

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner

IPR2023-00553
U.S. Patent No. 9,210,254

**PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. § 312 AND 37 C.F.R. § 42.104**

TABLE OF CONTENTS

Petitioner’s Exhibit List	5
I. Introduction.....	7
II. Grounds for standing	7
III. Note.....	8
IV. Technology background	8
V. Summary of the ’254 patent	8
VI. Prosecution history	10
VII. Effective priority date of the ’254 patent	11
VIII. Level of ordinary skill in the art	11
IX. Claim construction.....	12
X. Relief requested and the reasons for the requested relief.....	12
XI. Identification of how the claims are unpatentable.....	12
A. Challenged claims	12
B. Statutory Grounds for challenges.....	13
C. Petitioner’s Reasons for Presenting Multiple Grounds.....	14
D. Ground 1: Claims 1, 2, 6-8, and 12-14 are obvious under 35 U.S.C. § 103 over Bodart, Banister, and Dharmarajan.....	15
1. Bodart.....	15
2. Banister	17
3. Dharmarajan.....	18

4.	Reasons to Combine Bodart, Banister, and Dharmarajan	19
5.	Claim 1	27
6.	Claim 2	43
7.	Claim 6	46
8.	Claim 7	46
9.	Claims 8 and 12	47
10.	Claim 13	48
11.	Claim 14	50
E.	Ground 2: Claims 1, 2, 6-8, and 12-14 are obvious under 35 U.S.C. § 103 over de Castro	50
1.	De Castro	50
2.	Claim 1	53
3.	Claim 2	69
4.	Claim 6	72
5.	Claim 7	72
6.	Claims 8 and 12	74
7.	Claim 13	74
8.	Claim 14	76
XII.	Discretionary denial is inappropriate	76
A.	No basis for 35 U.S.C. § 325(d) denial	76
B.	Discretionary denial under the <i>Fintiv</i> factors is not appropriate	76

1.	Whether evidence exists that a stay may be granted if a proceeding is instituted	77
2.	Proximity of the court’s trial date to the board’s projected statutory deadline for a final written decision	77
3.	Investment in the parallel proceeding	78
4.	Overlap between issues raised in the petition and in the parallel proceeding	79
5.	Whether the petitioner is a defendant	79
6.	Other circumstances that impact the Board’s exercise of discretion, including the merits	79
C.	Discretionary denial under <i>General Plastic</i> is not appropriate	80
XIII.	Conclusion	80
XIV.	Mandatory notices	82
A.	Real party-in-interest	82
B.	Related matters	82
C.	Lead and back-up counsel and service information	82
	Appendix of the Challenged Claims	84
	Certificate of Word Count	89
	Certificate of Service	90

PETITIONER’S EXHIBIT LIST

Ex.1001	U.S. Pat. No. 9,210,254 to Hunt
Ex.1002	Prosecution History of U.S. Pat. No. 9,210,254
Ex.1003	Declaration of Henry Houh, Ph.D., under 37 C.F.R. § 1.68
Ex.1004	<i>Curriculum Vitae</i> of Henry Houh, Ph.D.
Ex.1005	U.S. Pat. Pub. No. 2007/0043687 to Bodart <i>et al.</i>
Ex.1006	U.S. Pat. No. 6,621,892 to Banister <i>et al.</i>
Ex.1007	U.S. Pat. No. 7,089,585 to Dharmarajan
Ex.1008	U.S. Pat. Pub. No. 2012/0016932 to de Castro <i>et al.</i>
Ex.1009	Ben Langfeld, Jose de Castro, “XEP-0327: Rayo,” XMPP Standards Foundation (<i>archived</i> August 18, 2013)
Ex.1010	U.S. Pat. No. 6,298,128 to Ramey <i>et al.</i>
Ex.1011	John Letzing “Facebook Raises Hackles on Email,” Wall Street Journal Online, June 26, 2012 (<i>archived</i> June 30, 2012)
Ex.1012	Shelley O’Hara, “Easy Microsoft® Office Outlook® 2003,” (Que Publishing, September 2003)
Ex.1013	U.S. Pat. No. 8,130,751 to Berkley <i>et al.</i>
Ex.1014	U.S. Pat. No. 7,929,519 to Goldman <i>et al.</i>
Ex.1015	U.S. Pat. Pub. No. 2007/0143231 to Pegaz-Paquet <i>et al.</i>
Ex.1016	U.S. Pat. No. 1,276,825 to Swope
Ex.1017	U.S. Pat. No. 2,308,778 to Prince
Ex.1018	Affidavit of Authenticity for Ex.1009
Ex.1019	Amazon.com listing for “Easy Microsoft® Office Outlook® 2003”

