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

DAMAKA, INC., Plaintiff, v. CISCO SYSTEMS, INC., Defendant.	CASE NO. 2:25-cv-00593-RWS-RSP

DEFENDANT CISCO SYSTEMS, INC.'S
INVALIDITY AND SUBJECT MATTER ELIGIBILITY CONTENTIONS

I. INTRODUCTION

Pursuant to the Docket Control Order (ECF No. 41), Defendant Cisco Systems, Inc. ("Cisco") provides these Preliminary Invalidity Contentions to Plaintiff Damaka, Inc. ("Damaka") with respect to U.S. Patent Nos. 9,027,032 (the "'032 Patent"), 9,578,092 (the "'092 Patent"), 9,270,744 (the "'744 Patent"), 11,576,046 (the "'046 Patent"), 11,930,362 (the "'362 Patent"), 12,279,116 (the "'116 Patent") (collectively, the "Asserted Patents"). Plaintiff's November 13, 2025 Infringement Contentions assert the following claims of the Asserted Patents against Cisco:

- Claims 15-20 of the '032 Patent;
- Claims 1-5, 7-8, 10, 12-19, 22-25, and 27 of the '092 Patent;
- Claims 20-27 of the '744 Patent;
- Claims 1-4, 7-15, 17, and 19-20 of the '046 Patent;
- Claims 1-3, 5-10, 14-16, 18-25, 27-28, and 75 of the '362 Patent; and
- Claims 1, 9-13, 16-24, 26-29, 37-40, 43-51, and 53-54 of the '116 Patent.

(collectively, the "Asserted Claims").

With respect to each Asserted Claim and based on its investigation to date, Cisco (a) identifies each item of prior art that anticipates each asserted claim or renders it obvious; (b) specifies whether each such item of prior art anticipates each Asserted Claim or renders it obvious, and, if it renders it obvious, explain why the prior art renders the asserted claim obvious and identify combinations of prior art showing obviousness; (c) submits charts specifically identifying where and how in each item of prior art each limitation of each Asserted Claim is found; (d) identifies any limitations that are indefinite, lack written description, or fail to satisfy the enablement requirement under 35 U.S.C. § 112; and (e) identifies any claims that are directed to ineligible subject matter under 35 U.S.C. § 101.

Damaka has failed to provide infringement contentions that comply with the patent local rules or that put Cisco on notice of Damaka's infringement theories. P.L.R. 3-1 (requiring an identification of "specifically where each element of each asserted claim is found within each

Accused Instrumentality."); *see also Computer Acceleration Corp. v. Microsoft Corp.*, 503 F. Supp. 2d 819, 823 (E.D. Tex. 2007) ("The contentions must be specific enough to give a defendant notice of plaintiff's infringement claims"). Given Damaka's failure to identify the full scope of the accused instrumentalities and where Damaka contends that each element of the Asserted Claims is found within each instrumentality, Cisco lacks a complete understanding of Damaka's alleged infringement and/or claim construction positions. Accordingly, Cisco reserves the right to amend, supplement, and/or otherwise modify these Preliminary Invalidity Contentions if Damaka provides new infringement theories and/or claim construction theories inconsistent with its November 13, 2025 infringement contentions.

A. Claim Construction

The Court has not yet construed any of the terms in the Asserted Claims of the Asserted Patent. These Preliminary Invalidity Contentions are served in view of Cisco's current understanding of the Asserted Claims, without the benefit of claim construction proceedings in this action. Accordingly, Cisco does not take any position herein regarding the proper scope or construction of the Asserted Claims. To the extent that Cisco's contentions reflect, imply, or suggest a particular interpretation or construction of any claim element or term, Cisco does not adopt, advocate, or acquiesce to such an interpretation or construction. Cisco's contentions therefore should not be relied upon as a statement of Cisco's proposed claim constructions or as any admission regarding the proper scope of the claims. Nor do these Preliminary Invalidity Contentions constitute any admission by Cisco that any accused products or services, including any current or past versions of those products or services, are covered by any Asserted Claim.

These Preliminary Invalidity Contentions may consider Damaka's apparent interpretations of the Asserted Claims as reflected in Damaka's Infringement Contentions. Accordingly, any assertion herein that a particular limitation is disclosed by a prior art reference or references may be based in part on Damaka's apparent interpretation, but is not intended to be, and is not, an admission by Cisco that any such construction is supportable or correct, or that any claim terms of the Asserted Claims are not invalid under 35 U.S.C. § 112 for being indefinite,

failing to satisfy the written description requirement, or failing to satisfy the enablement requirement. Moreover, nothing in these Preliminary Invalidity Contentions should be construed as an admission or a waiver by Cisco of any particular construction of any claim term. Further, Cisco specifically denies that Damaka's apparent claim constructions are supportable or correct.

B. Ongoing Discovery and Disclosures

These Preliminary Invalidity Contentions are based on Cisco's current knowledge and understanding of the Asserted Claims and review of prior art items as of the date of service, and are made without the benefit of discovery regarding the parties' claim construction contentions, any expert discovery, any third-party discovery, and any claim construction opinion or order by the Court. What's more, Damaka has failed to serve Infringement Contentions that comply with the Patent Local Rules, which prejudices Cisco's ability to adequately prepare its defenses. Accordingly, these Preliminary Invalidity Contentions are provided without prejudice to Cisco's right to revise, amend, correct, supplement, modify, or clarify the same. Cisco also reserves the right to complete their investigation and discovery of the facts, to produce subsequently discovered information, and to introduce such subsequently discovered information at the time of any hearing or trial in this action.

These Preliminary Invalidity Contentions are also based at least in part on Cisco's current understanding of the Asserted Claims in view of Damaka's Infringement Contentions, which fail to provide Cisco with requisite notice of Damaka's infringement theories under the Patent Local Rules. If Damaka amends its Infringement Contentions to address any deficiency therein, or for any other reason, Cisco reserves the right to amend or supplement these Preliminary Invalidity Contentions.

Cisco further reserves the right to supplement and amend these Preliminary Invalidity Contentions and its accompanying document production based on further investigation, analysis, and discovery, Cisco's consultation with experts and others, and contentions or court rulings on relevant issues such as claim construction and priority dates. For example, since fact discovery has not yet begun, deposing the alleged inventors may reveal information that affects the

disclosures and contentions herein. In addition, Cisco has not completed discovery from third parties who have information concerning the prior art cited herein and possible additional art, including additional evidence regarding prior art system disclosed herein.

Because Cisco is continuing its search for and analysis of relevant prior art and seeking discovery from third parties with information concerning the prior art cited herein and possible additional art, Cisco reserves the right to revise, amend, and/or supplement the information provided herein, including identifying, charting, and/or relying upon additional prior art references, relevant disclosures, and bases for these Preliminary Invalidity Contentions. Additional prior art, disclosures, and invalidity grounds, whether or not cited in this disclosure and whether known or not known to Cisco, may become relevant as investigation, analysis, and discovery continue, and following claim construction proceedings in this case.

These Preliminary Invalidity Contentions also incorporate by reference all bases for invalidity identified in subsequent pleadings and discovery responses in this case, and any bases of invalidity identified during the prosecution of the Asserted Patents, as well as all related patents and/or patent applications. Cisco further incorporates by reference all admissions regarding the Asserted Patent including, but not limited to, admissions in the specification of the Asserted Patents and the prosecution of the Asserted Patents and related patents and/or patent applications.

These Preliminary Invalidity Contentions further incorporate by reference (1) any and all prior art identified in documents produced by Damaka to Cisco in this case; (2) any and all materials regarding invalidity that should have been produced to Cisco but have not been produced to date, to the extent that any exist; (3) any prior art of which a named inventor of the Asserted Patents is aware and/or on which he, she, and/or Damaka contends the alleged invention(s) of the Asserted Patents builds upon or improves; and (4) any and all admissions by Damaka and/or a named inventor regarding the Asserted Patents including, but not limited to, admissions in the specifications of the Asserted Patents and the prosecution of the Asserted Patents and related patents and/or patent applications. In addition, these Preliminary Invalidity

Contentions incorporate all prior art identified and invalidity contentions served in any related litigation, including any petitions for *inter partes* review of the Asserted Patents.

II. PRIORITY DATE OF THE ASSERTED PATENT AND CLAIMS

Damaka asserts that the Asserted Patents are "entitled to at least the filing date of [their] earliest application, U.S. Provisional Patent Application No. 61/846,958, filed July 16, 2013." Damaka's 2025-11-13 Infringement Contentions. Damaka has the burden to show entitlement to the asserted priority date that is earlier than the filing date of the Asserted Patent. Damaka has not met that burden and is not entitled to that alleged priority date. The claims of the Asserted Patents are not supported by U.S. Provisional Patent Application No. 61/846,958 and are not entitled to the alleged priority date, and the alleged invention of the Asserted Patents was not disclosed in U.S. Provisional Patent Application No. 61/846,958. Cisco reserves the right to modify, amend, or supplement its Preliminary Invalidity Contentions with additional prior art references if any Asserted Claim is shown to not be entitled to the alleged priority date or if Damaka alleges any other priority date for any of the Asserted Claims.

III. PRIOR ART IDENTIFICATION AND INVALIDITY CLAIM CHARTS

The accompanying invalidity claim charts (identified in Section IV below) cite to particular teachings and disclosures of the prior art references as applied to elements of the Asserted Claims. Persons having ordinary skill in the art, however, may view an item of prior art generally in the context of other publications, literature, products, and understanding. Accordingly, the cited portions are only exemplary and are intended to put Damaka on notice of the basis for Cisco's contentions. Cisco has endeavored to identify relevant portions of the references, but the references may contain additional support for particular claim limitations. Cisco reserves the right to rely on uncited portions of the prior art references, other documents, and/or operational systems, as well as fact and expert testimony, to provide context or to aid in understanding the cited portions of the references and interpreting the teachings of the prior art and to establish bases for combinations of certain cited references that render the Asserted Claims obvious. Cisco also reserves the right to rely on any prior art system referenced,

embodied, or described in any of the prior art references identified herein, or which embodies any of the prior art references identified herein.

Moreover, Cisco reserves the right to rely on inventor admissions concerning the scope of the prior art relevant to the Asserted Patents found in, *inter alia*, the prosecution history of the Asserted Patents or related patents and/or patent applications, any testimony or declarations of the named inventors concerning the Asserted Patents or related patents, and any papers or evidence submitted by Damaka in connection with this litigation, any other pending or future litigation brought by Damaka involving the Asserted Patents or related patents, any future *ex parte* reexamination proceedings involving the Asserted Patents or related patents, or any post grant proceedings (e.g., *inter partes* review) involving the Asserted Patents or related patents. Cisco also may establish what was known to a person having ordinary skill in the art through treatises, published industry standards, other publications, products, and/or testimony.

Where the invalidity claim charts (identified in Section IV below) cite to a particular figure in a reference, the citation should be understood to encompass the caption of the figure and other text relating to and/or describing the figure. Similarly, where the invalidity claim charts cite to particular text referring to a figure, the citation should be understood to include the figure and related figures as well.

The prior art references listed herein and in the accompanying invalidity claim charts may disclose the elements of the Asserted Claims explicitly and/or inherently. The prior art references are also relevant for their showing of the state of the art and reasons and motivations for making improvements, additions, and combinations. The suggested obviousness combinations are provided in the alternative to Cisco's anticipation contentions and are not to be construed to suggest that any reference is not itself anticipatory.

Further, the combinations of prior art references contained herein demonstrating the obviousness of the Asserted Patents under 35 U.S.C. § 103 are merely exemplary and are not intended to be exhaustive. All such combinations are intended to include and be in view of the knowledge of a person of ordinary skill in the art. Additional obviousness combinations of the

identified prior art references are possible, and Cisco reserves the right to use any such combination(s) in this litigation. In particular, Cisco is currently unaware of the extent to which Damaka may contend that limitations of any particular claim(s) are not disclosed in the art that Cisco has identified as anticipatory. To the extent that Damaka does so, Cisco reserves the right to identify other evidence or references that anticipate or render obvious the particular claim(s).

Nothing in these contentions should be treated as an admission that any of Cisco's accused instrumentalities meet any limitation of the Asserted Claims. Cisco denies infringement of the Asserted Claims. To the extent that any prior art references identified by Damaka contains a claim element that is the same as or similar to an element in an accused instrumentality, based on a claim construction inferred from Damaka's Infringement Contentions, inclusion of that reference in these Preliminary Invalidity Contentions is not a waiver by Cisco of any claim construction or non-infringement position, nor is it an admission or suggestion by Cisco that any accused instrumentality satisfies the limitations of the Asserted Claims under a proper construction of those claims.

IV. PRELIMINARY INVALIDITY CONTENTIONS FOR THE ASSERTED PATENTS

A. Prior Art References[1]

1. Prior Art Patents and Publications

Cisco identifies the following prior art patents and printed publications, which anticipate the Asserted Claims under 35 U.S.C. §§ 102(a), (b), (e), and/or (g), and/or render the Asserted Claims obvious under 35 U.S.C. § 103, either alone or in combination.

Prior Art	Date of Issue / Publication	Effective Filing Date
U.S. Pub. No. 2014/0184723 ("Morrison")	Jul. 3, 2014	Dec. 31, 2012

[1] To the extent one or more prior art patents, publications, or systems are identified in the claim charts attached hereto but are not included in the tables and lists below, those prior art patents, publications, or systems are also prior art to the Asserted Patents.

