used to produce 1104 a graph visualization of users, servers, devices, and other resources correlating physical relationships (such as a user's personal computer or smartphone, or physical connections between

servers) with logical relationships (such as access privileges or database connections), to produce a meaningful and contextualized visualization of a security infrastructure that reflects the current state of the internal relationships present in the infrastructure." *Id.* at 19:11-22. A POSITA would have been motivated to use known techniques for including within a plurality of entities in a graph a plurality of accounts and a plurality of resources because the purpose of a computation graph is to map relationships between such entities. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to include as each edge of the first plurality of edges of the graph a respective relationship between a respective pair of entities. The inclusion of relationships between entities as the edges of a computation graph and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, providing a visualization of the relationships between devices and resources in a security infrastructure and contextualizing security information with physical device relationships that are easily understandable for security personnel and users. Brezinski, for example, discloses "a schematic illustrating an example graph as an arrangement of event data, including vertices corresponding to entities such as devices, processes, or services in a computing environment, and including edges corresponding to events in which one entity accesses another entity." Brezinski at 1:27-31. Similarly, Crabtree discloses that "According to the aspect, a cyber-physical system graph may comprise a visualization of hierarchies and relationships between devices and resources

in a security infrastructure, contextualizing security information with physical device relationships that are easily understandable for security personnel and users." Crabtree at 18:66-19:4. Crabtree further discloses that "time-series information … may be received and incorporated, updating CPG information as changes occur and events are logged. This information may then be used to produce 1104 a graph visualization of users, servers, devices, and other resources correlating physical relationships (such as a user's personal computer or smartphone, or physical connections between servers) with logical relationships (such as access privileges or database connections), to produce a meaningful and contextualized visualization of a security infrastructure that reflects the current state of the internal relationships present in the infrastructure." *Id.* at 19:11-22. A POSITA would have been motivated to use known techniques for including as each edge of the first plurality of edges of the graph a respective relationship between a respective pair of entities because the purpose of a computation graph is to map relationships between such entities. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to receive streaming data comprising time-stamped data about events relating to one or more entities of the first plurality of entities. The receipt of time stamped data about events relating to entities on a network was a standard aspect of security monitoring and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, receiving and updating the graph information as changes occur and events are logged to reflect the current state of the internal

relationships present in the infrastructure. Brezinski, for example, discloses "An edge may include a timestamp attribute indicating a time (e.g., and date) when the event occurred." Brezinski at 2:45-64. Crabtree discloses that "time-series information … may be received and incorporated, updating CPG information as changes occur and events are logged. This information may then be used to produce 1104 a graph visualization of users, servers, devices, and other resources correlating physical relationships (such as a user's personal computer or smartphone, or physical connections between servers) with logical relationships (such as access privileges or database connections), to produce a meaningful and contextualized visualization of a security infrastructure that reflects the current state of the internal relationships present in the infrastructure." Crabtree at 19:11-22; *see also* Durairaj at 6:40-45 ("The event data include time-related data, such as timestamp data, which enables some or all of the event data to be sequenced. The LM security application analyzes the timestamp data to sequence the event data, and creates a data structure which represents an associated time constrained graph."). A POSITA would have been motivated to use known techniques for receiving streaming data comprising time-stamped data about events relating to one or more entities because it would enable the tracking of specific network activities and potential threats and accurately reflect the current state of the internal relationships present in the infrastructure. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to, based on a first portion of the streaming data, identify a first entity that does not correspond to any of the first plurality of nodes, wherein

the first entity is not of the first plurality of entities. The capability of identifying from streaming data an entity that does not correspond to any of the first plurality of nodes and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, detecting new objects or events as they occur over time. Brezinski, for example, discloses "At 1014, on receiving additional event data 110 the graph 118 may be modified to incorporate one or more additional edges 404 corresponding to recently detected events on the host device(s) 102. The graph 118 may also be modified to incorporate one or more additional vertices 402, in cases where the recently detected events involve entities not already described by vertices 402 in the graph 118." Brezinski at 17:23-29. *See also* Coull at 1:63-67 ("In some embodiments, the semantic graphs are continuously updated, for example in response to detection, over time, of one or more new objects, one or more new events, and/or one or more newly received alerts."); Crabtree at 18:53-63 ("This collated time-series data may then be used to produce a visualization 1004 of changes over time, quantifying collected data into a meaningful and understandable format. As new events are recorded, such as changing user roles or permissions, modifying servers or data structures, or other changes within a security infrastructure, these events are automatically incorporated into the time-series data and visualizations are updated accordingly, providing live monitoring of a wealth of information in a way that highlights meaningful data without losing detail due to the quantity of data points under examination."). A POSITA would have been motivated to use known techniques for identifying a first entity that does not correspond to any of the first plurality of nodes because the purpose of using a computational graph to monitor for network security risks is to identify potential threats from entities on the network. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to

yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to, based on a second portion of the streaming data that is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges. The capability of identifying from streaming data a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, detecting new relationship between objects or events as they occur over time. Brezinski, for example, discloses "[a]t 1014, on receiving additional event data 110 the graph 118 may be modified to incorporate one or more additional edges 404 corresponding to recently detected events on the host device(s) 102. The graph 118 may also be modified to incorporate one or more additional vertices 402, in cases where the recently detected events involve entities not already described by vertices 402 in the graph 118." Brezinski at 17:23-29. *See also* Coull at 1:63-67 ("In some embodiments, the semantic graphs are continuously updated, for example in response to detection, over time, of one or more new objects, one or more new events, and/or one or more newly received alerts."); Crabtree at 18:53-63 ("This collated time-series data may then be used to produce a visualization 1004 of changes over time, quantifying collected data into a meaningful and understandable format. As new events are recorded, such as changing user roles or permissions, modifying servers or data structures, or other changes within a security infrastructure, these events are automatically incorporated into the time-series data and visualizations are updated accordingly, providing live monitoring of a wealth of information in a

way that highlights meaningful data without losing detail due to the quantity of data points under examination."). A POSITA would have been motivated to use known techniques for identifying a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges because the purpose of using a computational graph to monitor for network security risks is to relationships between entities that indicate potential security threats. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a first node corresponding to the first entity and a representation of a first edge corresponding to the first relationship, wherein the first node is not of the first plurality of nodes and the first edge is not of the first plurality of edges. The capability of modifying a computation graph representation in memory to include such representations of entities and edges and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, the ability to graphically represent potential security threats. Brezinski, for example, discloses "The anomalous activity data 122 may be stored in memory on the analysis device(s) 114, or elsewhere, and made available for further analysis. . . . In some cases, the anomalous activity data 122 may be graphically presented as an overlay to the full graph 118 or as an overlay to a surrounding or nearby region of the graph 118 to facilitate analysis. In some cases, the anomalous