	(<i>archived</i> February 1, 2008)
Ex.1020	Affidavit of Authenticity for Ex.1011 and Ex.1019.
Ex.1021	Declaration of Jose de Castro under 37 C.F.R. § 1.68
Ex.1022	U.S. Pat. No. 8,090,787 to Polis <i>et al.</i>
Ex.1023	U.S. District Courts - National Judicial Caseload Profile (June 2022), available at: https://www.uscourts.gov/statistics/table/na/federal-court-management-statistics/2022/06/30-2
Ex.1024	Docket Control Order, <i>Zilkr Cloud Technologies, LLC v. Cisco Systems, Inc.</i> , No. 2-22-cv-00166-JRG-RSP, ECF No. 31 (E.D. Tex., Oct. 13, 2022)
Ex.1025	Notice of Disclosures, <i>Zilkr Cloud Technologies, LLC v. Cisco Systems, Inc.</i> , No. 2-22-cv-00166-JRG-RSP (E.D. Tex., Sep. 22, 2022)
Ex.1026	Original Complaint for Patent Infringement, <i>Zilkr Cloud Technologies, LLC v. Cisco Systems, Inc.</i> , No. 2-22-cv-00166-JRG-RSP, ECF No. 1 (E.D. Tex., May 20, 2022)
Ex.1027	Director Katherine K. Vidal, “Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation,” June 21, 2022

I. INTRODUCTION

Pursuant to 35 U.S.C. §§ 311, 314(a), and 37 C.F.R. § 42.100, Cisco Systems, Inc. (“Petitioner”) requests that the Board review and cancel as unpatentable under 35 U.S.C. §103 (AIA) claims 1, 2, 6-8, and 12-14 (the “Challenged Claims”) of U.S. 9,210,254 (the “’254 patent,” Ex.1001).

The ’254 patent describes providing third-party telecommunications services (e.g., text messaging, voicemail etc.) to a telephone service subscriber. Ex.1001, Abstract, 3:55-58, 11:37-42. The ’254 patent assumes that each third-party service provider uses IDs, application programming interfaces (APIs), and rules specific to its ecosystem. Ex.1001, 8:36-39. Rather than offer services using IDs and APIs from each eco-system, the ’254 patent proposes a unified platform allowing service providers to integrate their ecosystems with the unified platform. Ex.1001, 1:16-19, 4:22-31, 4:60-62. The ’254 patent uses a telephone number as a user ID in its common software interface to provide access to these telecommunications services. Ex.1001, 8:40-53.

As shown below and confirmed in Dr. Henry Houh’s Declaration (Ex.1003), these concepts were well known before the ’254 patent’s earliest priority date. The challenge presented in this Petition renders obvious the Challenged Claims.

II. GROUNDS FOR STANDING

The ’254 patent is eligible for IPR and Petitioner is not barred or estopped

from requesting IPR of the Challenged Claims. 37 C.F.R. § 42.104(a).

III. NOTE

Petitioner cites to exhibits' original page numbers. **Emphasis** in quoted material has been added. Claim terms are presented in *italics*.

IV. TECHNOLOGY BACKGROUND

Dr. Henry Houh's Declaration (Ex.1003) provides a background of the relevant technology. Ex.1003, ¶¶28-33.