Prior Art	Date of Issue / Publication	Effective Filing Date
U.S. Pub. No. 2012/0019610 ("Hornyak")	Jan. 26, 2012	Apr. 28, 2010
U.S. 7,987,233 ("Osborne")	Jul. 26, 2011	Jan. 21, 1998
U.S. Pub. No. 2011/0249075 ("Abuan")	Oct. 13, 2011	Apr. 2, 2007
U.S. Pub. No. 2010/0142542 ("Van Wie")	Jun. 10, 2010	Dec. 5, 2008
U.S. Pub. No. 2013/0019297 ("Lawson")	Jan. 17, 2013	May 23, 2011
U.S. Pub. No. 2014/0063344 ("Dureau")	Mar. 6, 2014	Aug. 29, 2012
U.S. Pub. No. 2012/0092438 ("Guzman")	Apr. 19, 2012	Oct. 18, 2010
U.S. Pub. No. 2011/0249077 ("Abuan 077")	Oct. 13, 2011	Apr. 7, 2010
U.S. Pub. No. 2011/0196969 ("Tarte")	Aug. 11, 2011	Feb. 8, 2010
U.S. 10,182,085 ("Mayer")	Jan. 15, 2019	Apr. 27, 2011
U.S. 9,071,681 ("Vasquez")	Jun. 30, 2015	Mar. 19, 2012
U.S. Pub. No. 2009/0193345 ("Wensley")	Jul. 30, 2009	Jan. 28, 2008
U.S. Pub No. 2012/0311686 ("Medina")	Dec. 6, 2012	Jun. 3, 2011
U.S. Pub No. 2015/0358448 ("Jeong")	Dec. 10, 2015	Jan. 8, 2013
U.S. 5,689,641 ("Ludwig")	Nov. 18, 1997	Oct. 1, 1993
U.S. 8,396,490 ("Platt")	Mar. 12, 2013	Dec. 17, 2009
U.S. 2013/0290982 ("Beilis")	Oct. 31, 2013	Apr. 30, 2012

Prior Art	Date of Issue / Publication	Effective Filing Date
U.S. Pub. No. 2010/0066807 ("Eisenberg")	Mar. 18, 2010	Dec. 1, 2000

2. Prior Art Systems and Products

Cisco also identifies the following prior art products or systems, which invalidate the Asserted Claims under 35 U.S.C. §§ 102(a), (b), and/or (g), and/or render the Asserted Claims obvious under 35 U.S.C. § 103, either alone or in combination.

Prior Art	Short Name	Date of Use, Sale, Offer for Sale, or Invention
WebRTC	WebRTC	At least as early as 2011
Jabber and Jabber Web SDK	Jabber	At least as early as 2011
Cisco Client Services Framework (CSF) and Unified Communications Manager (CUCM)	Cisco CSF	At least as early as 2007
TokBox OpenTok System	TokBox	At least as early as 2010
IBM Sametime	IBM	At least as early as 1998
Twilio Client SDK	Twilio	At least as early as 2012
VidyoMobile/ VidyoWorks / VidyoClient SDK and Vidyo Technology SDK (including in combination with the Vidyo.io Platform)	Vidyo	At least as early as 2011
Microsoft Lync	Lync	At least as early as 2010
RADVISION (Softil) BEEHD Client Framework and Scopia Mobile	Radvision	At least as early as 2010
QuickBlox Video Chat SDK	QuickBlox	At least as early as 2009
ooVoo Mobile for iOS and Android	ooVoo	At least as early as 2010
AddLive Developer Platform, APIs, and SDKs	AddLive	At least as early as 2012

Prior Art	Short Name	Date of Use, Sale, Offer for Sale, or Invention
Skype Mobile	Skype	At least as early as 2010
Apple FaceTime	FaceTime	At least as early as 2010
Fring, including with Symbian S60 Fring, Windows Mobile Fring, VoIP, iOS, and video support	Fring	At least as early as 2007
Qik / T-Mobile Video Chat	Qik	At least as early as 2010
Tango for iOs and Android	Tango	At least as early as 2010
Google Talk Video Chat	Google Talk	At least as early as 2011
Google+ Hangouts (mobile)	Google Hangouts	At least as early as 2011
Yahoo Messenger (mobile)	Yahoo	At least as early as 2010
Polycom RealPresence Mobile	Polycom	At least as early as 2011

Discovery is not yet complete and Cisco continues to investigate these and other prior art products and systems and thus may uncover additional documentation and evidence regarding their operation and functionality.[2] Cisco may also rely on physical samples, executable software, or source code as evidence of the relevant functionality of these prior art products and systems. Cisco will make available for inspection any additional documentation and evidence and 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.

3. The Accused Functionality

Damaka's January 7, 2025 Infringement Contentions accuse the following functionality of infringement: "all versions and variants of the Webex Application, including, but not limited to, Webex for iOS, Webex for Android, Webex for Web Browser, Webex for the Windows

[2] Cisco will serve third party subpoenas with discovery requests related to these prior art products and systems and intends to amend or supplement these Preliminary Invalidity Contentions after it receives additional information and evidence related to these prior art products and systems.

operating system, Webex for the Mac operating system, and Webex for the Linux operating system" and "Cisco's Webex SDK for iOS, Android and Web." Cisco denies that Webex, Webex SDKs, or any other Cisco products, services, or functionalities infringe any valid or enforceable claim of the Asserted Patents. Further, Cisco's Webex pre-dates the earliest priority date of the Asserted Patents. Cisco Webex was also in commercial use at least a year before either the effective filing date of the alleged invention in the Asserted Patents or the date on which the alleged invention in the Asserted Patents was purportedly disclosed to the public. *See* 35 U.S.C. § 273.

Webex offered real-time online meeting capabilities since the 1990s. *See, e.g.*, *WebEx Introduces Industry's First Collaboration Application Server for E-Commerce, Customer Care and Conferencing*, WebEx (1998), *available at* https://web.archive.org/web/20040817212040/http://www.webex.com/pr/pr01.html; https://web.archive.org/web/20040618061239/http://www.webex.com/webex/press-webex-archive.html; https://web.archive.org/web/19970621232021/http://www.webex.com/; https://web.archive.org/web/19990508084427/http://webex.com/webex/site/default.asp; https://web.archive.org/web/20130216230441/http://www.webex.com/webex/site/faq.asp?Rnd=0.590392601323453; *WebEx Communications, Inc.*, Cisco (2016), *available at* https://www.encyclopedia.com/books/politics-and-business-magazines/webex-communications-inc. WebEx Communications was founded in 1995 and was "the world's leading provider of online meeting services." Through its web site, clients could "host real-time business meetings, conferences, and training seminars in which multimedia presentations and software programs can be simultaneously viewed and/or controlled by participants."

By the late 1990s, WebEx offered participants access to information in a variety of formats, including "presentation sharing, screen synchronization, form completion, and other features." It also offered "a free service that allowed up to six Internet-connected customers to meet in real time to share presentations, documents, and software applications." Prior to 2000,

"[c]ustomized versions of the software were also available, and some companies licensed it for incorporation into their own programs."

WebEx also offered A/V conferencing services on mobile devices at least as early as 2004 and could integrate with publicly available chat and instant messaging platforms, including AIM Web Meetings. *See id.*; *see also WebEx Ties Web Conferencing with IP Phones*, eWeek, April 1, 2004, *available at* https://www.eweek.com/enterprise-apps/webex-ties-web-conferencing-with-ip-phones/; *WebEx Provides Advanced Web Meeting Capabilities to AOL Instant Messenger™*, WebEx (2004), *available at* https://web.archive.org/web/20040916083022/http://www.webex.com/pr/pr296.html; Siemens Integrates WebEx into OpenScape™ Enterprise Communications Solution, WebEx (2004), *available at* https://web.archive.org/web/20050830220551/http://www.webex.com/pr/pr288.html. By the early 2000s, WebEx had announced an alliance with cell phone maker Siemens to bring conferencing services to mobile phones via that firm's OpenScape program, and integrated its software into the online customer service offerings of Salesforce.com. *WebEx Ties Web Conferencing with IP Phones*, eWeek, April 1, 2004; *WebEx Provides Advanced Web Meeting Capabilities to AOL Instant Messenger™*, WebEx (2004), available at https://web.archive.org/web/20040916083022/http://www.webex.com/pr/pr296.html; *Siemens Integrates WebEx into OpenScape™ Enterprise Communications Solution*, WebEx (2004), available at https://web.archive.org/web/20050830220551/http://www.webex.com/pr/pr288.html. By this time, WebEx offered "seamless integration between telephone handsets, IP and traditional telephony conferencing systems, and Web meetings. *WebEx Network Enhanced to Enable Instant Web Meetings From Telephone Keypad*, WebEx (2003), available at https://web.archive.org/web/20050210233130/http://webex.com/pr/pr234.html.

WebEx also offered a Developer Program with access to a Web Collaboration Platform Toolkit at least as early as 2005. *See, e.g., WebEx Opens MediaTone Platform to Third Party Developers*, WebEx (2005), *available at*

https://web.archive.org/web/20060831175800/http://www.webex.com/pr/pr332.html. The Program offered "the integration of real-time collaboration capabilities into third party applications" and offered "the tools, resources and support developers need to integrate web meeting capabilities into enterprise applications, VoIP platforms and telephony offerings or develop new, customized applications to meet specific industry needs." It also offered integration with Jabber, including by using the platform's APIs "to add one click WebEx meetings to the Jabber Messenger client." *Id.*; *see also Integration Tools to Accelerate Your Business*, WebEx (2005), *available at* https://web.archive.org/web/20060811161952/http://www.webex.com/webex/developer-overview.html.

Cisco's 2007 acquisition of Webex brought feature-complete A/V conferencing platform that "extend[ed] Cisco's vision for Unified Communications." *Cisco Announces Agreement to Acquire WebEx*, Cisco (2007), *available at* https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2007/m03/cisco-announces-agreement-to-acquire-webex.html. The core Webex meetings functionality was already offered by Cisco and well-established before the earliest priority date of Damaka's patents. *See, e.g., Cisco WebEx Meeting Center Product Overview*, Cisco (2013), *available at* https://www.webex.com/content/dam/webex/eopi/Americas/USA/en_us/documents/pdf/webex_meeting_center_datasheet.pdf (discussing web browser, mobile phone, or tablet device meeting capabilities, document, application, and desktop sharing, integrated video, integrated audio, enterprise integration, and cross-platform support).

Cisco Webex mobile applications and mobile A/V functionality were also publicly available well before Damaka's claimed priority date. *See id.*; *see also Cisco WebEx Meetings for iPad and iPhone Quickstart Guide v4*, Cisco (2013), *available at* https://voipinfo.net/docs/cisco/Cisco_WebEx_Meetings_iPad_iPhone_quickstart_v4_0.pdf; ; *Scheduling and Launching Meetings on Mobile Devices Using WebEx APIs*, Cisco (2011), *available at*

https://web.archive.org/web/20130420213506/http://developer.cisco.com/documents/4733862/69a56c79-8e45-4522-88d0-4f90334cd283. This included Meetings mobile applications and support for core Webex functionality (including A/V conferencing, document sharing, integrated content viewing, Enterprise integration, chat capabilities, and embedded video windows). (*See Cisco WebEx Meetings for iPad and iPhone Quickstart Guide v4*, Cisco (2013), *available at* https://voipinfo.net/docs/cisco/Cisco_WebEx_Meetings_iPad_iPhone_quickstart_v4_0.pdf; *see also Cisco Unified Communications System 9.0 SRND*, Cisco (2013), *available at* https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/srnd/9x/uc9x/confernc.html?; *Scheduling and Launching Meetings on Mobile Devices Using WebEx APIs*, Cisco (2011), *available at* https://web.archive.org/web/20130420213506/http://developer.cisco.com/documents/4733862/69a56c79-8e45-4522-88d0-4f90334cd283.)

Cisco also offered Webex developer integrations before the earliest priority date of Damaka's patents. Damaka accuses Webex SDK/API functionality and the claimed "superblock" and "function block" around meeting APIs, A/V options, and embedded meetings. Public developer and community materials show that WebEx XML and URL APIs were available well before the earliest priority date. Cisco's XML API allowed external applications to programmatically schedule and manage WebEx meetings and was in use by third party developers at least as early as 2009. *See WebEx Web Conferencing Developer*, Cisco (2013), *available at* https://web.archive.org/web/20130423173243/https://developer.cisco.com/web/webex-meetings-developer/home; *see also* https://community.cisco.com/t5/collaboration-knowledge-base/responses-on-createmeeting-requests-via-xml-api/ta-p/3617285?; https://community.cisco.com/t5/collaboration-knowledge-base/xml-api-russian-language-support/ta-p/3621313. Webex also offered the accused API calling and meeting URL-based integration functionality prior to 2013. *See, e.g.*, *WebEx Web Conferencing URL API FAQ*, Cisco, (2012) *available at*

https://web.archive.org/web/20121002114710/http://developer.cisco.com/web/webex-developer/url-api-faq; *WebEx Web Conferencing XML API FAQ*, Cisco (2012), *available at* https://web.archive.org/web/20120828201914/http://developer.cisco.com/web/webex-developer/xml-api-faq; WebEx Web Conferencing TSP API FAQ, Cisco (2013), *available at* https://web.archive.org/web/20130420095113/http://developer.cisco.com/web/webex-developer/tsp-api-faq; *WebEx Single-Sign On FAQ*, Cisco (2012), *available at* https://web.archive.org/web/20130420095306/http://developer.cisco.com/web/webex-developer/sso-faq; *General WebEx Web Conferencing API Service*, Cisco (2012), *available at* https://web.archive.org/web/20120730220755/http://developer.cisco.com/web/webex-developer/general-api-service.