activity data 122 may include a description of one or more events as tuples of vertex-edge-vertex, including the first entity ID 204 and the second entity ID 206 associated with the vertices 402. The tuples may also include various attributes of the edge 404, including but not limited to the timestamp attribute 406, the event type attribute 408, the user attribute 410, the risk metric 412, or the rarity metric 502." Brezinski at 12:12-34. *See also* Coull at 1:63-67 ("In some embodiments, the semantic graphs are continuously updated, for example in response to detection, over time, of one or more new objects, one or more new events, and/or one or more newly received alerts."); Crabtree at 18:53-63 ("This collated time-series data may then be used to produce a visualization 1004 of changes over time, quantifying collected data into a meaningful and understandable format. As new events are recorded, such as changing user roles or permissions, modifying servers or data structures, or other changes within a security infrastructure, these events are automatically incorporated into the time-series data and visualizations are updated accordingly, providing live monitoring of a wealth of information in a way that highlights meaningful data without losing detail due to the quantity of data points under examination."). A POSITA would have been motivated to use known techniques for modifying a computation graph representation in memory to include such representations of entities and edges because it facilitates the analysis and identification of potential security threats by system administrators, security analysists, etc. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to identify, based on the modified representation of the first graph, an attack path that could be involved in an attack involving the first entity. The use of a modified computational graph to identify an attack path and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, the ability to more quickly and easily identify entities and relationships involved in a potential security breach. Brezinski, for example, discloses "In some cases, the anomalous activity data 122 may be accessed and examined by system administrators, security analysts, system developers, or other personnel to determine whether the events included in the anomalous activity data 122 indicate an attack, exploit, or other form of security breach. In such cases, the anomalous activity data 122 may be presented in a graphical form (e.g., as the subset of the graph 118) to facilitate analysis." Brezinski at 12:12-34. *See also* Crabtree at 18:31-42 ("In the event of an incident, the impact assessment score for any compromised accounts may be used to produce a 'blast radius' calculation 903, identifying exactly what resources are at risk as a result of the intrusion and where security personnel should focus their attention. To provide proactive security recommendations through a simulation module 125, simulated intrusions may be run 904 to identify potential blast radius calculations for a variety of attacks and to determine 905 high risk accounts or resources so that security may be improved in those key areas rather than focusing on reactive solutions."); Durairaj at 9:47-51 ("Based on the detected anomaly, the method may further determine a security threat based on the relationship information by identifying, in a relationship graph, a relationship path from a user through anomalies to a critical resource."). A POSITA would have been motivated to use known techniques for using of a modified computational graph to identify an attack path because it facilities more efficient monitoring of and identification of security risks. A POSITA would have been motivated to combine these

teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to identify an attack path by identifying a second entity that can be reached using the first entity, wherein the second entity corresponds to a second node, and the second node is related by one or more edges to the first node corresponding to the first entity in the modified representation of the first graph, identifying a third entity that can be reached using the second entity, wherein the third entity corresponds to a third node, and the third node is related by one or more edges to the second node in the modified representation of the first graph, and generating a report comprising an identification of the first entity and at least one of the second entity and the third entity. The use of a computational graph to identify second and third entities that may be reached by a first entity, and generating a report identifying these entities, and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, the ability to identify entities that are at risk from a security breach, including entities that may be reachable through other entities in the network. *See, e.g.*, Brezinski at 3:52-58 ("In some implementations, the graph may be traversed to identify anomalous activity data included in the graph. Starting from a start vertex, the graph may be traversed along at least a portion of the edges in the graph to identify those edges that include a rarity metric that satisfies a predetermined rarity threshold and that include a risk metric that satisfies a predetermined risk threshold."); Durairaj at 15:8-13 ("Relationship path 620 includes user nodes and device nodes interlinked by anomaly nodes A602, A604 and A606. In other words, the

relationship path 620 starts from user U601 through anomaly nodes A602, A604 and A606 to the device D607 designated as a critical resource of the computer network."); Crabtree at 18:31-42 ("In the event of an incident, the impact assessment score for any compromised accounts may be used to produce a 'blast radius' calculation 903, identifying exactly what resources are at risk as a result of the intrusion and where security personnel should focus their attention. To provide proactive security recommendations through a simulation module 125, simulated intrusions may be run 904 to identify potential blast radius calculations for a variety of attacks and to determine 905 high risk accounts or resources so that security may be improved in those key areas rather than focusing on reactive solutions."). A POSITA would have been motivated to use known techniques for identifying multiple entities that are connected through edges to a first entity because it enables identification of entities that are at risk from a security breach that may be reachable through other entities in the network. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated for the modified representation of the first graph to include a representation of a node corresponding to the first entity that is identified based on active reconnaissance results. The use of active reconnaissance results to identify threats and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, using time-series data graphing to identify vulnerabilities that would otherwise be obscured within a large dataset. Crabtree, for example, discloses: "FIG. 2B is

a process diagram showing a general flow of a process 210 for performing active reconnaissance using DNS leak information collection. In an initial step 211, publicly-available DNS leak disclosure information may be collected to maintain current information regarding known leaks and vulnerabilities. In a next step 212, third-level domain (TLDR) information may be collected and used to report domain risk factors, such as domains that do not resolve properly (due to malformed DNS records, for example). In a next step 213, a DNS trust map may be created using a hybrid graph/time-series data structure, using a graph stack service 145 and MDTSDB 120. This trust map may be produced as the output of an extraction process performed by a DCG 155 through a plurality of data pipelines 155 a, analyzing collected data and mapping data points to produce hybrid structured output representing each data point over time. In a final step 214, the trust map may then be analyzed to identify anomalies, for example using community detection algorithms that may discover when new references are being created, and this may be used to identify vulnerabilities that may arise as a byproduct of the referential nature of a DNS hierarchy. In this manner, DCG pipeline processing and time-series data graphing may be used to identify vulnerabilities that would otherwise be obscured within a large dataset." Crabtree at 12:36-60. A POSITA would have been motivated to use known techniques for identifying entities based on active reconnaissance results because representing such identified entities as graph nodes enables correlation, anomaly detection, and risk assessment across time-series data, thereby improving visibility into security-relevant relationships and behaviors that cannot be reliably detected from isolated or static data sources alone. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have

had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated for the modified representation of the first graph to include a representation of a node corresponding to the first entity that is identified based on passive reconnaissance results. The use of passive reconnaissance results and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, identifying externally visible entities, relationships, technologies, and content using publicly available information, and structuring that information in a manner that supports correlation, contextual analysis, and downstream vulnerability identification. Crabtree, for example, discloses: "FIG. 2A is a block diagram showing general steps 200 for performing passive network reconnaissance. It should be appreciated that the steps illustrated and described may be performed in any order, and that steps may be added or omitted as needed for any particular reconnaissance operation. In a step 201, network address ranges and domains or sub-domains associated with a plurality of targets may be identified, for example to collect information for defining the scope of further scanning operations. In another step 202, external sites may be identified to understand relationships between targets and other third-party content providers, such as trust relationships or authoritative domain name service (DNS) resolution records. In another step 203, individual people or groups may be identified using names, email addresses, phone numbers, or other identifying information that may be useful for a variety of social engineering activities. In another step 204, technologies used may be identified, such as types or versions of hardware or software used by an organization, and this may include collecting and extracting information from job descriptions (for example) to identify technologies in use by an organization (for example, a job description for an administrator familiar with specific database software

indicates that said software is in use within the organization). In another step 205, content of interest may be identified, for example including web and email portals, log files, backup or archive files, and other forms of sensitive information that may be contained within HTML comments or client-side scripts, as may be useful for vulnerability discovery and penetration testing activities. In another step 206, publicly-available information may be used to identify vulnerabilities that may be exploited with further active penetration testing." Crabtree at 12:4-35. A POSITA would have been motivated to use known techniques for identifying entities based on passive reconnaissance results because representing such passively identified entities as graph nodes enables systematic correlation of publicly available information, relationship mapping, and contextual analysis across multiple data sources, thereby improving visibility into an organization's external attack surface and associated security risks without requiring direct interaction with the target systems. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to use a cyber-physical graph. The use of a cyber-physical graph and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, modeling both physical and logical relationships among users, devices, services, and resources in a unified graph structure, and providing contextualized visibility into system state, access relationships, and security-relevant interactions that cannot be effectively captured otherwise. Crabtree, for example, discloses: "FIG.