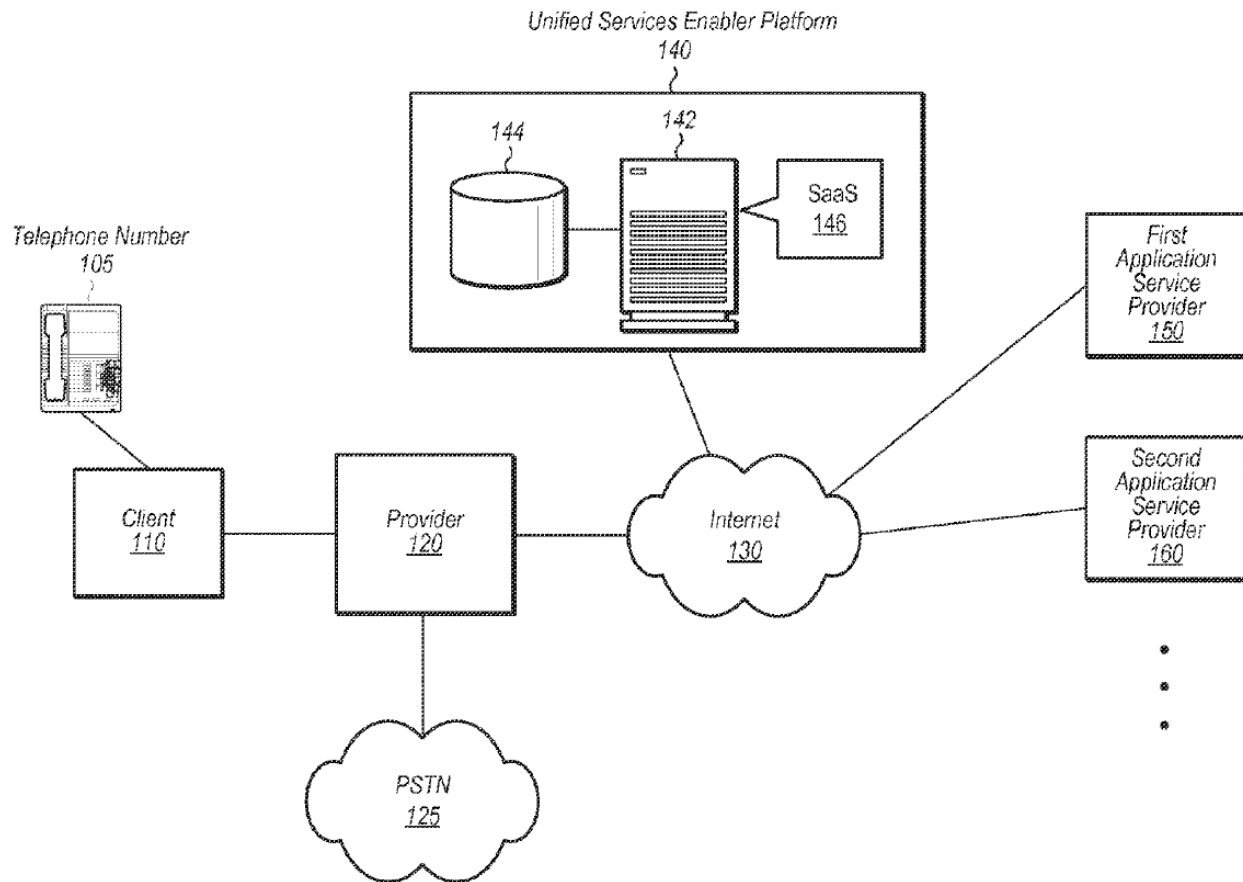
V. SUMMARY OF THE '254 PATENT

The '254 patent "relates generally to integrating communications services, and in particular to managing services and applications using telephone numbers as common subscriber identifiers." Ex.1001, 1:15-19. The '254 patent describes provisioning telecommunications services from service providers (e.g., interactive texting, voicemail) with different ecosystems using a unified software platform. Ex.1001, Abstract, 3:55-58, 11:37-42. The '254 patent explains that for "large numbers of customers...the telephone number remains the universal identifier (ID) for communicating with the outside world." Ex.1001, 1:20-26. Customers may wish to "add services and applications to their existing telephone service." Ex.1001, 1:27-33. However, third-party service providers have different "ecosystems [which] may not be compatible and may require separate IDs (for the same customer) and may have different types of rules and APIs for accessing,

provisioning, and using their various services.” Ex.1001, 4:9-13. The ’254 patent offers allegedly “new techniques for integrating advanced services and applications with legacy telephone service.” Ex.1001, 1:34-36; Ex.1003, ¶¶34-35.

The ’254 patent proposes “using the telephone number as the common subscriber ID.” Ex.1001, 1:40-42. When a customer adds a service, the customer’s “telephone number may be translated to a second ID that is utilized as a login ID or user name within the ecosystem of the second [service] provider.” Ex.1001, 8:47-49. This allegedly provides “a unified services enabler platform” that “may allow for the association of internet protocol (IP) enabled services with telephone numbers tied to legacy telephone services.” Ex.1001, 1:43-46. The customer’s telephone number is “associated with the relevant interworking data which links all of the IDs and third-party applications and services associated with the telephone number.” Ex.1001, 12:53-58; Ex.1003, ¶36.

Fig.1 below illustrates the unified services enabler platform 140 connected by internet 130 to client users 110, telephone service provider 120, and application service providers 150, 160. Ex.1001, Fig.1, 3:35-37, 4:14-28.



Ex.1001, Fig.1; Ex.1003, ¶37

As demonstrated below, it was well-known in the prior art to integrate and provision services from different ecosystems for a single user. Ex.1003, ¶38. It was also well-known in the prior art to associate a user's telephone number with user IDs from different service providers to provision services. *See, e.g.*, Ex.1006, 2:1-5; Ex.1003, ¶¶34-38.

VI. PROSECUTION HISTORY

The '254 patent claims priority to provisional application 61/888,856. The application that issued as the '254 patent was filed on November 22, 2013. The

Patent Office issued a non-final office action on May 18, 2015, rejecting the independent claims as obvious over several references and allowing a dependent claim reciting “*wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.*” Ex.1002, 41-47.