Cisco has numerous patents and publications that pre-date the earliest priority date of the Asserted Patents and that either anticipate or render obvious the asserted claims. Cisco identifies examples of these references in the chart below.

Reference	Date of Issue / Publication	Effective Filing Date
US 6,167,432	Dec 26, 2000	Feb 29, 1996
US 6,789,119	Sep 7, 2004	Nov 24, 1999
US 6,654,032	Nov 25, 2003	Dec 23, 1999
US 6,763,501	Jul 13, 2004	Jun 9, 2000
US 6,901,448	May 31, 2005	Dec 29, 2000
US 7,069,298	Jun 27, 2006	Dec 29, 2000
US 6,925,645	Aug 2, 2005	Dec 29, 2000
US 6,567,813	May 20, 2003	Dec 29, 2000
US 7,130,883	Oct 31, 2006	Dec 29, 2000
US 7,213,051	May 1, 2007	Mar 28, 2002
US 7,216,172	May 8, 2007	Sep 25, 2001

Reference	Date of Issue / Publication	Effective Filing Date
US 7,203,755	Apr 10, 2007	Dec 29, 2000
US 7,814,150	Oct 12, 2010	Jun 3, 2003
US 7,636,754	Dec 22, 2009	Mar 21, 2002
US 7,672,996	Mar 2, 2010	Jan 25, 2007
US 8,473,850	Jun 25, 2013	May 24, 2007
US 2007/0005699	Jan 4, 2007	Jun 29, 2005
US 8,755,342	Jun 17, 2014	Oct 5, 2011
US 7,043,677	May 9, 2006	Jul 19, 2001
US 2008/0250110	Oct 9, 2008	Oct 7, 2002
US 8,312,081	Nov 13, 2012	Jun 29, 2005

After Damaka served its Infringement Contentions, Cisco put Damaka on notice that Webex pre-dates the earliest priority date of the Asserted Patents. *See* 2026-01-15 Nash Ltr to Fabricant re *Damaka, Inc. v. Cisco Systems, Inc.*, Case No. 2:25-cv-00593-RWS-RSP (E.D. Tex). Damaka has since failed to provide any reasonable basis for maintaining that Webex (or any other Cisco products) infringes the Asserted Claims even though Webex pre-dates the earliest claimed priority date of the Asserted Patents. Cisco denies that Webex or any other Cisco products, services, or functionalities infringe any valid or enforceable claim of the Asserted Patents. By maintaining that Webex infringes, Damaka is admitting that the same Webex functionality that predates the Asserted Patents would render the Asserted Claims invalid as anticipated. *Upsher-Smith Labs., Inc. v. Pamlab, L.L.C.*, 412 F.3d 1319, 1322 (Fed. Cir. 2005) ("A century-old axiom of patent law holds that a product which would literally infringe if later in time anticipates if earlier.") (citation omitted); accord *Peters v. Active Mfg. Co.*, 129 U.S. 530, 537 (1889) ("That which infringes, if later, would anticipate if earlier."); *see also Vanmoor v. Wal-Mart Stores, Inc.*, 201 F.3d 1363, 1366 (Fed. Cir. 2000) (affirming invalidity of patent

based on the patentee's own infringement allegations because accused product was placed on-sale before critical date of the invention).

4. State of the Art and Admitted Prior Art

The Asserted Patents themselves acknowledge that many aspects of the alleged invention were already known in the art. The Asserted Patents are directed to "improving functionality in software applications" by reworking an existing application (a so-called "superblock" application) to add "desired functionality" by incorporating additional executable instructions that "provide such additional functionality" such as audio/video calling, instant messaging, and collaboration via an API) while remaining within the same application context. '032 patent, 2:55-3:3. The claimed invention does little more than characterize the added instructions as a "function block" that is added to a "superblock." *Id*., 3:47-56. The specification itself, however, acknowledges that the core problems the Asserted Patents purport to solve—and the building blocks relied on—were already well known in the art.

First, the Asserted Patents admit that mobile operating systems already "provide[d] a separate memory space for each application that is being run and so must keep track of the memory boundaries for each application," and that switching between applications (switching contexts) was a known behavior that consumed resources and affected performance and battery life. *Id*., 2:31-39. The Asserted Patents further acknowledge that such resource and performance issues were less problematic relative to desktops and laptops "due to the greater amount of resources available for such devices compared to mobile devices." *Id*., 2:41-43.

The Asserted Patents further acknowledge that using a separate application to provide missing functionality (such as "audio, video, email, and/or file transfer") was already standard for mobile devices. The specification describes a conventional scenario where a "superblock" application needs email functionality: the user taps an email link, the system launches a separate email application (or switches to one already running), the user composes and sends the email, and then must switch back to the original application. *Id*., 3:4-18. The Asserted Patents characterize this as a "common occurrence for mobile device users," and expressly recognize

that this process is "disruptive, difficult to manage, and pulls the user away from the Superblock application while the provider of the superblock application may want the user to continue to stay in their application." *Id*. 3:19-22.

The Asserted Patents identify similar admitted-prior-art examples for video calling and virtual meetings. They describe a known scenario in which a user on a mobile device conducts a video call in one application while needing to view data in another application. *Id*. 3:30-39. According to the Asserted Patents, the user "has to conduct the call while switching back and forth between the video window of the call and the superblock application," which is "extremely disruptive to the user and limits the benefits of the video call, which include the ability to interact with some one on a visual basis rather than just an audio basis." *Id*., 3:37-39. Likewise, Asserted Patents explain that a worker on a shop floor who needs to join a virtual meeting must exit the current application, locate or enter meeting information in a separate meeting application, and then join the meeting there—again characterized as "extremely disruptive," time-wasting, and productivity-reducing. *Id*., 3:40-46.

As the basis for the claimed "function block," the Asserted Patents merely incorporate prior art endpoint technology and standard communication protocols. The specification states that the function block "may contain functionality similar to that of an endpoint as described in detail in U.S. Pat. No. 7,656,870," filed nearly a decade before the earliest priority date of the Asserted Patents, and that this prior-art endpoint's graphical user interface can simply be replaced with an API while some unneeded functionality is removed or additional functionality (such as mobile device management) is added. *Id*., 16:52-67. The '870 patent explains that "the basic functionality of the described endpoint may exist as logic embodied in the instruction set of the function block." *Id*., 16:65-67.

The '870 patent provides a C++ header for "AmadeoBlock," which exposes, through an API, a broad set of already-known communication and collaboration functions: presence, contact management, in app video, instant messaging, IM conferences, file transfer, audio and video call setup and control (start, accept, hold, mute, transfer, forward, voicemail, merge, DTMF),

voicemail management, sharing, whiteboard, logging, background operation, and network monitoring. *Id*., 6:25-7:47. These are all conventional features of real-time communication endpoints and collaboration clients, built on top of standard telephony and messaging protocols.

The '870 patent further acknowledges that the function block uses standard protocols such as "SIP, HTTP, HTTPS, STUN, RTP, SRTP, and ICMP" and that the device's operating system provides underlying functionality like network access and media rendering. *Id*., 17:45-52. The function block architecture relies on familiar concepts and using tables to track which capabilities are available. *Id*., 12:1-38.

The net effect of these admissions is that the Asserted Patents do not purport to have invented mobile audio/video calling and related functionality, SDKs, collaboration endpoints, or the underlying protocols. Those were already well-developed and even embodied in the applicants' own earlier endpoint patent that is incorporate by reference. Nor do the Asserted Patents claim to have discovered the problems related to switching context, resource usage, or user disruption; it explicitly describes those as existing, "common" behaviors on then-current mobile platforms. Instead, the specification merely presents the labels of "superblock" / "function" as a way to package the already-known concept of reworking an existing application to add "desired functionality" by incorporating additional executable instructions that "provide such additional functionality." *Id*., 3:47-56.

Many prior art systems were in existence long-before the earliest claimed priority date of the Asserted Patents that solved these same purported problems. As just one example, Web RTC— an open-source technology and set of IETF/W3C standards enabling real time audio, video, and data communication directly in web browsers and mobile applications—was designed to solve precisely the problem that Damaka describes in the Asserted Patents. As the official FAQ explains, WebRTC "includes the fundamental building blocks for high-quality communications on the web, such as network, audio and video components used in voice and video chat applications." *See* WebRTC FAQ, *available at* https://webrtc.googlesource.com/src/%2B/main/docs/faq.md. That is, WebRTC is itself a

"function block" that web and mobile applications invoke via APIs to provide embedded A/V calling inside the host page or app. WebRTC was publicly available and actively standardized at least as early as 2011. In May 2011, for example, Google publicly announced the WebRTC project and made an initial open-source WebRTC codebase available at code.webrtc.org, explicitly stating that it would "allow developers to create voice and video chat applications via simple HTML and JavaScript APIs." *Google Building Skype-like App in the Browser*, TMCnet (2011), *available at* https://blog.tmcnet.com/blog/tom-keating/google/google-building-skype-like-app-in-the-browser.asp.

Cisco also already offered multiple prior art systems and functionalities that solved the purported problems identified in the Asserted Patents, including Jabber, Jabber SDKs, Cisco Unified Client Services Framework (CSF), Cisco's Unified Communications Manager (CUCM), and (as discussed above) Webex. Numerous non-Cisco vendors also already offered platforms and functionality to embed A/V calling functionality into websites and applications, including via SDKs and HTML/JavaScript embeds. These included, for example, TokBox / OpenTok, Vidyo Technology SDK, IBM Sametime 8.5, Twilio, Skype, FaceTime, Google Talk, Google Hangouts, Yahoo Messenger, Fring, T-Mobile Video Chat, Tango, ooVoo Mobile, and RADVISION. These, and other prior art systems, were implemented into commercial products long before the earliest claimed priority date of the Asserted Patents and addressed the purported problems identified in the Asserted Patents.

B. Identification of Anticipation and Obviousness Combinations

Pursuant to P. R. 3–3(b) and (c), Cisco provides the following exemplary charts identifying examples of prior art and prior art combinations that anticipate and/or render obvious each of the asserted claims of the Asserted Patents. Cisco's investigation, analysis, and review of these references is ongoing. Cisco will provide updated claim charts as necessary and as they are available.

Exhibit	Invalidity Chart Reference(s)
1	**Morrison**
2	**Hornyak**
3	**Jabber**
4	**CSF**
5	**WebRTC**
6	**Abuan**
7	**Radvision**
8	**Van Wie**
9	**Jeong**
10	**Tarte**
11	**Guzman**
12	**Ludwig**
13	**TokBox**
14	**Vidyo**
15	**Lawson**
16	**Abuan 077**
17	**Platt**
18	**Beilis**
19	**Eisenberg**

In the Exhibits attached hereto, where Cisco cites to a particular figure in a prior art reference, the citation should be understood to encompass the caption and description of the figure and any text relating to the figure in addition to the figure itself. Conversely, where a cited portion of text refers to a figure, the citation should be understood to include the figure as

well. Additional evidence regarding the features and elements of the prior art reference may be provided by witness testimony, or by additional documents that describe the prior art reference that are discovered through the course of ongoing discovery.

To the extent any of the prior art references in the attached exhibits are deemed not to disclose one or more limitations of the asserted claims, each of the prior art references in the attached exhibits can be used in combination with any one or more of the references identified in these contentions, including those disclosed in the attached exhibits, to render the asserted claims invalid under 35 U.S.C. §103. For example, Cisco identifies the following non-exhaustive combinations that render the Asserted Claims obvious under 35 U.S.C. § 103:

- Abuan alone, or in combination with at least one of Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Morrison alone, or in combination with at least one of Abuan, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Hornyak alone, or in combination with at least one of Abuan, Morrison, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Guzman alone, or in combination with at least one of Abuan, Morrison, Hornyak, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Lawson alone, or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo,

Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.

- Ludwig alone, or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Dureau alone, or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Van Wie alone, or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Jeong alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Medina alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Tarte alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Mayer alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Jabber, CSF, TokBox, Vidyo,

Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.

- Jabber alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- CSF alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, TokBox, Vidyo, Lync, Radvision, WebRTC, Eisenberg, Beilis, and/or the admitted prior art.
- TokBox alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, Vidyo, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Vidyo alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Lync, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Lync alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Radvision, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Radvision alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, WebRTC, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- WebRTC alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, Abuan 077, Platt, Eisenberg, Beilis, and/or the admitted prior art.

- Abuan 077 alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, WebRTC, Platt, Eisenberg, Beilis, and/or the admitted prior art.
- Platt alone or in combination with at least one of Abuan, Morrison, Hornyak, Guzman, Lawson, Ludwig, Dureau, Van Wie, Jeong, Medina, Tarte, Mayer, Jabber, CSF, TokBox, Vidyo, Lync, Radvision, Abuan 077, WebRTC, Eisenberg, Beilis, and/or the admitted prior art.

These obviousness combinations are provided in the alternative to Cisco's anticipation and single-reference obviousness contentions and are not to be construed to suggest that any reference included in the combination is not itself anticipatory or would not render the Asserted Claims obvious in light of the knowledge of a person having ordinary skill in the art. Cisco also hereby incorporates by reference the prior art, invalidity grounds, and expert testimony submitted in connection with any related proceedings related to the Asserted Patents, including any future *ex parte* reexamination proceedings involving the Asserted Patents or related patents, or any post grant proceedings (e.g., *inter partes* review) involving the Asserted Patents or related patents.