11 is a flow diagram of an exemplary method 50966 for mapping a cyber-physical system graph (CPG), according to one aspect. According to the aspect, a cyber-physical system graph may comprise a visualization of hierarchies and relationships between devices and resources in a security infrastructure, contextualizing security information with physical device relationships that are easily understandable for security personnel and users. … time-series information … may be received and incorporated, updating CPG information as changes occur and events are logged. This information may then be used to produce 1104 a graph visualization of users, servers, devices, and other resources correlating physical relationships (such as a user's personal computer or smartphone, or physical connections between servers) with logical relationships (such as access privileges or database connections), to produce a meaningful and contextualized visualization of a security infrastructure that reflects the current state of the internal relationships present in the infrastructure." Crabtree at 18:64-22. A POSITA would have been motivated to use known techniques for using a cyber-physical graph because correlating physical entities with logical relationships within a single graph enables improved contextual analysis, risk assessment, and anomaly detection by allowing security-relevant events and access patterns to be evaluated in light of both physical topology and logical system interactions. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to generate a report that is an alert. The use of alerts and their related benefits were well-known by the time of the claimed priority date of

the ’627 patent. Those benefits include, inter alia, providing timely notification of detected vulnerabilities or security-relevant events. Crabtree, for example, discloses: “When a new vulnerability is discovered, a blast radius score may be assessed 1803 and the network's resiliency score may be updated 1804 as needed. A security alert may then be produced 1805 to notify an administrator of the vulnerability and its impact, and a proposed patch may be presented 1806 along with the predicted effects of the patch on the vulnerability's blast radius and the overall network resiliency score. This determines both the total impact risk of any particular vulnerability, as well as the overall effect of each vulnerability on the network as a whole. This continuous network assessment may be used to collect information about new vulnerabilities and exploits to provide proactive solutions with clear result predictions, before attacks occur.” Crabtree at 21:13-26. A POSITA would have been motivated to use known techniques for generating a report that is an alert because producing an alert enables rapid awareness, prioritization, and response by administrators. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to utilize software instructions that determine whether an event, of the events relating to one or more entities of the first plurality of entities, is anomalous, wherein determining whether the event is anomalous includes determining that the event relates to the first entity, determining at least one behavior pattern associated with the first entity, and comparing the event to the at least one behavior pattern. The use of software instructions for determining anomalous events based on entity-specific behavior patterns and its

related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, detecting deviations from established behavior patterns, identifying potentially malicious or high-risk activity associated with specific entities, and enabling proactive security analysis and response based on observed anomalies. Crabtree, for example, discloses: "a web crawler 115 may passively collect activity information, which may then be processed 802 using a DCG 155 to analyze behavior patterns. Based on this initial analysis, anomalous behavior may be recognized 803 (for example, based on a threshold of variance from an established pattern or trend) such as high-risk users or malicious software operators such as bots. These anomalous behaviors may then be used 804 to analyze potential angles of attack and then produce 805 security suggestions based on this second-level analysis and predictions generated by an action outcome simulation module 125 to determine the likely effects of the change." Crabtree at 17:58-18:3. A POSITA would have been motivated to use known techniques for determining whether an event relating to an entity is anomalous by comparing the event to behavior patterns associated with that entity because entity-specific behavioral baselining enables more accurate anomaly detection, reduces false positives relative to generic thresholds, and provides meaningful security context for assessing the significance of observed events. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to compare an event to the at least one behavior pattern using a threshold. The use of comparing an event to at least one pattern using a threshold

and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, providing an efficient mechanism for distinguishing normal activity from anomalous behavior, enabling consistent anomaly detection across large volumes of event data, and reducing false positives by requiring deviations to exceed a defined variance from an established pattern or trend. Crabtree, for example, discloses: "a web crawler 115 may passively collect activity information, which may then be processed 802 using a DCG 155 to analyze behavior patterns. Based on this initial analysis, anomalous behavior may be recognized 803 (for example, based on a threshold of variance from an established pattern or trend) such as high-risk users or malicious software operators such as bots. These anomalous behaviors may then be used 804 to analyze potential angles of attack and then produce 805 security suggestions based on this second-level analysis and predictions generated by an action outcome simulation module 125 to determine the likely effects of the change." Crabtree at 17:58-18:3. A POSITA would have been motivated to use known techniques for comparing an event to a behavior pattern using a threshold because threshold-based evaluation provides a predictable and scalable method for determining when deviations from expected behavior warrant further analysis or response. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to compare an event to at least one behavior pattern by determining that the first entity is associated with a resource, wherein the resource is a sensitive resource. The use of resource-sensitive context in evaluating events against behavior

patterns and its related benefits were well-known by the time of the claimed priority date of the ’627 patent. Those benefits include, inter alia, incorporating resource sensitivity into anomaly evaluation and prioritizing events based on the potential impact to sensitive resources. Crabtree, for example, discloses that “The blast radius metric for any other devices involved in the connection may also be checked 1605, to examine the context of the connection for risk potential (for example, an unknown connection to an internal data server with sensitive information may be considered a much higher risk than an unknown connection to an externally-facing web server).” Crabtree at 20:37-44. A POSITA would have been motivated to use known techniques for comparing an event to a behavior pattern based on an entity’s association with a sensitive resource because incorporating resource sensitivity into behavioral analysis provides critical context for assessing risk, enables differentiation between benign deviations and high-risk anomalies, and supports prioritization of security responses toward events that pose greater potential harm. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to utilize a plurality of physical computing machines. The use of a plurality of physical computing machines and its related benefits were well-known by the time of the claimed priority date of the ’627 patent. Those benefits include, inter alia, distributing computational workloads across multiple machines, improving scalability and performance when processing large volumes of event and time-series data, increasing system reliability and fault tolerance, and enabling parallel execution of data collection, analysis, and