Applicant amended the independent claims to incorporate the allowed language. Ex.1002, 25-35. The ’254 patent issued on December 8, 2015. Ex.1002, 2-3.

VII. EFFECTIVE PRIORITY DATE OF THE ’254 PATENT

The ’254 patent claims priority to a provisional application filed October 9, 2013. Determining whether the ’254 patent is entitled to an earlier priority date is unnecessary in this proceeding, however, as all of the asserted prior art predates October 9, 2013. Accordingly, Petitioner has not undertaken a priority date analysis. Petitioner does not waive any right or opportunity it may have to dispute the priority date of the ’254 patent in this or another forum where appropriate.

VIII. LEVEL OF ORDINARY SKILL IN THE ART

A Person of Ordinary Skill in The Art (“POSITA”) in October 2013 would have had a working knowledge of telecommunications and networking art that is pertinent to the ’254 patent. Ex.1003, ¶¶21-22. A POSITA would have had a bachelor’s degree in electrical engineering, computer science, computer engineering, or equivalent, and two years of relevant experience in telecommunications and networking. Ex.1003, ¶22. Lack of professional

experience can be remedied by additional education, and vice versa. Ex.1003, ¶¶22-23.

IX. CLAIM CONSTRUCTION

Claim terms in IPR are construed according to their “ordinary and customary meaning” to those of skill in the art. 37 C.F.R. § 42.100(b). Petitioner submits that, for the purposes of this proceeding, the terms of the Challenged Claims should be given their ordinary and customary meaning, and no terms require specific construction. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017). Ex.1003, ¶39.

X. RELIEF REQUESTED AND THE REASONS FOR THE REQUESTED RELIEF

Petitioner asks that the Board institute an IPR trial and cancel the Challenged Claims as obvious.

XI. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

A. Challenged claims

The Challenged Claims are asserted in *Zilkr Cloud Technologies, LLC v. Cisco Systems, Inc.* No. 2-22-cv-00166 (E.D. Tex. filed May 20, 2022). A finding that the Challenged Claims are unpatentable in this proceeding will resolve the parties’ dispute and obviate the need for a trial regarding the ’254 patent in the co-

pending litigation, substantially reducing the time and expense of that litigation for all parties.

B. Statutory Grounds for challenges

Grounds	Claims	Basis
#1	1, 2-6, 12-14	35 U.S.C. § 103 over Bodart in view of Banister and Dharmarajan
#2	1, 2-6, 12-14	35 U.S.C. § 103 over de Castro

U.S. Publication 2007/0043687 (Ex.1005, “Bodart”) published on February 22, 2007.

U.S. Patent No. 6,621,892 (Ex.1006, “Banister”) issued on September 16, 2003.

U.S. Patent No. 7,089,585 (Ex.1007, “Dharmarajan”) issued on August 8, 2006.

U.S. Publication 2012/0016932 (Ex.1008, “de Castro”) published on January 19, 2012.

Bodart, Banister, Dharmarajan, and de Castro are prior art under 35 U.S.C. § 102(a)(1).¹

¹ Petitioner notes that the “Swear Behind” provisions of 37 C.F.R. § 1.131 do not apply to the ’254 patent. *See* MPEP §§ 715, 2159.02.

Petitioner’s obviousness Grounds rely on the combined teachings of the references as understood by a POSITA and not on a physical incorporation of elements from one reference into another. *See In re Mouttet*, 686 F.3d 1322, 1332 (Fed. Cir. 2012); Ex.1003, ¶¶40-42.

Petitioner cites to additional prior art below as evidence of the background knowledge of a POSITA and to provide contemporaneous context to support Petitioner’s assertions regarding what a POSITA would have understood from the asserted prior art. Ex.1009 is an XMPP Specification document referenced by de Castro as the Rayo Specification. Its prior art status is confirmed by Ex.1018 and Ex.1021. Ex.1011 is an article in the Wall Street Journal. Its prior art status is confirmed by Ex.1020. Ex.1012 is a user manual for Microsoft Outlook. Its prior art status is confirmed by Ex.1019 and Ex.1020. The Petition’s use of supporting tertiary evidence has been repeatedly approved. *See Yeda Research v. Mylan Pharm. Inc.*, 906 F.3d 1031, 1041-1042 (Fed. Cir. 2018) (affirming the use of “supporting evidence relied upon to support the challenge”); 37 C.F.R. § 42.104(b); *see also K/S HIMPP v. Hear-Wear Techs., LLC*, 751 F.3d 1362, 1365-66 (Fed. Cir. 2014); *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1363 (Fed. Cir. 2016).