1. Motivations to Combine, Reasonable Expectation of Success, and Obviousness

The ultimate determination of whether an invention is or is not obvious is a legal conclusion based on underlying factual inquires including "(1) the scope and content of the prior art; (2) the level of ordinary skill in the art; (3) the differences between the claimed invention and the prior art; and (4) objective evidence of nonobviousness." *Miles Labs., Inc. v. Shandon, Inc.*, 997 F.2d 870, 877 (Fed. Cir. 1993); *see also Graham v. John Deere Co. of Kan. City*, 383 U.S. 1, 17–18 (1966). The U.S. Supreme Court's decision in *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415–16 (2007), reaffirmed *Graham* and further held that a claimed invention can be obvious even if there is no teaching, suggestion, or motivation for combining the prior art to produce that invention. Under *KSR*, patents based on new combinations of elements or components already known in a technical field are obvious. *See generally KSR*, 550 U.S. 398. Specifically, the

Court in *KSR* rejected a rigid application of the "teaching, suggestion, or motivation [to combine]" test. *Id*. at 418–19. "In determining whether the subject matter of a patent claim is obvious, neither the particular motivation or the avowed purpose of the patentee controls. What matters is the objective reach of the claim." *Id*. at 419. "Under the correct analysis, any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed." *Id*. at 420. In particular, in *KSR*, the Supreme Court emphasized the principle that "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *Id*. at 416. A key inquiry is whether the "improvement is more than the predictable use of prior art elements according to their established functions." *Id*. at 417.

Cisco contends that the Asserted Claims of the Asserted Patents are invalid as obvious because they, at most, are either an obvious modification to known solutions to common issues at the time, or merely arrange known elements, with each performing the same function that had previously been known, to perform and yield no more than what one of ordinary skill would expect from such an arrangement. The Asserted Patents claim many generic elements of methods and/or systems for allegedly to "improving functionality in software applications" by reworking an existing application (a so-called "superblock" application) to add "desired functionality" by incorporating additional executable instructions that "provide such additional functionality" such as audio/video calling, instant messaging, and collaboration via an API) while remaining within the same application context. For example, as discussed above, the Asserted Patents admit that problems related to switching between applications to provide missing functionality was already known (including on mobile devices), and that methods already existed to supply missing functionality (including A/V functionality). The Asserted Patents merely incorporate already-known prior art endpoint technology and standard communication protocols to provide already-existing communication and collaboration functions. It would have been obvious to, in order to circumvent the resource, disruption, and performance issues related to switching between applications, to add "desired functionality" by

incorporating additional executable instructions that "provide such additional functionality" such as audio/video calling, instant messaging, and collaboration via an API) while remaining within the same application context.

Indeed, the Asserted Patents incorporates a prior patent (US 7,656,870) as the basis for the "function block," and reuse standard protocols (SIP, RTP/SRTP, STUN/ICE, HTTP/HTTPS), admitting that none of those building blocks are new. That is exactly the kind of scenario KSR describes: the patent does not identify a new problem or a surprising mechanism; it just proposes to use existing endpoint technology in the expected way to reduce a well-understood nuisance. The "superblock" / "function block" framing is nothing more than a relabeling of then-existing technology.

All of the Asserted Claims of the Asserted Patents are also obvious based on one or more combinations of the prior art references above. The accompanying claim charts explain how different portions of each prior art reference discloses each limitation of the Asserted Claims. To the extent Damaka argues that any particular prior art reference lacks any feature, a person of ordinary skill in the art as of the Asserted Patents' earliest claimed priority date would at a minimum have been motivated to modify the reference to include the allegedly missing feature(s), or to combine it with other references that include that feature. Cisco also hereby incorporates by reference any motivations to combine discussed in any related proceedings related to the Asserted Patents, including any future *ex parte* reexamination proceedings involving the Asserted Patents or related patents, or any post grant proceedings (e.g., *inter partes* review) involving the Asserted Patents or related patents.

One of ordinary skill in the art, at the time of the purported inventions of the Asserted Claims, would have been motivated to combine the teachings of these references and would have had a reasonable expectation of success, as set forth in the combinations above, because these references relate to common objectives and subject matter, and for at least the reasons set forth herein, in the attached claim charts, in the prior art references themselves, and based on the knowledge of one of ordinary skill in the art. The references share commonalities in terms of

their general subject matter as well as the types of equipment and/or approaches used. Further, some of the prior art references explicitly or implicitly reference each other, share common authors or inventors, and/or were developed at common companies, schools, or organizations, which would have motivated one of skill in the art to combine them. For example, these references are directed to the fields of providing additional functionality to existing software in an integrated manner. These references also identify and address many of the same technical issues (*e.g.*, those related to context switching and standalone applications, including on mobile applications) and suggest similar solutions to those issues (*e.g.*, providing functionality—including A/V and related functionality, such as real-time communications, video conferencing, etc.—inside one application, exposing UC capabilities to third-party web apps, and calling a communications engine via APIs to embed A/V functionality).

The open-source nature of many of the already-existing frameworks and tools would have also lowered implementation barriers, allowing for rapid prototyping and integration. In addition, many platforms offered SDKs for third-party extensions, making interoperability between endpoint and network layers both anticipated and technically achievable. Given the modular and flexible nature of these tools, and the evolving programmability of modern applications and platforms, a skilled artisan would have expected that integrating functional elements into existing applications, including mobile applications, would not only be feasible but also yield predictable improvements performance and reliability without requiring undue experimentation. A person of ordinary skill in the art would have also had a reasonable expectation of success in combining these technologies because they operate within well-understood architectures and use interoperable protocols and data structures (such as browser and application APIs).

For instance, a person of ordinary skill in the art would have found it obvious, been motivated to combine, and would have had a reasonable expectation of success in combining the teachings of Abuan, Guzman, Lawson, and Ludwig with the capabilities of Jabber, Webex, Cisco CSF, TokBox, Radvision, and Vidyo. Abuan, Guzman, Lawson, and Ludwig each disclose

technologies for initiating and managing real-time audio/video or messaging sessions, sending notifications, communicating between applications and communication platforms, determining availability/presence, and managing session establishment across networks. Jabber, Webex, CSF, Vidyo, Radvision, and TokBox were well-known, widely deployed systems that solved these exact same problems using established APIs and SDKs. Because all operate within the same field of real-time communications, a POSITA would naturally look to combine them. Moreover, all systems use modular, API-based architectures fully compatible with the printed publications. Each of the asserted printed publications describes a separation between an application layer, and a communication subsystem or engine. They explicitly disclose application-to-platform interactions through APIs, notifications, requests, and responses. Similarly, Jabber, Webex, Cisco CSF, TokBox, Radvision, and Vidyo all provide well-defined APIs for session initiation, signaling, presence, media control, and notifications. These mature communication engines are plug-compatible substitutes for the communication logic described in Abuan, Guzman, Lawson, and Ludwig. A POSITA would expect predictable results from integrating them.

A POSITA would have been motivated to combine these technologies to expand the functionality and enhance the flexibility of then-existing video conference platforms. At the time, mobile developers sought to avoid re-implementing complex call signaling, offload media transport to proven platforms, embed communication capabilities inside third-party applications, and achieve cross-platform interoperability and reliability. This demand is reflected in both the printed publications and the system art. Commercial systems like Jabber, Webex, Vidyo, Radvision, and TokBox existed precisely to provide reusable communication stacks. A POSITA would have been motivated to use them to implement or enhance what the printed publications describe. The combinations simply substitute one known communication engine for another, yielding predictable results. Each printed publication describes a well-known communication pattern for A/V communication sessions and initiating sessions. Each system art reference implements this very pattern, using standard protocols (SIP, XMPP, proprietary signaling,

WebRTC-style flows) and mobile-ready SDKs. Substituting the communication engines in the printed publications with Jabber, Webex, Vidyo, TokBox, etc. would have been routine, predictable, expected to succeed, and fully consistent with the references' teachings. This is exactly the type of predictable substitution that the Supreme Court in *KSR* recognized as obvious.

These publications, moreover, explicitly encourage combinations by describing device-agnostic, server-agnostic architectures. Abuan, Guzman, Lawson, and Ludwig emphasize that communication platforms may be implemented on various devices, signaling and media services may come from remote servers, applications interact with communication engines through software interfaces, and systems are designed to be modular and extensible. These teachings invite integration with established communication systems like Jabber, Webex, Vidyo, Radvision, TokBox, and CSF. A POSITA would reasonably expect successful integration because all such systems are standards-based, interoperable, API-driven, and widely used to embed real-time communication capabilities into third-party applications. Lastly, there are no technological incompatibilities preventing such combinations. Every system art reference was designed for mobile or desktop deployment, integration into existing applications, server-based signaling and media transport, rapid session creation, and presence and notification operations. The printed publications place no constraints on signaling protocols, transport layers, codecs, or OS-level behaviors. Nothing in the references would have deterred a POSITA from using known systems like Jabber, Webex, TokBox, Radvision, CSF, or Vidyo to implement the disclosed functions.

As another example, a person of ordinary skill in the art would have found it obvious, been motivated to combine, and would have had a reasonable expectation of success in combining the teachings of Tarte with the features of well-known communication systems such as Jabber, Vidyo, TokBox, Radvision, and Webex, and would have had a reasonable expectation of success in doing so.

Tarte is expressly directed to real-time communication on mobile devices, including: facilitating real-time audio/visual communications using network connectivity (Tarte, ¶¶ 2, 15), addressing restricted or limited device communications capabilities (Tarte, ¶ 2), enabling users to send and receive contacts' availability (presence), receive notifications, accept incoming communication sessions, and initiate sessions via APIs between applications and communication platforms (Tarte, ¶¶ 32, 45, 48–49). Each of the system art references—including Jabber, Vidyo, TokBox, Radvision, and Webex—was likewise designed to: support real-time voice, video, and messaging, enable presence, contact management, and notifications, integrate with mobile devices and mobile operating systems, expose well-defined APIs/SDKs for app-level communication services, solve challenges around network session management, NAT traversal, signaling, and consistent communication behavior across heterogeneous devices.

A POSITA would naturally look to combine references that address the same real-time communication problems, use similar architectures, and target the identical computing environments. This is a paradigmatic KSR rationale. *KSR*, 550 U.S. at 417. Tarte teaches application-to-platform communication via APIs. Tarte repeatedly discloses a separation between: a social networking application, and a communication platform, with interaction occurring via APIs to initiate or manage communication sessions. (Tarte, ¶¶ 32, 45, 48–49.) Likewise, Jabber's SDKs expose APIs for initiating sessions, presence, and messaging. Vidyo exposes client APIs enabling apps to request video sessions, manage rendering, and exchange session state. TokBox (OpenTok) provides a JavaScript and native mobile API for establishing video calls, handling notifications, and interacting with backend services. And Webex Mobile is API-driven and exposes session initiation and control mechanisms. A POSITA would immediately recognize that Tarte's API-driven separation of application logic and communications logic is compatible with—indeed, directly aligned with—the exact architecture employed in all of the system art. This provides a strong motivation to combine under the Federal Circuit's "useful for its intended purpose" and "predictable variation" rationales. What's more, all systems provided well-established, mature communication engines that would

predictably replace or complement Tarte's communication platform. Tarte's communication platform handles: presence, notifications, initiating real-time communication sessions, and routing requests to the appropriate communication engine. (Tarte, ¶¶ 32, 45, 48–49.) At the time, system-level engines like Jabber, Vidyo, TokBox, and Webex were widely recognized as providing robust, fully developed communication stacks, including: SIP signaling or proprietary signaling engines, NAT traversal technologies, media negotiation, low-latency A/V functionality, mobile-optimized transport layers, and presence and IM subsystems. A POSITA would have viewed these system art technologies as drop-in substitutes or enhancements for the communication platform described in Tarte. Because each system already functioned as a backend communication stack designed for app integration, combining them with Tarte would have been simple, predictable, and routine, satisfying *KSR's* "predictable results" requirement.

Tarte expressly states that its invention addresses the burden on application developers from having to independently implement complex communication logic. (Tarte, ¶¶ 28, 36.) Likewise, system art such as Jabber SDKs, Vidyo, TokBox, Radvision, and Webex Mobile were commercially successful precisely because they abstracted away the complexities of: signaling, session management, device compatibility, media handling, error handling, scalability. A POSITA would naturally be motivated to combine Tarte's architectural approach with these SDKs to achieve: easier application development, more reliable real-time communication services, reduction in engineering overhead, cross-platform interoperability, performance benefits, improved user experience. These motivations are squarely within recognized rationales for obviousness: market pressure, design need, and substitution of one known technology for another. Tarte even explicitly teaches that its communication platform and associated applications: "can be implemented on any type of device capable of executing computer-executable instructions or may be implemented and distributed as a standalone application separate from any hardware." (Tarte, ¶ 40.) This is a clear teaching toward interoperability and modularity. Thus, a POSITA would have reasonably expected that Tarte's communication platform could be replaced with, improved by, or integrated into existing communication engines

like Jabber, Vidyo, TokBox, Webex. A POSITA would recognize that Tarte is expressly teaching a pluggable architecture, which transforms the combination from merely obvious to explicitly encouraged. And because each system art technology had already been deployed at scale on mobile devices (e.g., Jabber, Webex), a POSITA would have had a reasonable expectation of success integrating them with the architecture described in Tarte.