graph-processing operations. Crabtree, for example, discloses that the disclosed techniques "may be implemented on hardware or a combination of software and hardware" and "may be implemented on one or more general-purpose computers associated with one or more networks, such as for example an end-user computer system, a client computer, a network server or other server system, a mobile computing device (e.g., tablet computing device, mobile phone, smartphone, laptop, or other appropriate computing device), a consumer electronic device, a music player, or any other suitable electronic device, router, switch, or other suitable device, or any combination thereof." Crabtree at 21:28-56. A POSITA would have been motivated to use known techniques for utilizing a plurality of physical computing machines because distributing processing across multiple physical machines enables efficient handling of concurrent data ingestion, analysis, and monitoring tasks, supports scalability as system demands grow, and yields predictable performance and reliability improvements. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to utilize a plurality of virtual computing machines. The use of a plurality of virtual computing machines and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, enabling flexible and scalable allocation of computing resources, supporting isolation of workloads, facilitating efficient deployment and management of distributed services, and allowing systems to scale or adapt dynamically in response to changing processing demands. Crabtree, for

example, discloses "at least some of the features or functionalities of the various aspects disclosed herein may be implemented in one or more virtualized computing environments (e.g., network computing clouds, virtual machines hosted on one or more physical computing machines, or other appropriate virtual environments)." Crabtree at 21:57-62. A POSITA would have been motivated to use known techniques for utilizing a plurality of virtual computing machines because virtualization provides a predictable and widely adopted mechanism for deploying and scaling distributed computing systems, improves resource utilization, and enables parallel execution of data collection and analysis processes without requiring dedicated physical hardware for each function. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated for at least one entity of the first plurality of entities in the graph to be at least one of a user, a place, a device, a resource, a group, or a service. The use of graph representations that include such entity types and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, enabling comprehensive modeling of security-relevant actors and components, capturing both physical and logical relationships among users, devices, services, and resources, and supporting contextual analysis of events and behaviors across heterogeneous entity types within a unified graph structure. Crabtree, for example, discloses that "time-series information … may be received and incorporated, updating CPG information as changes occur and events are logged. This information may then be used to produce 1104 a graph visualization of users, servers, devices, and

other resources correlating physical relationships (such as a user's personal computer or smartphone, or physical connections between servers) with logical relationships (such as access privileges or database connections), to produce a meaningful and contextualized visualization of a security infrastructure that reflects the current state of the internal relationships present in the infrastructure." Crabtree at 19:11-22. A POSITA would have been motivated to use known techniques for representing entities such as users, places, devices, resources, groups, and services within a graph because including these entity types enables accurate representation of real-world security infrastructures, facilitates correlation of events and behaviors across different classes of entities, and yields predictable improvements in visibility, analysis, and risk assessment. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated for the representation of the first graph comprises a representation of at least one node that does not correspond to an entity. The use of graph nodes representing abstract, logical, or analytical constructs—rather than only real-world entities—and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, enabling representation of events, states, relationships, transformations, or analytical constructs within a graph, supporting richer modeling of system behavior, and facilitating analysis of interactions and processes that cannot be accurately captured using only entity-based nodes. Crabtree, for example, discloses that within one embodiment of a directed computational graph, "transformations within transformation pipeline

are represented as directed graph with each transformation comprising a node and the output messages between transformations comprising edges," thereby explicitly teaching some graph nodes that do not correspond to entities. Crabtree at 6:65-7:2. A POSITA would have been motivated to use known techniques for including nodes that do not correspond to entities because representing non-entity constructs as graph nodes enables comprehensive modeling of system dynamics, event flows, and analytical processes, and yields predictable improvements in flexibility, expressiveness, and analytical capability of the graph. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to use software instructions that, based on a third portion of the streaming data, identify an anomalous event, store in the hardware memory a dependency tree for the anomalous event, and based on the dependency tree, determine a point of origin for the anomalous event. The use of graph-based anomaly detection, dependency tracking, and origin determination and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, enabling early detection of anomalous or suspicious behavior, preserving causal and temporal relationships among events, and facilitating root-cause analysis by tracing the origin and propagation of anomalous activity through a system. Huistra, for example, discloses: "Analysis of an organization's security and the threats it faces is nowadays often done using attack trees that describe all possible threats facing a system or organization…. this project focuses on generating attack trees automatically from a

given model that describes a system or organization. It improves upon previous efforts by providing an approach to identify all possible attacks from a given model in a more scalable manner, compared to the previous approach of constructing an attack graph, while remaining (security-)domain independent." Huistra at iii. A POSITA would have been motivated to use known techniques for identifying anomalous events, storing dependency trees, and determining points of origin because such techniques improve detection accuracy, reduce false positives, and enable actionable response by identifying the underlying cause of an anomaly rather than merely reporting its symptoms. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to use software instructions that analyze an attack path of the plurality of attack paths using a graph analysis algorithm. The use of graph analysis algorithms to evaluate attack paths and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, systematically evaluating sequences of relationships or connections within a graph, identifying feasible or high-risk attack paths through interconnected entities and resources, and enabling quantitative assessment of potential impact, reach, or propagation of an attack across a system. Crabtree, for example, discloses: "According to the aspect, an advanced cyber decision platform may continuously monitor a network 1701, informing a CPG in real-time of all traffic associated with people, places, devices, or services 1702. Machine learning algorithms detect behavioral anomalies

as they occur in real-time 1703, notifying administrators with an assessment of the anomalous event 1704 as well as a blast radius score for the particular event and a network resiliency score to advise of the overall health of the network. By automatically detecting unusual behavior and informing an administrator of the anomaly along with contextual information for the event and network, a compromised ticket is immediately detected when a new authentication connection is made." Crabtree at 20:58-21:3. A POSITA would have been motivated to use known techniques for analyzing attack paths using a graph analysis algorithm because graph-based path analysis provides a predictable and effective mechanism for modeling attacker movement through complex systems, identifying critical nodes or edges, and prioritizing defensive actions based on the structure and connectivity of the underlying graph. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

A POSITA would also have been motivated to use a representation of a first graph that includes an identification of a security vulnerability associated with the third entity, and wherein an attack path of the second plurality of attack paths is based on the security vulnerability. The use of graph representations that incorporate identified security vulnerabilities and their relationship to potential attack paths, and its related benefits were well-known by the time of the claimed priority date of the '627 patent. Those benefits include, inter alia, contextualizing vulnerabilities within the structure of a system, enabling analysis of how known vulnerabilities may be exploited to traverse entities and resources, and supporting identification and prioritization

of attack paths based on the presence and location of security weaknesses. Crabtree, for example, discloses: "According to the aspect, a baseline score can be used to measure an overall level of risk for a network infrastructure, and may be compiled by first collecting 1201 information on publicly-disclosed vulnerabilities, such as (for example) using the Internet or common vulnerabilities and exploits (CVE) process. This information may then 1202 be incorporated into a CPG as described previously in FIG. 11, and the combined data of the CPG and the known vulnerabilities may then be analyzed 1203 to identify the relationships between known vulnerabilities and risks exposed by components of the infrastructure. This produces a combined CPG 1204 that incorporates both the internal risk level of network resources, user accounts, and devices as well as the actual risk level based on the analysis of known vulnerabilities and security risks." Crabtree at 19:25-39. A POSITA would have been motivated to use known techniques for incorporating security vulnerabilities into a graph and basing attack-path analysis on those vulnerabilities because mapping known vulnerabilities to specific entities and resources within a graph enables systematic evaluation of exploitability, identification of feasible attack paths, and prioritization of remediation efforts based on the structure and connectivity of the affected system components. A POSITA would have been motivated to combine these teachings, and other similar teachings, with other cited references because it involved combining prior art elements according to known methods to yield predictable results and benefits. Given that such features had already been implemented in the prior art references, a POSITA would have had a reasonable expectation of success in combining the above references. *See, e.g.*, Exhibits D1-D26, L.