C. Petitioner’s Reasons for Presenting Multiple Grounds

The Petition provides two obviousness Grounds based on non-overlapping

combinations of references. The two Grounds demonstrate the obviousness of the Challenged Claims in different technical contexts. Ground 1 relies on prior art describing higher-level, front-end functionality in a system that is used and configured by the end user. Ground 2 relies on prior art describing lower-level networking logistics in an application server environment.

D. Ground 1: Claims 1, 2, 6-8, and 12-14 are obvious under 35 U.S.C. § 103 over Bodart, Banister, and Dharmarajan

1. Bodart

Bodart relates to “the integration of personal and public information to enable more productive individuals by automating routine tasks.” Ex.1005, [0001]. One task is “having a user’s email read to her by phone.” Ex.1005, [0002]. Bodart teaches a virtual assistant having “access to a user’s contacts, calendar, and location,” as well as “information about the weather, traffic, and mass transit...” Ex.1005, Abstract. The virtual assistant includes “one or more user interfaces 140 which allow a user to interact with virtual assistant,” including “cell phone 145” and “conventional telephone 146.” Ex.1005, [0030]. “[A] remote user can call into his or her virtual assistant and provide voice commands.” Ex.1005, [0026]; Ex.1003, ¶¶43-44.

Further, the virtual assistant “may include the ability to receive and direct calls and messages directed to user 201.” Ex.1005, [0044]. “[I]ncoming messages

including instant messages and/or SMS messages, and even email messages, may similarly be routed using message routing 214.” Ex.1005, [0044]. “[A] user may be able to send messages using an instant message client....” Ex.1005, [0034].

Additionally, a “personal information manager (PIM),” such as Microsoft Outlook, “may form the basis of user data to be used by virtual assistant.” Ex.1005, [0040].

The PIM data may “be managed by individual applications (e.g., iCal ... and Mail on an Apple Computer).” Ex.1005, [0040]. The virtual assistant also accesses information sources, schedules appointments, retrieves contact information, and sets alarms. Ex.1005, [0027], [0029], [0038], [0041], [0045]; Ex.1003, ¶¶45-46.

Bodart is analogous art to the ’254 patent and is “from the same field of endeavor,” namely, integrating communications services. *In re Bigio*, 381 F.3d 1320, 1325 (Fed. Cir. 2004); Ex.1001, 1:16-18; Ex.1003, ¶47. Bodart describes a system “for receiving data from a plurality of information providers.” Ex.1005, [0005]. The system’s information sources and functionalities are accessible to the user by telephone. Ex.1005, [0020], [0030]. Similarly, embodiments of the ’254 patent allow a user to “activate many different services and applications from multiple different providers, and these different services may be activated and associated with the user’s existing telephone number.” Ex.1001, 11:18-22. Both Bodart and the ’254 patent provide users access to internet-based communications services such as voicemail, text messaging, and voice attendants. Ex.1001, 4:14-

15, 24-28, 11:37-40; Ex.1005, Abstract, [0022], [0025], [0026], [0044], [0064], [0072]; Ex.1003, ¶47. Bodart is also reasonably pertinent to the particular problem of the '254 patent, which is provisioning communication services from different providers to a user of telephone service. Ex.1001, 11:33-42; Ex.1005, [0005], [0030]); Ex.1003, ¶47.

2. Banister

Banister teaches a method for “converting an electronic mail message to audio for delivery via a telephone call.” Ex.1006, 1:8-10. Banister’s method includes “retrieving a telephone number associated with an intended recipient of the message, ... establishing a telephonic connection with the recipient and playing an audio form of the message text.” Ex.1006, 1:51-57; Ex.1003, ¶¶48-50.

Banister is analogous art to the '254 patent and is from the same field of endeavor, namely integrating communications services. Ex.1001, 1:14-17; Ex.1003, ¶51. Banister describes systems making email messages accessible via telephone. Ex.1006, Abstract. Similarly, embodiments of the '254 patent allow a user to “activate many different services and applications from multiple different providers, and these different services may be activated and associated with the user’s existing telephone number.” Ex.1001, 11:18-21, 33-42; Ex.1003, ¶51. Both Banister and the '254 patent describe storing a mapping between an email address to a telephone number. Ex.1001, 7:51-55, 9:55-58; Ex.1006, Abstract, 2:1-5;

Ex.1003, ¶51. Banister is also reasonably pertinent to the particular problem of the '254 patent, that is, to associate an electronic ID with a telephone number to provision a communication service to the user. Ex.1001, 11:33-42; Ex.1006, 2:1-5, 4:14-17; Ex.1003, ¶51.