As another example, a person of ordinary skill in the art would have found it obvious, been motivated to combine, and would have had a reasonable expectation of success in combining the teachings of Morrison, Hornyak, and Van Wie with the features and capabilities of well-known communication systems and SDK platforms such as OpenTox, Jabber, Webex, Cisco CSF, Vidyo, WebRTC, and/or Radvision. Morrison, Hornyak, and Van Wie each disclose complementary concepts directed to initiating and managing real-time audio/video communication sessions on devices, embedding communication functionality into user workflows and application user interfaces, and abstracting complex connection and session management behind reusable frameworks and APIs. OpenTox, Jabber, Webex, Cisco CSF, Vidyo, WebRTC and Radvision were well-known systems that solved these same problems in practice through modular architectures that the publications describe at a functional level that were also designed to be embedded into third-party applications and deployed across web and mobile platforms well before the earliest priority date.

Morrison, Hornyak, and Van Wie as well as these systems all arise in the same technical field: real-time communications and video conferencing in mobile and web applications, including how such sessions are initiated, presented, and managed while users interact with other application content. Morrison specifically addresses concurrent display of a video-call interface and an ongoing activity interface on a device, including resizing or overlaying one user interface over the other and adjusting the video call service (e.g., reduced frame rate or quality) when the video call interface is not full-screen. Hornyak addresses integrating third-party communication functionality into a core application user interface so users can select and initiate a communication session without separately launching a separate communication application. Van

Wie teaches a "pervasive realtime framework" that abstracts session establishment, connection management, and switching across communicants or virtual areas so application developers can implement realtime applications without re-implementing low-level connection handling.

TokBox, Vidyo, WebRTC, and Radvision each likewise represent well-known systems that solve these same problems through modular architectures expressly intended for integration into applications. TokBox provides an API platform and SDKs enabling web and mobile applications to create sessions and publish/subscribe to audio/video streams, thereby embedding video calling within the host application rather than requiring a standalone calling client. Vidyo provides SDKs and platforms that enable developers to build and embed multipoint video conferencing (including on mobile devices) into their own applications and portals, expressly to avoid forcing users to toggle between the host application and a standalone communications client. WebRTC provides real-time audio/video communication capabilities through standardized, simple APIs accessible to web applications, allowing applications to embed and control real-time communications directly within their own user interfaces rather than relying on separate applications or plug-ins. And Radvision provides a mobile-focused client framework for standards-based audio/video communications and conferencing client development across platforms, again designed as a reusable framework for embedding real-time communications in products and mobile applications. Given that Morrison, Hornyak, and Van Wie describe application-level session initiation, UI presentation techniques, and reusable framework abstractions in the context of the same RTC field—and because TokBox, Vidyo, WebRTC, and Radvision were well-known implementations providing RTC engines and SDKs designed to be integrated into applications—a POSITA would naturally look to combine them to implement and improve real-time communication functionality in mobile and web applications.

What's more, Morrison, Hornyak, and Van Wie each teach architectures in which the application layer interacts with a communication subsystem/service through software interfaces, notifications, requests, responses, and/or framework abstractions. Morrison includes explicit interactions with a video call service, including notifying the service that the call UI is not full-

screen and receiving reduced video frame rate or quality accordingly—i.e., an application/UI layer influencing service-level media delivery parameters via an interface. Hornyak's core teaching likewise centers on the application layer controlling invocation of communication functionality from within a core app UI, selecting a third-party communication application and executing it through system/application interfaces rather than requiring separate manual app switching. Van Wie explicitly teaches a reusable framework layer that handles complex realtime connection management and switching so that higher-level applications can invoke realtime behaviors through framework APIs rather than implement the low-level communications plumbing themselves. TokBox, Vidyo, WebRTC, and Radvision are direct real-world counterparts to these architectural teachings because each provides a mature, modular communication engine exposed via APIs/SDKs for application integration. Accordingly, a POSITA would understand these systems as compatible substitutes for the generic communication engine/service described in Morrison, Hornyak, and Van Wie. Combining them would require no conceptual leap: it would merely implement the disclosed application/UI patterns and framework abstractions using a known RTC engine as the underlying communications subsystem, yielding predictable results.

A POSITA would likewise have been motivated to implement the publications' generic communication-service/framework concepts using Cisco's mature UC and conferencing systems. For example, prior to the earliest priority date, Cisco with Cisco CSF already provided a reusable client communications layer integrated with enterprise call control. Jabber (including its web integration/SDK model) also provided a unified communications client that combines presence, IM, voice/video calling, conferencing, and related collaboration features in a single application, along with developer-facing integration mechanisms to embed UC features into other applications well before the earliest priority date. Webex similarly provided a mature conferencing system and APIs supporting meeting initiation/join flows and embedded meeting experiences across desktop/web/mobile clients before the earliest priority date. A POSITA would be motivated to use these established Cisco systems to implement and enhance the

workflows and abstractions described by Morrison, Hornyak, and Van Wie because doing so would reduce engineering effort, reuse proven call-control/media stacks, and improve reliability/interoperability—precisely the standard motivation for adopting an SDK/framework approach in RTC.

A POSITA would have had strong motivation to combine these teachings to improve user experience, reduce friction, and increase productivity on mobile and web platforms. Hornyak identifies that users want to initiate communications from within a core application interface without separately opening a communications application, which directly addresses the widespread, well-known problem of disruptive context switching. Vidyo's prior-art systems expressly aimed to avoid forcing users to toggle between a host application and a standalone communications client, demonstrating that the same problem and solution were recognized and commercially important well before Damaka's asserted priority date. TokBox and WebRTC similarly reflect the industry-wide shift toward embedding RTC into applications and web pages via SDKs/APIs. Morrison addresses a closely related usability problem: maintaining a video call while concurrently performing another task on the device, using overlay/resizing layouts and service-level adjustments (e.g., reduced frame rate) to conserve resources and maintain user experience. A POSITA embedding TokBox/Vidyo/WebRTC/Radvision sessions into host applications would be motivated to incorporate Morrison's techniques because embedded RTC is inherently designed to support multitasking workflows. Applying Morrison's UI and resource-control techniques predictably improves performance and user experience for exactly those embedded calling scenarios. Van Wie also teaches the value of realtime session establishment and switching behind a framework so developers can rapidly implement realtime applications without difficult connection-management logic. This motivation aligns with the existence and widespread adoption of RTC SDKs and frameworks like TokBox, Vidyo SDKs, WebRTC, and Radvision. A POSITA would therefore be motivated to implement the Van Wie framework layer using (or wrapping) one of these known RTC engines to accelerate development, ensure interoperability, and leverage mature, tested call stacks.

A POSITA would also have had a strong reasonable expectation of success because the combination is a routine integration of compatible layers: (1) an application/UI layer (Morrison's concurrent display/overlay and resource adjustment; Hornyak's initiation of communications from within a core app UI), together with (2) a reusable RTC engine or framework (TokBox, Vidyo, WebRTC, Radvision, Jabber, Webex, Cisco CSF) that already exposes session initiation, media control, and notifications through APIs/SDKs, and (3) a framework abstraction for connection management and switching (Van Wie). Morrison and Hornyak are largely UI/workflow-level teachings that do not depend on any signaling protocol or codec; Van Wie is an architecture/framework teaching expressly intended to sit above an underlying communications substrate. TokBox, Vidyo, WebRTC, Radvision, Jabber, Webex, and Cisco CSF are exactly such communications substrates, and substituting one known communications engine for another to implement the publications' disclosed functions would have been routine and predictable, yielding the expected result of embedded or integrated A/V communications in host applications without technical incompatibility.

A person of ordinary skill in the art ("POSITA") would further have been motivated to combine the teachings of Lawson, Guzman, Medina, Dureau, and where applicable Mayer with the capabilities of established real-time communications systems and SDKs including TokBox/OpenTok, Vidyo, WebRTC, Radvision, Jabber, Webex, and Cisco CSF. Each of these printed publications addresses well-known sub-problems in the same technical field—real-time audio/video communications embedded within applications—and supplies complementary teachings that naturally and predictably integrate with the system art.

For example, a POSITA would be motivated to combine Lawson's platform-to-client architecture—including authorization tokens, client subscriptions for receiving incoming communication notifications, and system-side bridging of communications—with known RTC engines and SDKs. Lawson teaches how a communication platform securely authenticates and signals client applications and then establishes communication sessions. The RTC systems discussed above already implement the underlying media, signaling, and client-side call control

via well-defined APIs. Substituting or pairing Lawson's platform-side mechanics with any of these known RTC engines would be a routine design choice, yielding predictable results: authenticated client applications receive call notifications and establish voice/video sessions using an embedded SDK or framework. A POSITA would reasonably expect success because Lawson expressly contemplates web and mobile client applications, and the RTC systems are designed for exactly that mode of integration.

As another example, a POSITA would also be motivated to combine Guzman's overlay-based video conferencing UI with Morrison's concurrent activity/video display techniques and any of the RTC systems. Guzman teaches presenting configuration, contacts, and controls as a semi-transparent overlay on top of a video display, thereby keeping the user within the same application and avoiding disruptive switching. Morrison teaches resizing or overlaying video call interfaces and adjusting service-level parameters (e.g., frame rate or quality) during multitasking. The RTC systems discussed above provide the underlying video streams and call control. Combining these teachings is straightforward and predictable: the host application embeds the RTC video stream (TokBox, Vidyo, WebRTC, etc.) and presents controls using Guzman-style overlays while applying Morrison-style resource adaptations. A POSITA would expect this to succeed because these are UI-level techniques layered on top of standard RTC APIs.

As another example, a POSITA would further be motivated to combine Medina's secure identity and ticket/token-based connectivity establishment with Lawson's client authentication and notification mechanisms and with RTC systems such as WebRTC, TokBox, Vidyo, or Radvision. Medina reinforces that exchanging tickets/tokens and connectivity information to establish direct or relayed communication sessions was a known and accepted approach. Lawson supplies complementary platform-to-client signaling and invitation flows, while the RTC systems provide the actual media/session engine. These teachings operate at compatible layers and address the same known problems of secure session establishment and connectivity, giving a POSITA a reasonable expectation of success.

As another example, a POSITA would also have been motivated to combine Dureau's capability-augmentation architecture with Van Wie's realtime framework abstraction and known RTC systems. Dureau teaches that an application can obtain or invoke capabilities not natively present on the host by communicating with an external component or service. Van Wie teaches abstracting realtime connection management and switching behind a reusable framework. The RTC systems (TokBox, Vidyo, WebRTC, Radvision, etc.) are precisely such external capability providers and frameworks. Combining these teachings would be routine and predictable, yielding a modular architecture in which a host application invokes a reusable communications component to supply A/V capabilities.

Mayer further enforces that integrated unified communications user interfaces—combining presence, messaging, voice, and video within a client—were well known. A POSITA would have been motivated to apply Mayer's UC integration concepts using Cisco's mature UC systems (Jabber, Webex, Cisco CSF), reinforcing the expectation that such integrations were routine and successful before the asserted priority date.

A POSITA would have also been motivated to combine Abuan '077's mobile video conference network management techniques with known A/V systems such as WebRTC, TokBox/OpenTok, Vidyo, Radvision, Jabber/Webex/Cisco CSF, and/or with printed publications like Morrison and Van Wie, because all operate in the same field of mobile real-time A/V conferencing and address the same known constraints: variable wireless bandwidth, mobile resource limitations, and the need to maintain call quality while network conditions change. Abuan '077 discloses managing a video conference by identifying ceiling bit rates, receiving networking data from the remote device, and incrementally increasing (or decreasing) the bitrate based on that data—expressly recognizing that available bandwidth can change during the conference and that flooding the network with excessive bitrates/signaling is undesirable. Abuan '077 also teaches using RTP stream mechanisms (e.g., RTP header extensions) to convey network data used for bitrate adaptation, reinforcing that these are conventional, standards-oriented techniques in A/V and RTC implementations.

A POSITA would reasonably expect success in combining Abuan ’077 with the system art and publications because the combination is a straightforward layering: Abuan ’077 supplies concrete adaptive-network-management logic for mobile video conferences, while the RTC systems provide mature media/session engines and SDKs, and Morrison/Van Wie provide complementary UI/workflow and framework abstractions for mobile multitasking and session management. Nothing in Abuan ’077 requires a particular proprietary signaling stack; it is compatible with the same kinds of real-time media transport architectures used by the prior A/V / RTC systems and the application-layer frameworks described in the printed publications.

A POSITA would also have been motivated to combine Platt’s real-time communication initiation and recipient discovery techniques with the same A/V systems and publications because Platt addresses a common, well-known problem in real-time communications: users often wish to initiate communication with people nearby or within a context (e.g., a conference or event) without knowing traditional identification information such as phone number, email address, or username. Platt teaches enabling real-time communication where traditional identification is unknown by selecting criteria such as geographic proximity (and other attributes) to identify recipients and communicate with them. This combines naturally with Hornyak (integrating communications initiation within a core app UI) and with platform/session-establishment references like Lawson and Medina (token-based session initiation, notifications, identity/connectivity mechanisms), as well as with A/V engines like WebRTC/TokBox/Vidyo/Radvision/Jabber/Webex/Cisco CSF, which provide the underlying session/media stack once a recipient (or group) is identified. A POSITA would have had a reasonable expectation of success because Platt’s contribution is primarily at the discovery/addressing layer (how to identify who to contact without traditional IDs), while the other publications and systems provide well-established, API-driven mechanisms for initiating and maintaining real-time communication sessions after the recipient(s) are selected.

As another example, a POSITA would be motivated to combine Beilis with the publications discussed above and the systems and RTC engines discussed because Beilis is

directed to sharing a physical display among multiple applications on a mobile platform using a layered approach with transparency and "dominancy," including intercepting inputs and dispatching them to the appropriate application—explicitly aimed at reducing the need to toggle between apps while still viewing updated information. That objective directly aligns with the asserted "disruptive context switching" problem and with Morrison's concurrent/overlay display approach and Guzman's overlay UI concepts. A POSITA would reasonably expect success in combining Beilis with RTC systems (Jabber/Webex/CSF/WebRTC/TokBox/Vidyo/Radvision) because the combination simply applies known UI-layer compositing/transparency/input-routing techniques (Beilis/Morrison/Guzman, for example) on top of known RTC engines to keep the call window visible while users interact with other app content—predictable layering, not protocol innovation.