VI. INVALIDITY UNDER 35 U.S.C. § 112 (P.R. 3-3(C))

A. Enablement

To satisfy the enablement requirement, the patent specification must enable a POSITA, as of the filing date, to practice the full scope of the claimed invention without undue

experimentation. *Alza Corp. v. Andrx Pharms., LLC*, 603 F.3d 935, 941-42 (Fed. Cir. 2010); *Sitrick v. Dreamworks, LLC,* 516 F.3d 993, 999 (Fed. Cir. 2008). "Enabling the full scope of each claim is part of the quid pro quo of the patent bargain." *Sitrick,* 516 F.3d at 999. If a patent enables some embodiments within the scope of a claim, but not others, then the claim is invalid for lack of enablement. *Alza,* 603 F.3d at 939-43; *Sitrick*, 516 F.3d at 999-1001; *Auto. Tech. Int'l, Inc. v. BMW of N. Am., Inc.,* 501 F.3d 1274, 1281-85 (Fed. Cir. 2007). The patentee is "required to provide an adequate enabling disclosure in the specification; it cannot simply rely on the knowledge of a POSITA to serve as a substitute for the missing information in the specification." *Alza*, 603 F.3d at 941; *see also Auto. Tech*., 501 F.3d at 1283-84.

The Federal Circuit has identified several factors that a court may consider when deciding whether the specification requires undue experimentation: "(1) the quantity of experimentation necessary, (2) the amount of direction or guidance presented, (3) the presence or absence of working examples, (4) the nature of the invention, (5) the state of the prior art, (6) the relative skill of those in the art, (7) the predictability or unpredictability of the art, and (8) the breadth of the claims." *Alza,* 603 F.3d at 940. However, "when there is no disclosure of any specific starting material or of any of the condition under which a process can be carried out, undue experimentation is required." *Id*. at 941.

B. Written Description

For the written description requirement, each asserted patent must clearly allow a POSITA to understand and recognize that the patentee invented what is claimed. *Gentry Gallery, Inc. v. Berkline Corp*., 134 F.3d 1473, 1479 (Fed. Cir. 1998). The patent must demonstrate by disclosure in the specification to those skilled in the art that the patentee had "possession" of the subject matter asserted to be the claimed invention. *Vas-Cath Inc. v. Mahurkar*, 935 F.2d 1555, 1561 (Fed. Cir. 1991). The written description requirement is not sufficed by saying that undisclosed subject

matter would have been obvious or within the normal skill set of a POSITA. *ICU Medical, Inc. v. Alaris Medical Sys., Inc*., 558 F.3d 1368, 1377 (Fed. Cir. 2009). A written description that discloses only a certain method does not "necessarily support a broad claim as to every possible type of [method], no matter how different in structure or operation from the inventor's [discussion]." *LizardTech, Inc. v. Earth Resource Mapping, Inc*., 424 F.3d 1336, 1346 (Fed. Cir. 2005). That is, an inventor's description of one type of method does not entitle the inventor to claim "any and all means for achieving that objective." *Id.*; *see also ICU Medical,* 558 F.3d at 1377-79.

C. Indefiniteness

To satisfy the definiteness requirement of 35 U.S.C. § 112, the specification of a patent must "particularly point[] out and distinctly claim[] the subject matter which the [inventor or a joint inventor] regards as the invention." To meet this requirement, the "claims, when read in light of the specification and the prosecution history, must provide objective boundaries for those of skill in the art." *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1371 (Fed. Cir. 2014) (citing *Nautilus, Inc. v. Biosig Instruments, Inc.,* 134 S. Ct. 2120, 2130 (2014)). Claims are invalid as indefinite where they fail this test. *See id.*; *Allen Eng'g Corp. v. Bartell Indus., Inc.,* 299 F.3d 1336, 1349 (Fed. Cir. 2002).

Clear and definite boundaries to patent claims are important to our patent system. Patents are monopolies. The claims serve a public notice function by informing the public of the boundaries of that monopoly. "The inventor must 'inform the public during the life of the patent of the limits of the monopoly asserted, so that it may be known which features may be safely used or manufactured without a license and which may not.'" *United Carbon Co. v. Binney & Smith Co.,* 317 U.S. 228, 232 (1942) (quoting *General Elec. Co. v. Wabash Corp.,* 304 U.S. 364, 369 (1938)).

"Stated differently, 'if patent infringement can be compared to trespassing, the claims serve as the boundary markers that define what is, or is not, an encroachment on the inventor's exclusive territory.'" 3 Chisum on Patents § 8.03 (quoting Alan L. Durham, Patent Law Essentials: A Concise Guide 66 (1999)).

Therefore, "a patent must be precise enough to afford clear notice of what is claimed, thereby 'appris[ing] the public of what is still open to them.'" *See, e.g.*, *Nautilus, Inc. v. Biosig Instruments, Inc.,* 134 S. Ct. 2120, 2129 (2014). Unclear claims "discourage invention" because they create a "zone of uncertainty" that "enterprise and experimentation may enter only at the risk of infringement." *United Carbon Co. v. Binney Co*., 317 U.S. 228, 236 (1942); *Halliburton Energy Servs., Inc. v. M-I LLC*, 514 F.3d 1244, 1249 (Fed. Cir. 2008) ("Because claims delineate the patentee's right to exclude, the patent statute requires that the scope of the claims be sufficiently definite to inform the public of the bounds of the protected invention, *i.e.*, what subject matter is covered by the exclusive rights of the patent. Otherwise, competitors cannot avoid infringement, defeating the public notice function of patent claims.").

Under 35 U.S.C. § 112, "a patent is invalid for indefiniteness if its claims, read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention." *Nautilus, Inc. v. Biosig Instruments, Inc*., 572 U.S. 898, 901 (2014).

D. The Asserted Claims Are Invalid as Indefinite and/or Lacking Adequate Written Description or Enabling Disclosure

Based on Qomplx's Infringement Contentions, PANW identifies grounds upon which it contends that the Asserted Claims are invalid for failure to meet the requirements of 35 U.S.C. §§ 112 ¶¶ 1 and/or 2. Although its investigation continues, PANW believes that the Asserted Claims identified below are invalid for failure to comply with 35 U.S.C. § 112 for at least the following

reasons: (1) the specifications of the Asserted Patents lack a written description of the alleged inventions of the Asserted Claims as required by 35 U.S.C. § 112 ¶ 1; and/or (2) the specifications of the Asserted Patents do not enable the Asserted Claims as required by 35 U.S.C. § 112 ¶ 1; and/or (3) one or more of the Asserted Claims fail to specify the scope of the claimed inventions with reasonable particularity as required under 35 U.S.C. § 112 ¶ 2. PANW specifically reserves the right to amend and/or supplement these Invalidity Contentions based on a failure to comply with the requirements of 35 U.S.C. § 112, including, for example, based on supplementation or amendments to Qomplx's deficient Infringement Contentions.