3. Dharmarajan

Dharmarajan describes providing “instant messaging” to a user with an “instant messaging client application program 12 communicat[ing] through the Internet 8 with an instant message server application 4.” Ex.1007, 4:43-52.

Dharmarajan explains, akin to the '254 patent, that a “login procedure may include the transmission of a username and password from the client computer to the instant message server computer in a secure fashion.” Ex.1007, 9:47-51; Ex.1003, ¶¶52-54.

Dharmarajan is analogous art to the '254 patent and is from the same field of endeavor, namely, integrating communications services. Ex.1001, 1:15-18; Ex.1003, ¶55. Dharmarajan relates to “a method and system for authorizing a client computer to access a server computer system.” Ex.1007, 1:8-12. It provides details on logging into an instant messaging (IM) server computer by providing a username and password. Ex.1007, 1:32-40. Analogously, the embodiments of the '254 patent describe using “a login ID or user name” to activate services. Ex.1001, 8:44-49, 8:65-9:5; Ex.1003, ¶55. Both the '254 patent and Dharmarajan describe

authenticating customers for communication services using an email address.

Ex.1001, 13:44-66; Ex.1007, 5:39-49; Ex.1003, ¶55. Dharmarajan is also reasonably pertinent to the particular problem of the '254 patent, which is to allow a user to request access to third-party communication services associated with a user ID. Ex.1001, 13:53-64; Ex.1007, 2:19-23; Ex.1003, ¶55.

4. **Reasons to Combine Bodart, Banister, and Dharmarajan**

A POSITA considering the teachings of Bodart would have also considered the teachings of Banister and Dharmarajan, as they are analogous and pertinent to the same field of endeavor. Ex.1005 [0028], [0030]; Ex.1006, 1:7-8; Ex.1007, 4:8-13; Ex.1003, ¶¶47, 51, 55-56. Each of these references describes providing users access to telephony and internet-based communications services such as IM and email. Ex.1005, Abstract, [0025], [0064], [0072]; Ex.1006, 1:8-11; Ex.1007, 4:43-55; Ex.1003, ¶56. A POSITA would have been motivated to combine the references for at least the reasons discussed below and would have had a reasonable expectation of success in the combination. Ex.1003, ¶56.

*a) **Associating a telephone number with an email address***

Bodart describes “having a user’s email read to her by phone.” Ex.1005, [0002]. To facilitate such a feature, Bodart’s virtual assistant “**intercept[s]** incoming telephone calls, instant messages, and other real time **communication requests.**” Ex.1005, [0025]. However, Bodart provides limited details about how

the virtual assistant “intercepts” and reads emails to the user by telephone.

Ex.1003, ¶¶57-59.

Banister provides details on how an electronic mail server converts an email into audio and delivers the audio to the email recipient by telephone. Ex.1006, Abstract, 3:35-50, 4:14-17. Banister explains the email recipient’s “address specified by an originator of a message may be converted to a telephone number or used to look-up [the recipient’s] telephone number when the message is received at electronic mail server.” Ex.1006, 4:14-17. To facilitate this look-up, the “mail server may maintain a list or database of **addresses with associated telephone numbers.**” Ex.1006, 2:1-5; Ex.1003, ¶¶60-61.

A POSITA implementing Bodart’s email-reading functionality would have found it obvious to consider other references like Banister which provides implementation details that Bodart expects. Ex.1005, [0002]; Ex.1006, Abstract 3:35-50; Ex.1003, ¶62. For instance, Banister explains that reading emails via telephone involves looking up a telephone number from an email address. Ex.1006, 2:1-5. Bodart describes storing, in a user data module, “contact email addresses and phone numbers,” as well as “user accounts, including email passwords, website passwords, and so forth.” Ex.1005, [0021], [0022]. In view of Banister’s disclosure of maintaining lists of email addresses with associated telephone numbers, it would have been obvious to a POSITA for Bodart’s user

data module to store an association between the user's email address and telephone number. Ex.1003, ¶62.

The combination of Bodart and Banister would be simply combining prior art elements according to known methods (computer programming) to yield predictable results (associating an email address with a telephone number in a database). Ex.1003, ¶63. The results of the combination would have been predictable because both Bodart and Banister describe reading emails to a user by phone, and because Banister merely provides implementation details that Bodart expects. *See* Ex.1005, [0002]; Ex.1006, Abstract; Ex.1003, ¶¶62-63. Furthermore, storing telephone numbers and electronic addresses in databases was well-known in the art. Ex.1003, ¶64; Ex.1013, 6:55 – 7:3; Ex.1014, 11:8-17; Ex.1015, [0116]. The proposed combination would have been predictable and a POSITA would have a reasonable expectation of success, because the combination of Bodart's email-by-telephone functionality with Banister's implementation details would have been well-within the skill level of a POSITA. Ex.1003, ¶64. Bodart itself evidences this expectation of success by presuming the implementation details would have been known to those of skill in the art. "[A] patent need not teach, and preferably omits, what is well known in the art." *Hybritech Inc. v. Monoclonal Antibodies, Inc.*, 802 F.2d 1367, 1384 (Fed. Cir. 1986). For example, a POSITA would have been capable of implementing software changes to enable Bodart's