Likewise, a POSITA would be motivated to combine Eisenberg with publications discussed above (such as Lawson/Van Wie/Hornyak) and the systems and RTC engines discussed above because Eisenberg teaches initiating and supporting network video conferencing using instant-messaging-enabled clients, with an IM server supporting instant messages between client nodes and conferencing server support (e.g., MCU-type conferencing). This maps cleanly onto: Hornyak's "initiate communication from within an application UI" concept (IM client UI initiates a video conference), Lawson's platform-to-client session establishment/notifications (use messaging/notifications to set up calls), and Van Wie's framework abstraction for managing realtime connections and switching (wrap IM + video conference initiation/management in a reusable realtime framework). A POSITA would also be motivated to implement Eisenberg's IM-initiated conferencing using mature A/V calling system, SDKs, and RTC engines (Jabber, Webex, CSF, TokBox, Vidyo, WebRTC, Radvision) because those systems provide the standardized call/media stacks and APIs needed once initiation/signaling occurs. This is a predictable "use a known communications engine to implement the disclosed conferencing session" substitution, with a reasonable expectation of success.

A POSITA would have been further motivated to combine Jabber with other known RTC systems (WebRTC, TokBox, Vidyo, Radvision) to obtain predictable benefits such as cross-platform/browser reach, interoperability, redundancy, and faster integration into web applications. As discussed in Cisco's accompanying invalidity claim charts, Cisco's DevNet materials describe that the Voice and Video SDK uses a browser add-on (Cisco Web Communicator add-on) to provide call control and media termination within the browser and is intended for major browsers.

A POSITA would view this browser plug-in approach as a known "RTC engine exposed via web APIs," and would be motivated to combine it with: WebRTC, to provide a standards-based, plugin-free RTC path in environments where WebRTC was available, while retaining the Jabber plug-in path for enterprise call-control integrations and legacy browser support; WebRTC itself is an open RTC framework available via JavaScript APIs; and TokBox / Vidyo / Radvision, as alternative or complementary RTC engines (cloud CPaaS, enterprise SDK, standards-based client framework) that could be integrated behind the same web-application UX (e.g., selecting an engine based on policy, network, or device), yielding predictable substitution results. (Vidyo explicitly positioned its SDK/platform for embedding conferencing inside applications without forcing users to toggle to a standalone client. Radvision likewise provided a client framework/SDK (BEEHD) enabling HD video-enabled applications and a mobile client (Scopia).) The Jabber Web SDK architecture was expressly designed for embedding UC calling into web applications through a browser add-on + JS interface. Pairing it with other RTC engines is a routine "known-substitute engine" design choice (choose/abstract the RTC engine behind a common app workflow), not a technical leap.

Moreover, a POSITA would have been motivated to combine Jabber with the publications discussed above given that they teach application/UI workflows, multitasking/overlay techniques, framework abstractions, session establishment, and call initiation patterns that are directly and predictably implemented using a mature browser-embedded calling engine like Jabber Web SDK. For example, Hornyak teaches initiating

communications from within a host/core application UI rather than forcing users to separately open another communications application. Using the Jabber Web SDK (voice/video in a browser via JS API + plug-in) provides a ready-made way to implement Hornyak's in-app initiation concept directly inside a web portal or line-of-business application. Morrison likewise teaches presenting video call UI concurrently with other activities and adjusting service parameters when the call UI is not full screen. And Guzman teaches overlay controls/UI on a video display. A POSITA would combine these UI teachings with Jabber Web SDK's browser-embedded call engine to present call windows/controls within the same web application context (e.g., overlays, reduced-size windows) while maintaining call quality predictably. As another example, Van Wie teaches a realtime framework that abstracts connection/session management and switching. Jabber Web SDK's call-control/media termination in-browser provides a natural underlying communications substrate for such a framework, letting the host application invoke high-level functions through JS interfaces rather than reimplementing low-level comm plumbing. As another example, Lawson's token/subscription/notification-style call establishment and Abuan's mobile video-conference operations are complementary platform/app workflows that a POSITA could implement using an embedded Jabber web calling engine (client side) plus standard server-side signaling, producing predictable results. Lawson describes authorization tokens and client subscriptions for incoming communication notifications. Abuan '075 describes operations during a real-time communication session. Moreover, Platt teaches initiating RTC where traditional identification information may be unknown (e.g., selecting recipients based on criteria such as proximity). A POSITA would pair that "who to contact" layer with Jabber Web SDK as the "how to establish the call" layer within a web UI, yielding a straightforward combination. Cisco's Jabber SDK was expressly marketed for integrating UC capabilities (voice, video, IM, presence, conferencing) into web applications. The publications are largely architecture/UI/workflow teachings and do not impose signaling/protocol constraints inconsistent with a browser-embedded SDK approach.

For these reasons, a POSITA would also have been motivated to adapt or provide Jabber SDK functionality on a broad variety of mobile devices and operating systems (including iOS and Android) because, by the relevant timeframe, mobile users increasingly needed embedded voice/video and collaboration inside mobile workflows. Systems like Vidyo and Radvision expressly supported mobile SDK-based multipoint video conferencing (e.g., VidyoTechnology SDK for smart devices), reinforcing that mobile SDK deployment was a known and commercially important direction. A POSITA would therefore be motivated to extend Jabber's SDK approach to mobile clients to reuse the same proven UC call-control features while implementing mobile-appropriate UIs (e.g., Morrison-style overlays/multitasking) and network-adaptation techniques (e.g., Abuan-style bitrate management), with a reasonable expectation of success given that the underlying RTC stacks, protocols, and SDK integration patterns were mature and widely deployed before July 2013.A person of ordinary skill in the art would have also been motivated to combine the teachings of any of the asserted printed publications (including Abuan, Morrison, Hornyak, Van Wie, Lawson, Guzman, Medina, and Dureau) with Webex, and would have had a reasonable expectation of success in doing so. Prior to the asserted priority date, Webex was a mature, widely deployed real-time collaboration platform providing integrated audio conferencing, video conferencing, content sharing, meeting management, and cross-platform client support across desktop, web, and mobile environments. As discussed above, Webex already solved the same core problems purportedly addressed by the asserted patents—enabling real-time A/V communication within a single application context, reducing disruptive context switching, and exposing communication functionality through modular, API-driven architectures.

As discussed above, before the earliest priority date, Webex offered full-featured online meetings that combined video windows, audio, participant lists, chat, and shared content panels within a single user interface, on both desktop and mobile clients. Webex mobile applications allowed users to join meetings, view shared content, participate in audio and video, and interact with meeting controls without switching between separate applications. These capabilities

directly addressed the same "stay in one app" and "embedded video window" concerns emphasized in the asserted printed publications (e.g., Abuan, Morrison, Hornyak, Van Wie, Lawson, Guzman, Medina, and Dureau). A POSITA would therefore recognize Webex as a natural implementation substrate for the application-level workflows and UI concepts disclosed in these publications.

Before the earliest priority date, Webex also provided XML and URL-based APIs that allowed third-party applications and enterprise portals to programmatically create meetings, schedule sessions, and initiate or join meetings. These APIs exemplify the same architectural separation described in the asserted publications: an application layer invoking a communication service/engine through well-defined interfaces. As such, a POSITA would have been motivated to implement the generic "communication subsystem," "function block," or "realtime framework" described in Van Wie, Lawson, Medina, or Dureau using Webex's existing APIs and backend services, rather than re-implementing signaling, media transport, or meeting logic from scratch.

Webex relied on standard networking concepts and protocols and was designed to interoperate across heterogeneous devices and networks. This made Webex readily combinable with other known A/V systems and frameworks, such as TokBox/OpenTok, Vidyo, WebRTC, and Radvision. For example, a POSITA could straightforwardly combine Webex's meeting services with: (i) Morrison-style multitasking and overlay UI techniques, by resizing or embedding Webex video windows while other application activities continued; (ii) Hornyak-style integrated initiation, by launching or joining Webex meetings from within a core application UI; (iii) Van Wie-style realtime frameworks, by wrapping Webex meeting/session management behind a reusable abstraction layer; (iv) Lawson or Medina token-based authentication and notification mechanisms, by using Webex's existing meeting access controls and invitation flows. Such combinations involve only predictable substitutions and layering of known components and would not require modification of Webex's fundamental operation.

A POSITA would have been motivated to combine Webex with the asserted publications and systems discussed above to reuse a proven, enterprise-grade communication platform, reduce engineering complexity, and deliver reliable, scalable A/V collaboration. Webex's long-standing commercial success and widespread deployment would give a POSITA a strong reasonable expectation of success in using Webex as the communication engine underlying the application-level concepts disclosed in the printed publications. There are no technical incompatibilities: the publications do not mandate any particular signaling protocol, codec, or backend architecture, and Webex already supported the required session establishment, media delivery, and UI integration patterns.

In sum, combining pre-2013 Webex with any of the asserted publications or with other RTC systems would have been an obvious design choice. Webex represents a known, interchangeable communication engine that performs the same functions attributed to the generic "communication service," "function block," or "realtime framework" in the asserted patents and publications. Substituting Webex for such a generic engine—or combining Webex with UI/workflow techniques discussed in the publications above (e.g., Abuan, Morrison, Hornyak, Guzman, Van Wie, Lawson, Medina, or Dureau)—would yield predictable results and is precisely the type of routine combination recognized as obvious under *KSR*.

Given the Asserted Patents' extensive admissions regarding the state of the art, a person of ordinary skill in the art would have also found it obvious to combine the admitted prior art with any of the printed publications or real-world systems discussed above. The Asserted Patents expressly acknowledge that (i) mobile operating systems already executed applications in separate contexts, (ii) switching between applications to obtain audio, video, messaging, or meeting functionality was common and well understood, (iii) such context switching was known to be disruptive and resource-intensive, and (iv) full-featured communication endpoints providing audio/video calling, messaging, and collaboration—implemented using standard protocols and exposed through APIs—already existed and were widely deployed. These admissions establish both a known problem (disruptive context switching and fragmented

functionality) and known solution components (modular communication endpoints, SDKs, APIs, and standard protocols).

In light of these admissions, a POSITA would have been strongly motivated to apply the application-level workflows, UI techniques, and framework abstractions disclosed in the printed publications (e.g., Morrison's concurrent/overlay display techniques, Hornyak's in-app initiation of communications, Van Wie's realtime framework abstraction, Lawson's token-based session establishment, Guzman's overlay call controls, Medina's ticket-based connectivity, or Dureau's capability-augmentation architecture) using the already-known communication systems and SDKs acknowledged in the specification and discussed above (including WebRTC, Jabber and Jabber, Cisco CSF/CUCM, Webex, TokBox/OpenTok, Vidyo, Radvision, and others). Doing so would merely substitute a known, interchangeable communication engine or framework for the generic "function block" described in the Asserted Patents, while applying well-known UI and integration patterns to reduce context switching and improve usability.

A POSITA would also have had a reasonable expectation of success because the Asserted Patents admit that the underlying communication functionality, protocols, and endpoint architectures were already mature, standardized, and API-driven. The printed publications do not require any novel signaling, codec, or operating-system behavior, and the prior art systems were expressly designed to be embedded into applications and invoked through APIs. Combining the admitted prior art with these publications and systems would therefore involve only predictable substitutions and routine integration of known components, yielding the expected result of embedded audio/video and collaboration functionality within a host application. Under *KSR*, such combinations—rearranging known elements according to known methods to address a known problem—are obvious as a matter of law.

V. GROUNDS OF INVALIDITY OTHER THAN ANTICIPATION OR OBVIOUSNESS

The Asserted Claims are also invalid for failure to meet one or more of the requirements of 35 U.S.C. § 112.

Cisco reserves the right to amend, supplement, and/or otherwise modify these Preliminary Invalidity Contentions with respect to grounds of invalidity other than anticipation or obviousness. Due in part to Damaka's failure to sufficiently identify (a) the full scope of the accused instrumentalities and (b) where Damaka contends that each element of the Asserted Claims is found within each instrumentality, Cisco lacks a complete understanding of Damaka's alleged infringement and/or claim construction positions. If and when Damaka develops an infringement and/or claim construction theory that implicates infringement by Cisco, Cisco reserves the right to argue that the specification does not enable and/or provide a written description for such an interpretation or that such an interpretation renders the claim indefinite. The Court has not yet construed any terms, phrases, or clauses of the Asserted Claims, and Damaka has not yet provided any validity contentions in response to Cisco's Preliminary Invalidity Contentions. In particular, Damaka has not identified any way in which it disputes that the Asserted Claims are invalid under 35 U.S.C. § 112. Moreover, Cisco's discovery and investigation in connection with this lawsuit is ongoing, and these grounds are based on information obtained to date. Accordingly, Cisco reserves the right to amend, supplement, and/or otherwise modify these Preliminary Invalidity Contentions. Cisco further reserves the right to rely on expert reports or testimony and any third-party discovery.