The Asserted Claims fail to meet the written description requirement because the alleged inventions claimed therein are not sufficiently described in the specification of the Asserted Patents. Specifically, the Asserted Claims fail to meet the written description requirement for at least the terms identified below because, based on those terms, the specifications of the Asserted Patents do not adequately convey to those skilled in the art that the inventors had possession of the claimed subject matter as of the filing date. The Asserted Claims additionally fail to meet the enablement requirement because each specification of the Asserted Patents does not describe the manner and process of making and using the alleged inventions in the Asserted Claims to enable a person of skill in the art to make and use the full scope of the alleged invention without undue experimentation. Specifically, the Asserted Claims fail to meet the enablement requirement for at least the terms identified below because the specifications of the Asserted Patents do not describe the manner and process of making and using the alleged invention in the Asserted Claims. Based on PANW's present understanding of the Asserted Claims and Qomplx's apparent interpretation of these claims as reflected in its Infringement Contentions, the Asserted Claims may fail to satisfy the requirements of § 112, ¶ 2 because the precise scope of at least the phrases listed below (or

terms contained therein) cannot be determined with reasonable certainty by a POSITA when reading the claims in light of the specification and prosecution history.

Palo Alto Networks reserves the right to modify, amend, or supplement its contentions relating to 35 U.S.C. § 112 as the case progresses, including in view of any claim construction orders entered by the Court in this matter.

1. '663 Patent

The Asserted Claims fail the requirements of 35 U.S.C. § 112 for at least the following reasons:

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
1	"automatically install a virtual appliance software application"	Lack of Written Description; Lack of Enablement; Indefinite
1	"the virtual appliance software application configured to automatically load a plurality of stored configurations on the midserver"	Lack of Written Description; Lack of Enablement; Indefinite
1	"establish a secure network connection to an external network"	Indefinite
1	"retransmit the received data over the secure connection as a single data stream"	Lack of Written Description; Lack of Enablement; Indefinite

2. '424 and '425 Patents

The Asserted Claims fail the requirements of 35 U.S.C. § 112 for at least the following reasons:

a. ’424 Patent

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
1, 9	“data represents a respective portion of a distributed computational graph”	Lack of written description; Lack of enablement~~;~~ ~~Indefinite~~
1, 3, 5, 9, 11	“substantially in real time”	Lack of enablement; Indefinite
1, 4, 6, 9, 12	“portion of a distributed computational graph”	Lack of enablement~~;~~ ~~Indefinite~~
1, 4, 6	“information about the [second] transformation pipeline”	Lack of enablement~~;~~ ~~Indefinite~~
1	“process the respective first data… to determine information about the second transformation pipeline.”	Lack of written description; Lack of enablement~~;~~ ~~Indefinite~~
9	“process the respective first data to determine an identification of the second transformation pipeline.”	Lack of written description; Lack of enablement~~;~~ ~~Indefinite~~
1, 9	“distributed computing cluster”	Lack of written description; Lack of enablement
1	“cause a fourth computer system… to execute software instructions”	Lack of written description; Lack of enablement
1, 9	“process the first stream of input data substantially in real time by executing software instructions that apply the first transformation pipeline to the first stream of input data to generate first pipeline output messages”	Lack of written description; Lack of enablement
1	“process the first pipeline output messages substantially in real time by executing software instructions that apply the second transformation pipeline to the first pipeline output messages to generate second pipeline output messages”	Lack of written description; Lack of enablement
9	“process the first pipeline output messages substantially in real time by executing software instructions that apply the second transformation pipeline to the first pipeline output messages”	Lack of written description; Lack of enablement

[6] Includes asserted dependent claims not explicitly listed.

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
3	"process a second stream of input data received from a second input feed substantially in real time by executing software instructions that apply the first transformation pipeline to the second stream of input data"	Lack of written description; Lack of enablement
5	"process third pipeline output messages substantially in real time by executing software instructions that apply the second transformation pipeline to the third pipeline output messages"	Lack of written description; Lack of enablement
11	"process a second stream of input data received from a second input feed substantially in real time by executing software instructions that apply the first transformation pipeline to the second stream of input data"	Lack of written description; Lack of enablement
1, 9	"transmit… in accordance with the determined information."	Lack of written description; Lack of enablement~~, Indefinite~~
1	"configured to execute software instructions that cause a fourth computer system of the second plurality of computer systems to execute software instructions that apply at least one of the first transformation pipeline and the second transformation pipeline."	Lack of enablement~~, Indefinite~~
3, 4, 5, 6, 11, 12	"identify a … computer system"	Lack of enablement~~, Indefinite~~
~~3, 5~~	~~"wherein the fifth computer system is not configured to apply the … transformation pipeline…"~~	~~Indefinite~~
~~11~~	~~wherein the fourth computer system is not configured to apply the first transformation pipeline to any stream of input data~~	~~Indefinite~~
3	"cause the fifth computer system to process a second stream of input data received from a second input feed"	Lack of enablement~~, Indefinite~~
~~1, 7, 9, 13,~~	~~"distinct" with respect to two "computer systems"~~	~~Indefinite~~

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
1	"a third computer system of the first plurality of computer systems is configured to execute software instructions that cause a fourth computer system of the second plurality of computer systems to execute software instructions that apply at least one of the first transformation pipeline and the second transformation pipeline"	Lack of written description, Lack of enablement~~, Indefinite~~
9	"wherein a third computer system of the first plurality of computer systems is configured to execute software instructions that cause the second plurality of computer systems to execute at least one of the first transformation pipeline and the second transformation pipeline."	Lack of written description, Lack of enablement~~, Indefinite, Means-plus-function indefinite~~
1	"a first computer system of the first plurality of computer systems is configured to: receive a first stream of input data from a first input feed, process the first stream of input data substantially in real time by executing software instructions that apply the first transformation pipeline to the first stream of input data to generate first pipeline output messages, process the respective first data stored in the memory of the first computer system to determine information about the second transformation pipeline, and transmit the first pipeline output messages to a second computer system of the first plurality of computer systems in accordance with the determined information"	Lack of enablement~~, Means-plus-function indefinite~~
1	"the second computer system is configured to: receive the first pipeline output messages, and process the first pipeline output messages substantially in real time by executing software instructions	Lack of enablement~~, Means-plus-function indefinite~~