virtual assistant to associate a telephone number with an email address in the user data module. Ex.1006, 2:1-5, 59-64; Ex.1013, 6:55 – 7:3; Ex.1014, 11:8-17; Ex.1015, [0116]; Ex.1003, ¶64. This confirms the predictability and reasonable expectation of success of the combination. *See* Ex.1003, ¶64.

b) Associating a telephone number with an IM username

Bodart discloses that the virtual assistant “intercept[s] incoming telephone calls, **instant messages**, and other real time communication requests.” Ex.1005, [0025]. In response to an incoming call, Bodart’s “virtual assistant 101 may put contact 215 on hold and attempt to reach user 201 via ... **instant message**.” Ex.1005, [0060]. However, Bodart provides limited details about how this is done. Ex.1003, ¶¶65-66.

Banister describes techniques for looking up telephone numbers associated with electronic addresses. Ex.1006, 3:35-50. For example, “[an] electronic mail server is configured to receive a mail message and parse an address to retrieve a telephone number included in or associated with said address.” Ex.1006, Abstract. To facilitate this telephone number look-up, the “mail server may maintain a list or database of addresses with associated telephone numbers.” Ex.1006, 2:1-5; Ex.1003, ¶¶67-69.

A POSITA implementing Bodart’s IM functionality to notify a user of an incoming call would have found it obvious to consider other references like

Banister which provides implementation details for associating a recipient's phone number with an electronic address. Ex.1003, ¶70. For instance, Bodart describes storing, in a user data module, “**user accounts**, including email passwords, website passwords, and so forth.” Ex.1005, [0022]. Bodart also describes storing “contact email addresses and phone numbers.” Ex.1005, [0021]. In view of Banister's disclosure of storing addresses with associated telephone numbers, it would have been obvious to a POSITA for Bodart's user data module to store an association between the user's IM username and telephone number. Ex.1003, ¶¶70-71.

The combination of Bodart's user data module and Banister's stored telephone numbers and associated addresses would be simply combining prior art elements according to known methods (computer programming) to yield predictable results. Ex.1003, ¶71. The result of the combination would be a stored association between IM usernames and telephone numbers in the user data module. Ex.1003, ¶71. This combination would have been predictable given that both Bodart and Banister describe storing contact addresses and telephone numbers. *See* Ex.1005, [0002]; Ex.1006, Abstract; Ex.1003, ¶71. Furthermore, storing account information including different types of addresses was well-known in the art. *See* Ex.1005, [0022]; Ex.1003, ¶71. Therefore, the proposed combination would have been predictable, and a POSITA would have a reasonable expectation of success because the combination of Bodart's IM functionality with Banister's

implementation would have been within the skill level of a POSITA. Ex.1003, ¶71.

A POSITA would have been capable of implementing software changes to enable Bodart’s virtual assistant to associate a telephone number with an IM address in the user data module. Ex.1003, ¶71. For example, the software change would involve creating a table or set of tables in the user data module that includes electronic addresses associated with a telephone number. Ex.1003, ¶71. This confirms the predictability and reasonable expectation of success of the combination. *See* Ex.1006, 2:1-5, 59-64; Ex.1013, 6:55 – 7:3; Ex.1014, 11:8-17; Ex.1015, [0116]; Ex.1003, ¶71.

c) Logging into an IM service

Bodart describes “message & call routing” functionality that includes both “intercept[ing]... instant messages” and “attempt[ing] to reach user 201 via ... instant message.” Ex.1005, [0025], [0060]. A user can also interact with Bodart’s virtual assistant “using an instant message client....” Ex.1005, [0034]; Ex.1003, ¶¶72-74. However, Bodart provides limited details about how the virtual assistant accesses IM services. Ex.1003, ¶75.

Dharmarajan describes “secure login mechanism[s] to gain access to [an] instant messaging server application program.” Ex.1007, 5:4-7. Dharmarajan discloses “when a user launches the instant messaging client, the user may be required to provide a **username and password**.” Ex.1007, 1:33-35; Ex.1003, ¶76.