Cisco lists below exemplary grounds upon which Cisco presently contends the Asserted Claims of the Asserted Patents are invalid for failure to meet one or more of the requirements of 35 U.S.C. § 112. Cisco identifies at least the following grounds of invalidity based on inadequate written description under 35 U.S.C. § 112(a), lack of enablement under 35 U.S.C. § 112(a), and indefiniteness under 35 U.S.C. § 112(b) for the Asserted Claims. These grounds are identified based on knowledge currently in Cisco's possession. A more detailed basis for Cisco's written description, enablement, and/or indefiniteness defenses will be set forth in Cisco's claim construction briefs and expert reports on invalidity, to be served in accordance with the Docket Control Order. Further investigation may uncover additional grounds for invalidity, and Cisco reserves the right to supplement these disclosures to include all additional

grounds.

A. Lack of Written Description and Lack of Enablement Under 35 U.S.C. § 112

Pursuant to 35 U.S.C. § 112(a), a patent specification:

> shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make, and use the same[.]

The Federal Circuit has held that this language creates two closely related, yet separate requirements for a specification: (i) a written description of the invention ("written description") and (ii) a written description of the manner and process of making and using the invention ("enablement"). *See Ariad Pharm., Inc. v. Eli Lilly & Co.*, 598 F.3d 1336, 1344 (Fed. Cir. 2010) (*en banc*).

To satisfy the written description requirement, the description must "clearly allow persons of ordinary skill in the art to recognize that [the inventor] invented what is claimed." *Ariad Pharm., Inc. v. Eli Lilly and Co*., 598 F.3d 1336, 1351 (Fed. Cir. 2010) (internal citation omitted). The test for sufficiency is whether the disclosure of the application relied upon reasonably conveys to those skilled in the art that the inventor had possession of the claimed subject matter as of the filing date. *Id*.

The test requires an objective inquiry into the four corners of the specification from the perspective of a person of ordinary skill in the art. Based on that inquiry, the specification must describe an invention understandable to that skilled artisan and show that the inventor actually invented the invention claimed. "Whether the written description requirement is satisfied is a fact-based inquiry that will depend on the nature of the claimed invention, and the knowledge of one skilled in the art at the time an invention is made and a patent application is filed." *Carnegie Mellon Univ. v. Hoffmann La Roche Inc.*, 541 F.3d 1115, 1122 (Fed. Cir. 2008) (internal citation omitted). Actual "possession" or reduction to practice outside of the specification is not enough. Instead, the specification itself must demonstrate possession.

While the written description requirement does not demand any particular form of disclosure, a description that merely renders the invention obvious does not satisfy the requirement. *Lockwood v. Am. Airlines*, 107 F.3d 1565, 1571-72 (Fed. Cir. 1997).

To satisfy the enablement requirement of 35 U.S.C § 112, the disclosure "must teach those skilled in the art how to make and use the full scope of the claimed invention without 'undue experimentation.'" *Genentech, Inc. v. Novo Nordisk, A/S*, 108 F.3d 1361, 1365 (Fed. Cir. 1997) (citations omitted). Moreover, "[i]t is the specification, not the knowledge of one skilled in the art, that must supply the novel aspects of [the] invention in order to constitute adequate enablement." *Id.* at 1366. The Federal Circuit has enumerated several factors to consider in determining whether a disclosure would require "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." *In re Wands*, 858 F.2d 731, 737 (Fed. Cir. 1988).

Cisco provides below an exemplary list of claim elements for which there is inadequate written description and/or lack of enablement under U.S.C. § 112(a). For example, certain claim limitations fail to describe an invention understandable to a skilled artisan at the time of the invention and/or fail to show that the inventor actually possessed the invention claimed. *See Ariad*, 598 F.3d at 1351. As a further example, certain claims are invalid because the specification fails to teach those skilled in the art at the time of the alleged invention how to make and use the full scope of the claimed invention without undue experimentation. *See ALZA Corp. v. Andrx Pharms., LLC*, 603 F.3d 935, 940 (Fed. Cir. 2010). As an additional example, certain claim limitations encompass any and all structures or acts for performing a recited function, including those beyond what their applicant(s) invented, such that the specification fails to provide a scope of enablement commensurate with the scope of the claim as asserted. *See Halliburton Oil Well Cementing Co. v. Walker*, 329 U.S. 1, 12–13 (1946). As yet another example, certain claims are invalid for failing to enable the full scope of the invention as defined

by the claims. *See Amgen Inc. v. Sanofi*, 143 S. Ct. 1243, 1254 (2023) ("If a patent claims an entire class of processes, machines, manufactures, or compositions of matter, the patent's specification must enable a person skilled in the art to make and use the entire class.").

In addition, and without limitation, the Asserted Claims taken as a whole (rather than only as the sum of its individual limitations) are invalid for lack of written description and enablement. *See, e.g., Novozymes A/S v. DuPont Nutrition Biosciences APS*, 723 F.3d 1336, 1346–51 (Fed. Cir. 2013). As described below, by way of example, a person of ordinary skill in the art at the time of the alleged invention would not understand that the named inventors of the Asserted Patents were in possession of the claimed subject matter as of the effective filing date, nor would the patent's disclosure have enabled one skilled in the art at the time of the alleged invention to practice the Asserted Claims without undue experimentation.

Any deficiencies that render claims invalid for written description and/or enablement under 35 U.S.C. § 112 also infect and thereby invalidate any and all claims depending therefrom.

In addition to the claim as a whole, at least the following claim elements, or portions thereof, of the claims in the Asserted Patents are not supported by the written description and/or are not enabled under 35 U.S.C. § 112(a):

- **'032 patent:**
    - The Asserted Claims are invalid for failure to satisfy the requirements of 35 U.S.C. § 112 because the specification as filed does not contain an adequate written description and/or enabling disclosure of the claimed "function block" or its claimed roles and functions, including "A/V call functionality provided by the function block," "establishing, by the function block, the A/V call," "providing, by the function block, video output," "providing, by the function block, audio for the A/V call," "whether the function block has the capability to provide the A/V call," and "whether the function block has the capability to render the A/V call."

- "Superblock application" and "establishing is not supported by the superblock application."
- "A/V call" and "A/V call functionality."
- "Resource that is external to the mobile device" and "establishing" calls via a network interface to such resources.
- "video window within a display window . . . ."
- "Capability to provide/render the A/V call"
- "authorized to access A/V call functionality."
- "API call from the superblock application to the function block."
- "mobile devices" and breadth across OS environments

- **'092 patent:**
  - The Asserted Claims of the '092 patent are invalid for failure to satisfy the requirements of 35 U.S.C. § 112 because the specification as filed does not contain an adequate written description and/or enabling disclosure of "function block" and its related functional definitions, including "function block having audio/video (A/V) call functionality," "configured to provide functionality for the communication session to the superblock application," and "capability to provide the communication session" / "capability to perform any rendering needed." The Asserted Claims recite a "function block" almost entirely by behavior ("providing," "determining," "capability to…") without defining specific structure, APIs, processes, data structures, algorithms, or interfaces that concretely implement those behaviors.
  - The Asserted Claims of the '092 patent are invalid for failure to satisfy the requirements of 35 U.S.C. § 112 because the specification as filed does not contain an adequate written description and/or enabling disclosure of "superblock" and its related functional definitions, including ""superblock

application stored in the memory" and "without switching context from the superblock application to another application." The Asserted Claims recite a "superblock application," which is merely a coined term with broad and indefinite functional descriptions (a set of executable instructions performing one or more functions) lacking structural limits. The claims recite various functions on what the "superblock" can or cannot allegedly support, but, the specification does not define what qualifies as a superblock application versus any other application or how to implement any of the claimed functionality.

- The Asserted Claims of the '092 patent are invalid for failure to satisfy the requirements of 35 U.S.C. § 112 because the specification as filed does not contain an adequate written description and/or enabling disclosure of the claimed "authorization" and "authorized to access" elements. The Asserted Claims recite a superblock or function block authorized to access certain resources, but lack any supporting disclosure of authorization systems, methods, or functions and fail to teach enabling the potential authorization mechanisms.

- The Asserted Claims of the '092 patent are invalid for failure to satisfy the requirements of 35 U.S.C. § 112 because the specification as filed does not contain an adequate written description and/or enabling disclosure of the claimed "Video output" and "audio output" elements. The written description does not describe, support, or show evidence of possession of any of the claimed A/V output mechanisms.

- "Establishing . . . not otherwise supported by the superblock application"

- "Resource that is external to the . . . system / computing entity" / "access to a resource that is external to the data processing system" / "external resource" / "external device"

- "Communication session" / "A/V service" / "Audio/Video (A/V) call"

- "Video window within a display window … displayed simultaneously"

- Claimed "authorized to access functionality" elements (e.g., "determine … that the superblock application is authorized to access . . ."

- "Capability to Render / Provide" functional phrases (e.g., "capability to render the communication session" / "capability to provide the communication session")

- "Without Switching Context . . ."

- "Initializing . . . when the . . . application is loaded and executed"

- "data processing system" / "a computing entity"

- “application”
- “Context switching” / “switching context”
- “display window” / “video window within the display window” / “a display window associated with the superblock application”
- “Communication session”
- “Rendering”
- “Configuration parameter enabling configuration after distribution”
- “Collaborate on a document / file / application”
- “External device”
- “Service provision requested”
- “Establishing connection with external devices via network interface”

- ’744 patent
  - “function block” / “superblock”
  - “wherein the superblock application is configured to access the function block for functionality that is lacking in the superblock application”
  - “wherein the establishing is not supported by the superblock application”
  - “configured to access the function block for functionality that is lacking in the superblock application”
  - “superblock application . . . configured to access the function block for functionality that is lacking”
  - “communication service”
  - “providing”
  - “API call from the superblock application to the function block”
  - “capability to. . .” / “capability to perform any rendering needed by the communication service” / “capability to provide the communication service”
  - “without switching context” / “without switching context from the superblock application to another application”

- “external device” / “requires access to an external device”
- “bridge” / “operating . . . as a bridge between the superblock application and the external device”
- “establishing a connection”
- “authorized to access functionality”
- “any rendering needed” / “whether the superblock has a capability to perform any rendering needed”
- “configuration parameter. . . . distributed”
- “initializing the function block . . . when the superblock application is loaded and executed”

- ’046 patent
  - “function block” / “superblock”
  - “wherein the function block is accessible only to the superblock application on an electronic device”
  - “wherein the service is not otherwise supported by the superblock application”
  - “and wherein the function block is accessible only to the superblock application on the electronic device”
  - “accessible only to the superblock application”
  - “establish a connection with an external device”
  - “provide a service requested by the superblock application”
  - “service”
  - “communication session”
  - “configured to provide functionality for the communication session”
  - “notify the superblock application”
  - “capability to perform any rendering needed by the communication session”
  - “authorized to access functionality provided by the function block for the service”

- “authorization credentials”
- “service is available on the external device”
- “accessing a resource”
- “resource”
- “service is not otherwise supported by the superblock application”
- “operating as a bridge between the superblock application and the resource”
- “initializing the function block when the superblock application is loaded and executed”
- “capable of performing any rendering needed by the service”
- “request … received as an API call”
- “resource is available”

- ’362 patent
  - : “receiving, by the superblock application”
  - “communicating, by the superblock application”
  - “providing, by the superblock application”
  - “creating for storage on a server”
  - “providing the rendering information”
  - “rendering information”
  - “function block compiled therein”
  - “communicates . . . using API calls”
  - “real-time communication session”
  - “remote resource”
  - “notification”
  - “rendering information”
  - “rendering . . . the real-time communication session”

- “information corresponding to a mic”
- “information corresponding to a speaker”
- “information for displaying a window”
- “video window”
- “instant message window”
- “resizable window”
- “information corresponding to a camera”
- “presence indicator”
- “status of the second user”
- “API calls compatible with an interface provided by its respective function block”
- “voice call”
- “video call”
- “adapted to operate with iOS”
- “adapted to operate with Android”
- “mobile device”
- “desktop computer”
- “compiled into the superblock application”
- “adapted to interact with one or more servers”
- “provide the real-time communication session within the superblock application”

- **’116 patent**
  - “third party”
  - “authorization key”
  - “superblock application”
  - “function block”

- “third party superblock application”
- “additional functionality”
- “configured to be compiled into the superblock application”
- “configured to add the additional functionality”
- “real-time communication session”
- “one or more servers connected over the internet”
- “interact … through API calls”
- “authorization key”
- “unlocks the function block”
- “authentication credentials”
- “notification”
- “notification requires a response”
- “instant messaging”
- “voice call”
- “video call”
- “adapted to operate with iOS”
- “adapted to operate with Android”
- “computing device”
- “cellular phone”
- “smart phone”
- “tablet”
- “laptop computer”
- “desktop computer”
- “interact with one or more servers to establish the real-time communication session”
- “API call indicates whether the request has been granted or denied”

- "provide the real-time communication session to the superblock application"
- "providing an authorization key"
- "providing authentication credentials"

These elements fail to meet the written description requirement and/or are not enabled. The Asserted Patents would not convey to one of skill in the art that, as of the filing date, the inventor had possession of the claimed subject matter. Nor would the Asserted Patents teach one of ordinary skill in the art how to practice these claim elements without undue experimentation. To the contrary, the Asserted Patents recite only generic components in the embodiments disclosed. Additionally, the claims containing these limitations are invalid for failing to enable the full scope of the invention as defined by the claim.

B. Indefiniteness Under 35 U.S.C. § 112

Below is an exemplary list of claim elements[3] that are indefinite and/or render claims that include such elements indefinite under 35 U.S.C. § 112(b). "[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); *see also Allen Eng'g Corp. v. Bartell Indus., Inc.*, 299 F.3d 1336, 1348 (Fed. Cir. 2002) (quoting *Solomon v. Kimberly-Clark Corp.*, 216 F.3d 1372, 1377 (Fed. Cir. 2000) ("[F]irst, [the claim] must set forth what 'the applicant regards as his invention' and second, it must do so with sufficient particularity and distinctness, i.e., the claim must be sufficiently, 'definite.'") (second alteration in original)).