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
	that apply the second transformation pipeline to the first pipeline output messages to generate second pipeline output messages"	
1	"a third computer system of the first plurality of computer systems is configured to execute software instructions that cause a fourth computer system of the second plurality of computer systems to execute software instructions that apply at least one of the first transformation pipeline and the second transformation pipeline"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
3	"the third computer system of the first plurality of computer systems is configured to: identify a fifth computer system of the first plurality of computer systems… and cause the fifth computer system to process a second stream of input data received from a second input feed substantially in real time by executing software instructions that apply the first transformation pipeline to the second stream of input data"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
4	"the third computer system is further configured to execute software instructions that: identify one or more of the respective memories that store respective first data representing at least a respective portion of the distributed computational graph, wherein the respective first data includes information about the first transformation pipeline, and store in the identified one or more memories information identifying the fifth computer system"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
5	"the third computer system of the first plurality of computer systems is configured to: identify a fifth computer system of the first plurality of computer systems, wherein the	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
	fifth computer system is not configured to apply the second transformation pipeline to any pipeline output message, and cause the fifth computer system to process third pipeline output messages substantially in real time by executing software instructions that apply the second transformation pipeline to the third pipeline output messages"	
6	"the third computer system is further configured to execute software instructions that: identify one or more of the respective memories that store respective first data representing at least a respective portion of the distributed computational graph, wherein the respective first data includes information about the second transformation pipeline, and store in the identified one or more memories information identifying the fifth computer system"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
7	"the first computer system is further configured to transmit the first pipeline output messages to the fourth computer system of the second plurality of computer systems"	Lack of enablement~~, Means-plus-function indefinite~~
8	"the first computer system is further configured to transmit the first pipeline output messages to another computer system of the second plurality of computer systems"	Lack of enablement~~, Means-plus-function indefinite~~
9	"a first computer system of the first plurality of computer systems is configured to: receive a first stream of input data from a first input feed, process the first stream of input data substantially in real time by executing software instructions that apply the first transformation pipeline to the first stream	Lack of enablement~~, Means-plus-function indefinite~~

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
	of input data to generate first pipeline output messages, process the respective first data to determine an identification of the second transformation pipeline, and transmit the first pipeline output messages to a second computer system of the first plurality of computer systems in accordance with the determined identification"	
9	"he second computer system is configured to: receive the first pipeline output messages, and process the first pipeline output messages substantially in real time by executing software instructions that apply the second transformation pipeline to the first pipeline output messages"	Lack of enablement~~, Means-plus-function indefinite~~
11	"the third computer system of the first plurality of computer systems is configured to: identify a fourth computer system of the first plurality of computer systems, wherein the fourth computer system is not configured to apply the first transformation pipeline to any stream of input data, and cause the fourth computer system to process a second stream of input data received from a second input feed substantially in real time by executing software instructions that apply the first transformation pipeline to the second stream of input data"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
12	"he third computer system is further configured to execute software instructions that: identify one or more of the respective memories that stores respective data that describes a respective portion of a configuration of the first transformation pipeline,	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~

Asserted Claims[6]	Claim Term	Invalidity Based on 35 U.S.C. § 112
	and store in the identified one or more memories information identifying the fourth computer system"	
13	" the first computer system is further configured to transmit the first pipeline output messages to a fourth computer system of the second plurality of computer systems"	Lack of enablement~~, Means-plus-function indefinite~~
1, 3, 5, 9, 11	"software instructions that apply…"	Lack of enablement~~, Means-plus-function indefinite~~
1, 9	"software instructions that cause…"	Lack of enablement~~, Means-plus-function indefinite~~
4, 6, 12	"software instructions that identify… and store…"	Lack of enablement~~, Means-plus-function indefinite~~

b. '425 Patent

Asserted Claims[7]	Claim Term	Invalidity Based on 35 U.S.C. § 112
1	"first computer system of the first plurality of computer systems is configured to execute software instructions that: receive a first stream of data, process first portions of the first stream of data by applying a first transformation pipeline to the first portions of the first stream of data to generate first pipeline output messages, and transmit the first pipeline output messages"	Lack of written description; Lack of enablement~~; Means-plus-function indefinite~~
1	a second computer system of the first plurality of computer systems is configured to execute second software instructions that: process first portions of the first pipeline output messages by applying a second transformation pipeline to the first portions of the first pipeline output messages	Lack of written description; Lack of enablement~~; Means-plus-function indefinite~~

[7] Includes asserted dependent claims not explicitly listed.

Asserted Claims[7]	Claim Term	Invalidity Based on 35 U.S.C. § 112
1	"third computer system is configured to execute third software instructions that: process the first data to determine information about at least one of the first transformation pipeline and the second transformation pipeline, monitor at least a portion of the execution of at least one of the first software instructions and the second software instructions in accordance with the determined information, and in response to the monitoring, cause a fourth computer system of the third plurality of computer systems to execute fourth software instructions…"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
1	"fourth computer system of the third plurality of computer systems to execute fourth software instructions that perform at least one of: processing second portions of the first stream of data by applying the first transformation pipeline to the second portions of the first stream of data, or processing second portions of the first pipeline output messages by applying a second transformation pipeline to the second portions of the first pipeline output messages"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
1	"portion of the distributed computational graph describes a flow of data from the first transformation pipeline to an input of the second transformation pipeline"	Lack of written description; Lack of enablement~~; Means-plus-function indefinite; Indefinite~~
1	"first data represents a portion of a distributed computational graph"	Lack of written description; Lack of enablement~~; Means-plus-function indefinite; Indefinite~~
1	"process the first data to determine information about at least one of the first transformation pipeline and the second transformation pipeline"	Lack of written description; Lack of enablement

Asserted Claims[7]	Claim Term	Invalidity Based on 35 U.S.C. § 112
1	"monitor at least a portion of the execution of at least one of the first software instructions and the second software instructions in accordance with the determined information"	Lack of written description; Lack of enablement~~; Indefinite~~
1	"cause a fourth computer system of the third plurality of computer systems to execute fourth software instructions"	Lack of written description; Lack of enablement
4	"respective second data represents at least a respective portion of the distributed computational graph"	Lack of written description; Lack of enablement~~; Means-plus-function indefinite~~
4	"first software instructions transmit the first pipeline output messages according to the flow of the distributed computational graph"	Lack of written description; Lack of enablement~~; Means-plus-function indefinite~~
4	"transmit the first pipeline output messages according to the flow of the distributed computational graph"	Lack of written description; Lack of enablement~~; Indefinite~~
5	"third software instructions are further configured to monitor at least a portion of the execution of at least one of the first software instructions and the second software instructions, to identify a transformation pipeline that causes at least one of the first computer system and the second computer system to transmit information that is outside established data stability boundaries"	Lack of written description, Lack of enablement~~, Means-plus-function indefinite, Indefinite~~
5	"monitor at least a portion of the execution of at least one of the first software instructions and the second software instructions"	Lack of written description; Lack of enablement~~, Indefinite~~
5	"identify a transformation pipeline that causes at least one of the first computer system and the second computer system to transmit information that is outside established data stability boundaries"	Lack of written description; Lack of enablement~~, Indefinite~~

Asserted Claims[7]	Claim Term	Invalidity Based on 35 U.S.C. § 112
5	"outside established data stability boundaries"	Lack of written description; Lack of enablement~~, Indefinite~~
~~1, 3~~	~~"distinct" with respect to two "computer systems"~~	~~Indefinite~~
1	"software instructions that: receive…"	Lack of enablement~~; Means-plus-function indefinite~~
1	"software instructions that: process…"	Lack of enablement~~; Means-plus-function indefinite~~
1	"software instructions that: perform…"	Lack of enablement~~; Means-plus-function indefinite~~
4	"software instructions transmit…"	Lack of enablement~~; Means-plus-function indefinite~~
5	"software instructions are further configured to monitor…"	Lack of enablement~~; Means-plus-function indefinite~~

3. '627 Patent

The Asserted Claims fail the requirements of 35 U.S.C. § 112 for at least the following reasons:

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
1, 18	"the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises representations of a first plurality of edges"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1, 18	"the first plurality of entities comprises a plurality of accounts and a plurality of resources"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1, 18	"wherein each edge of the first plurality of edges corresponds to a respective relationship between a respective pair of entities"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1	"based on a first portion of the streaming data, identify a first entity that does not correspond to any of the first plurality of nodes, wherein the	Lack of Written Description; Lack of Enablement~~; Indefinite~~

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
	first entity is not of the first plurality of entities,"	
1	"based on a second portion of the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1	"modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a first node corresponding to the first entity and a representation of a first edge corresponding to the first relationship, wherein the first node is not of the first plurality of nodes and the first edge is not of the first plurality of edges,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1	"identify, based on the modified representation of the first graph, an attack path that could be involved in an attack involving the first entity,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1	"identifying a second entity that can be reached using the first entity, wherein the second entity corresponds to a second node, and the second node is related by one or more edges to the first node corresponding to the first entity in the modified representation of the first graph"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
1	"identifying a third entity that can be reached using the second entity, wherein the third entity corresponds to a third node, and the third node is related by one or more edges to the second node in the modified representation of the first graph"	Lack of Written Description; Lack of Enablement~~; Indefinite~~

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
5	"wherein the modified representation of the first graph comprises a representation of a node corresponding to the first entity, wherein the first entity is identified based on active reconnaissance results"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
6	"wherein the modified representation of the first graph comprises a representation of a node corresponding to the first entity, wherein the first entity is identified based on passive reconnaissance results"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
11	"determine whether an event, of the events relating to one or more entities of the first plurality of entities, is anomalous"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
11	"determining at least one behavior pattern associated with the first entity"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
11	"comparing the event to the at least one behavior pattern"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
12	"wherein comparing the event to the at least one behavior pattern comprises using a threshold"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
13	"determining that the first entity is associated with a resource"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
13	"wherein the resource is a sensitive resource"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
16	"wherein at least one entity of the first plurality of entities is at least one of a user, a place, a device, a resource, a group, or a service"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
17	"wherein the representation of the first graph comprises a representation of at least one node that does not correspond to an entity"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
18	"identify, based on the representation of the first graph, a first plurality of attack paths comprising a first entity of the first plurality of entities,	Lack of Written Description; Lack of Enablement~~; Indefinite~~

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
	wherein each attack path of the first plurality of attack paths targets a second entity of the first plurality of entities,"	
18	"receive streaming data comprising time-stamped data about events relating to one or more entities of the first plurality of entities,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
18	"based on a first portion of the streaming data, identify a third entity that does not correspond to any of the first plurality of nodes, wherein the third entity is not of the first plurality of entities,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
18	"based on a second portion of the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
18	"modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a node corresponding to the third entity and an edge corresponding to the first relationship, wherein the node is not of the first plurality of nodes and the edge is not of the first plurality of edges"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
18	"identify, based on the modified representation of the first graph, a second plurality of attack paths comprising the first entity, wherein each attack path of the second plurality of attack paths targets the second entity, and wherein an attack	Lack of Written Description; Lack of Enablement~~; Indefinite~~

Asserted Claims	Claim Term	Invalidity Based on 35 U.S.C. § 112
	path of the second plurality of attack paths comprises the third entity"	
20	"identifying a first plurality of event flows that could involve the first entity,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
20	"identifying a first plurality of dependencies associated with the first plurality of event flows"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
21	"based on a third portion of the streaming data, identify an anomalous event"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
21	"store in the hardware memory a dependency tree for the anomalous event"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
21	"based on the dependency tree, determine a point of origin for the anomalous event"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
21	"based on the dependency tree, determine a point of origin for the anomalous event"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
22	"wherein a portion of the representation of the first graph is derived from reconnaissance data received by the computer system from a plurality of computer systems,"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
22	"wherein a first computer system of the plurality of computer systems performs passive reconnaissance"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
22	"wherein a second computer system of the plurality of computer systems performs active reconnaissance"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
24	"analyze an attack path of the plurality of attack paths using a graph analysis algorithm"	Lack of Written Description; Lack of Enablement~~; Indefinite~~
28	"wherein the representation of the first graph comprises an identification of a security vulnerability associated with the third entity, and wherein an attack path of the second plurality of attack paths is based on the security vulnerability"	Lack of Written Description; Lack of Enablement~~; Indefinite~~

VII. INVALIDITY UNDER 35 U.S.C. § 101 (STANDING ORDER REGARDING SUBJECT MATTER ELIGIBILITY)

Pursuant to Judge Gilstrap's Standing Order, Exhibits E-H provide charts identifying (1) each exception to eligibility (e.g., abstract idea, law of nature, and natural phenomenon) to which each Asserted Claim of the Asserted Patents is directed and the factual and legal basis therefor; (2) whether one or more of the Asserted Claims of the Asserted Patents are representative of any other claims; (3) a description of the industry, at the relevant time, in which the Asserted Claims of the Asserted Patents are alleged to be well understood, routine, and conventional, and the factual and legal basis therefor; (4) a description of how each element of the Asserted Claims of the Asserted Patents, both individually and in combination with the other elements of that claim, was well understood, routine, conventional, in the relevant industry at the relevant time, and the legal and factual basis therefor; and (5) any other factual or legal basis for how the Asserted Claims of the Asserted Patents are otherwise ineligible for patent protection.

Exhibits E-H are based on the facts and law currently available to PANW, and PANW reserves the right to update these contentions in light of additional cases and guidance as they become available, the Court's claim construction order, or positions Qomplx takes on infringement.

VIII. DOCUMENT PRODUCTION

Concurrently herewith, PANW is producing the documents required under P.R. 3-3, P.R. 3-4, and this Court's Standing Order Regarding Subject Matter Eligibility.

Dated: ~~January 12~~September 1, 2026 */s/ Brian Biddinger*

Kevin P.B. Johnson
kevinjohnson@quinnemanuel.com
QUINN EMANUEL URQUHART &
SULLIVAN, LLP
555 Twin Dolphin Drive, 5th Floor
Redwood Shores, CA 94065
Telephone: (650) 801-5000

Brian Biddinger
brianbiddinger@quinnemanuel.com
QUINN EMANUEL URQUHART &
SULLIVAN, LLP
295 5th Avenue, 9th Floor
New York, New York 10016
Telephone: (212) 849-7000

Melissa R. Smith (Texas Bar No. 24001351)
melissa@gillamsmithlaw.com
GILLAM & SMITH, LLP
303 South Washington Avenue
Marshall, Texas 75670

Attorneys for Defendant Palo Alto Networks, Inc.

CERTIFICATE OF SERVICE

I hereby certify that, on ~~January 12~~September 1, 2026, all counsel of record who have appeared in this case are being served with a copy of the foregoing via email.

Dated: ~~January 12~~September 1, 2026 */s/ Benjamin M. Kleinman*