[3] Any discrepancy between the language included in the bullets of this paragraph and the language of the corresponding claim element is inadvertent, and it is Cisco's contention that the claim element is not adequately described and/or enabled under 35 U.S.C. § 112(a) and/or is indefinite under 35 U.S.C. § 112(b).

Because Cisco's investigation, prior art search, analysis, discovery, and trial preparation are still ongoing, Cisco expressly reserves the right to amend, supplement, and/or otherwise modify these Preliminary Invalidity Contentions, including by identifying and providing additional evidence and arguments that the Asserted Claims are indefinite under 35 U.S.C. § 112(b) following any constructions Damaka proposes or the Court adopts.

Any deficiencies that render claims invalid for indefiniteness under 35 U.S.C. § 112(b) also infect and thereby invalidate any and all claims depending therefrom.

At least the following claim elements, or portions thereof, used in the Asserted Patents are indefinite under 35 U.S.C. § 112(b) for failure to inform, with reasonable certainty, those skilled in the art about the scope of the invention:

- **'032 patent:**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)
  - "authorized to access A/V call functionality provided by the function block"
  - "the A/V call" / "A/V call functionality" / "audio/video (A/V) call"
  - "a resource that is external to the mobile device"
  - "establishing, by the function block, the A/V call between the Superblock application and the resource via a network interface" (also a §112(f) nonce term lacking corresponding structure)
  - "wherein the establishing is not supported by the superblock application" (also a §112(f) nonce term lacking corresponding structure)
  - "a video window within a display window of the superblock application on the mobile device" (also a §112(f) nonce term lacking corresponding structure)

- "wherein the video window is displayed simultaneously with the display window of the Superblock application" (also a §112(f) nonce term lacking corresponding structure)
- "providing . . . audio for the A/V call via the mobile device"
- "determining . . . whether the function block has the capability to provide the A/V call"
- "determining . . . whether the function block has the capability to render the A/V call"
- "determining . . . whether the superblock has the capability to render the A/V call."
- "initializing the function block, by the superblock application, when the superblock application is loaded into the memory and executed" (also a §112(f) nonce term lacking corresponding structure)
- "the request is an application programming interface (API) call from the super block application to the function block." (also a §112(f) nonce term lacking corresponding structure)

- **'092 patent:**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)
  - "not otherwise supported" / "wherein establishing is not otherwise supported by the superblock application"
  - "capability to provide…" / "capability to perform any rendering needed…" / (also a §112(f) nonce term lacking corresponding structure)
  - "any rendering needed" (also a §112(f) nonce term lacking corresponding structure)

- "resource that is external to the data processing system / computing entity" (also a §112(f) nonce term lacking corresponding structure)
- "external device"
- "operate as a bridge between the superblock application and the external resource / external device" (also a §112(f) nonce term lacking corresponding structure)
- "without switching context from the superblock application to another application …" (also a §112(f) nonce term lacking corresponding structure)
- "video window within a display window associated with the superblock application, the video window displayable simultaneously with the display window of the superblock application"
- "authorized to access . . . " (*also a §112(f) nonce term lacking corresponding structure*)
- "communication session"
- "audio/video (A/V) service" / "A/V call functionality"
- "function block having audio/video (A/V) call functionality"
- "configuration parameter" / "configured after the function block is distributed" (*also a §112(f) nonce term lacking corresponding structure*)
- "collaborate on at least one of a document, a file, or an application" (*also a §112(f) nonce term lacking corresponding structure*)
- "when the superblock application is loaded and executed"
- "The A/V service" vs. "an audio/video (A/V) service request"
- "the display window associated with the superblock application"
- "another application" / "another application associated with the computing entity" (*also a §112(f) nonce term lacking corresponding structure*)
- "the resource"
- "the communication session" vs. "a communication session"

- **'744 patent**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)

- "wherein the superblock application is configured to access the function block for functionality that is lacking in the superblock application"
- "wherein the establishing is not supported by the superblock application"
- "configured to access the function block for functionality that is lacking in the superblock application"
- "communication service"
- "requires access to an external device"
- "external device" / "establishing . . . a connection with the external device" (*also a §112(f) term lacking corresponding structure*)
- "wherein the establishing is not supported by the superblock application"
- "providing . . . the communication service to the superblock application" (*also a §112(f) nonce term lacking corresponding structure*)
- "performed without switching context from the superblock application to another application on the mobile device"
- "operating . . . as a bridge" (*also a §112(f) term lacking corresponding structure*)
- "application programming interface (API) call"
- "capability to provide the communication service"
- "capability to perform any rendering needed by the communication service"
- "whether the superblock has a capability to perform any rendering needed by the communication service"
- "initializing the function block . . . when the superblock application is loaded . . . and executed" (*also a §112(f) nonce term lacking corresponding structure*)
- "authorized to access functionality provided by the function block"
- "determining . . . that providing the communication service . . . requires access to an external device" / "determining . . . whether the function block has a capability to provide the communication service" / "determining . . . whether the function block has a capability to perform any rendering needed" / determining . . . whether the superblock has a capability to perform any rendering needed" / "determining . . . that the

superblock application is authorized to access functionality" (*also a §112(f) nonce term lacking corresponding structure*)

- **'046 patent**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)
  - "wherein the function block is accessible only to the superblock application on an electronic device"
  - "wherein the service is not otherwise supported by the superblock application"
  - "and wherein the function block is accessible only to the superblock application on the electronic device"
  - "accessible only to the superblock application"
  - "establish a connection with an external device"
  - "external device"
  - "provide a service requested by the superblock application"
  - "service"
  - "without switching context from the superblock application to another application"
  - "request for the service"
  - "communication session"
  - "configured to provide functionality for the communication session" (also arguably MPF)
  - "notify the superblock application"
  - "capability to perform any rendering needed by the communication session" (also arguably MPF)
  - "authorized to access functionality provided by the function block for the service" (also arguably MPF)
  - "authorization credentials"
  - "service is available on the external device"

- "accessing a resource"
- "resource"
- "service is not otherwise supported by the superblock application"
- "operating as a bridge between the superblock application and the resource" (also arguably MPF)
- "initializing the function block when the superblock application is loaded and executed" (also arguably MPF)
- "capable of performing any rendering needed by the service" (also arguably MPF)
- "request … received as an API call"
- "resource is available"

- **'362 patent**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)
  - "receiving, by the superblock application"
  - "communicating, by the superblock application"
  - "providing, by the superblock application"
  - "creating for storage on a server"
  - "providing the rendering information"
  - "rendering information"
  - "configured for use by a computing device"
  - "communicates . . . using application programming interface (API) calls"
  - "request to initiate a real-time communication session"
  - "communication session"
  - "remote resource"
  - "notification that the real-time communication session … is being requested"
  - "identifying . . . rendering information"

- “rendering information”
- “rendering . . . the real-time communication session” (also MPF)
- “information corresponding to a mic”
- “information corresponding to a speaker”
- “information for displaying a window”
- “window”
- “video window”
- “instant message window”
- “resizable”
- “information corresponding to a camera”
- “presence indicator”
- “status of the second user”
- “API calls … compatible with an interface provided by its respective function block”
- “voice call”
- “video call”
- “adapted to operate with iOS”
- “adapted to operate with Android”
- “mobile device”
- “cellular phone”
- “smart phone”
- “tablet”
- “laptop computer”
- “desktop computer”
- “adapted to interact with a function block” (also MPF)

- "adapted to interact with one or more servers . . .to provide the real-time communication session" (also MPF)

- **'116 patent**
  - "function block" / "superblock" (also a §112(f) nonce term lacking corresponding structure)
  - "third party"
  - "authorization key"
  - "third party superblock application"
  - "additional functionality"
  - "configured to be compiled into the superblock application"
  - "configured to add the additional functionality" (also arguably MPF)
  - "real-time communication session"
  - "interact with the superblock application through … API calls"
  - "enabling establishment of the real-time communication session" (also arguably MPF)
  - "authorization key"
  - "unlocks the function block"
  - "authentication credentials"
  - "enable the function block to use the one or more servers" (also arguably MPF)
  - "sending a notification"
  - "notification requires a response"
  - "API call indicates whether the request has been granted or denied"
  - "receiving … a notification from the function block"
  - "adapted to operate with iOS"
  - "adapted to operate with Android"
  - "computing device"

- "cellular phone"
- "smart phone"
- "tablet"
- "laptop computer"
- "desktop computer"
- "enabling the establishment of the real-time communication session" (also arguably MPF)
- "interact with one or more servers … to provide the real-time communication session" (also arguably MPF)
- "notification is a request to initiate the real-time communication session"
- "response to the notification"

These claim elements are indefinite for failure to inform, with reasonable certainty, those skilled in the art about the scope of the invention as claimed. Based on Cisco's present understanding of Damaka's Infringement Contentions, at least one or more of these claim terms, phrases, and limitations are indefinite because they are inconsistent with and/or broader than the alleged invention disclosed in the specification and during prosecution, and given Damaka's apparent construction of the claims, any person of ordinary skill in the art at the time of the invention would not understand what is claimed with reasonable certainty, even when the claims are read in light of the specification and prosecution history.

VI. SUBJECT MATTER INELIGIBILITY CONTENTIONS – INELIGIBLE SUBJECT MATTER UNDER 35 U.S.C. § 101

Pursuant to this Court's Standing Order Regarding Subject-Matter Eligibility Contentions ("Standing Order") and the Court's Docket Control Order, Cisco serves Preliminary Subject Matter Ineligibility Contentions ("Ineligibility Contentions") attached hereto as **Appendix A** addressing how the Asserted Claims of the Asserted Patents are invalid.

Pursuant to paragraph (c) of the Standing Order, Cisco reserves the right to amend or supplement these Preliminary Ineligibility Contentions if: (1) Damaka amends its infringement contentions; or (2) the Court's Claim construction ruling so requires.

The information provided should not be deemed an admission regarding the scope of any claims or the proper construction of those claims or any terms contained therein. Cisco's claim construction disclosures will be provided under P.R. 4 as required by the Court's Docket Control Order. Nothing contained in these Ineligibility Contentions should be understood or deemed to be an express or implied admission or contention with respect to the absence of factual disputes relating patent ineligibility, the absence of a need for construction of any terms in an asserted claim, any proper construction of any terms in an asserted claim, or alleged infringement of that claim. There is no claim construction issue or factual issue that precludes the Court finding that the claims of the asserted patents are patent-ineligible. Nothing in these disclosures should be treated as an admission that Cisco is obligated to produce documentation not under its custody or control, or that can be obtained from some other source that is more convenient, less burdensome and/or less expensive, or for which the burden or expense outweighs its likely benefit. Cisco expressly reserves the right to revise, amend, and/or supplement their disclosures and document production should additional documentation become available.

Cisco's Ineligibility Contentions address only the Asserted Claims. Cisco reserves the right to supplement these contentions if Damaka asserts infringement of any claim other than the Asserted Claims.

Cisco's discovery and investigations in this lawsuit are ongoing, and therefore, Cisco reserves the right to revise, amend, and/or supplement these Ineligibility Contentions as discovery progresses and as they discovers additional information. Discovery is ongoing, and Cisco's prior art investigation and third-party discovery are in the initial stages. Cisco reserves the right to revise, amend, and/or supplement the information provided herein, including identifying, and relying on additional references, should Cisco further search and analysis yield additional information or references, consistent with the Local Rules, Judge Gilstrap's Standing

Order, and the Federal Rules of Civil Procedure. In particular, Cisco reserves the right to rely on, and Cisco incorporates by reference into its Ineligibility Contentions, all prior art identified by Cisco in conjunction with its Invalidity Contentions. Cisco also reserves the right to amend, modify, or supplement these Ineligibility Contentions to include prior art under 35 U.S.C. §§ 102 and 103 identified in its Invalidity Contentions. Cisco reserves the right to rely on all documents produced by Cisco, as well as Damaka, any predecessors in interest, the named inventors, and any other third parties, as discovery is ongoing.

Dated: January 15, 2026

Respectfully submitted,

By: */s/ Nima Kiaei*

Melissa R. Smith
State Bar No. 24001351
GILLAM & SMITH, LLP
303 South Washington Avenue
Marshall, Texas 75670
Telephone: (903) 934-8450
Facsimile: (903) 934-9257
Email: Melissa@gillamsmithlaw.com

Bita Rahebi
Nicholas R. Fung
Nima Kiaei (*pro hac vice*)
MORRISON & FOERSTER LLP
707 Wilshire Boulevard
Los Angeles, California 90017-3543
Telephone: (213) 892-5200
Facsimile: (213) 892-5454
Email: BRahebi@mofo.com
Email: NFung@mofo.com
Email: NKiaei@mofo.com

Brian C. Nash
Elizabeth C. Knuppel
Hunter D. Ubersox (*pro hac vice*)

MORRISON & FOERSTER LLP
300 Colorado Street, Suite 1800
Austin, TX 78701
Telephone: (512) 617-0650
Facsimile: (737) 910-0730
Email: BNash@mofo.com
Email: BKnuppel@mofo.com
Email: HUbersox@mofo.com

Attorneys for Defendant
Cisco Systems, Inc.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that all counsel of record who have appeared in this case are being served with a copy of this document via email per Local Rule CV-5(d) on January 15, 2026.

/s/ Silvia Specht
Silvia Specht