A POSITA implementing the IM functionality of Bodart's virtual assistant would have found it obvious to consider other references like Dharmarajan for implementation details about accessing an IM service. Ex.1003, ¶77. For instance, Dharmarajan teaches that a user provides a username and password when launching an IM application. Ex.1007, 1:33-35. Bodart teaches storing user accounts, including "email passwords [and] website passwords." Ex.1005, [0022]. It would have been obvious to a POSITA to store an IM username and password in Bodart's user data module, which the virtual assistant would then use when accessing an IM service. Ex.1005, [0034]; Ex.1007, 2:22-26; Ex.1003, ¶77.

The combination of Bodart and Dharmarajan would be simply combining prior art elements (Bodart's virtual assistant and Dharmarajan's IM login procedure) according to known methods (computer programming) to yield predictable results (the virtual assistant accessing an IM service). Ex.1003, ¶78. A POSITA implementing the combination would have had a reasonable expectation of success. Bodart already describes storing and using login credentials to access online services. Ex.1005, [0022], [0042]; Ex.1003, ¶78. Further, both Bodart and Dharmarajan describe using the same software for IM service, such as MSN Messenger. *See* Ex.1005, [0034]; Ex.1007, 4:52-55; Ex.1003, ¶78. Combining Bodart's IM functionality with Dharmarajan's implementation for logging into an IM service would have involved simple software changes easily accomplished by a

POSITA. Ex.1003, ¶78. The software change would involve programming the virtual assistant to login to the IM server computer with an IM username and password retrieved from the user data module. Ex.1003, ¶78.

* * *

Bodart contemplates a system that combines different communication types, such as reading emails by telephone, and sending an IM to notify the user about an incoming telephone call. Ex.1005, [0002], [0060]; Ex.1003, ¶57. However, it provides limited details on how to implement these functionalities, which involve combinations of telephone, email, and IM services. Ex.1003, ¶¶57-59, 65-66, 75. As described above, Banister and Dharmarajan provide implementation details for these functionalities that Bodart expects. Ex.1003, ¶¶62-63, 69, 76-77.

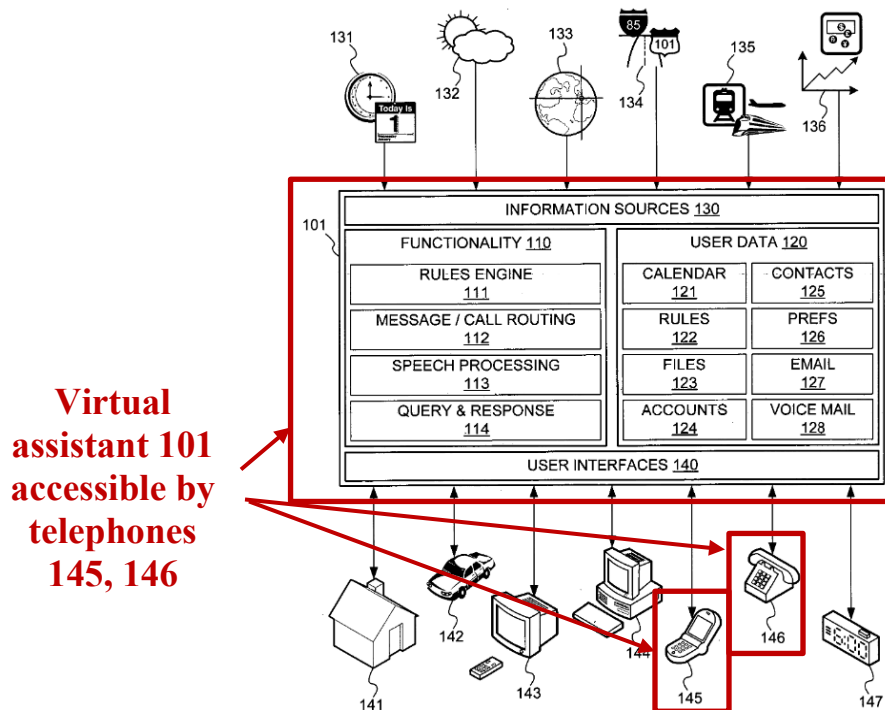
Specifically, Banister provides details for associating telephone numbers and electronic addresses. Ex.1006, 2:1-5; 4:14-18; Ex.1003, ¶60. And Dharmarajan describes details for logging into and accessing electronic communications services like email and IM. Ex.1007, 1:33-38; Ex.1003, ¶76. An exemplary combination of Bodart and Banister would be to configure the virtual assistant with a table or set of tables in the user data module that associate the user's telephone number with the user's electronic addresses. Ex.1003, ¶¶56, 63. Further combining Dharmarajan with Bodart and Banister would include configuring the virtual assistant to login to an IM service or email service using login credentials (e.g., username, password)

stored in the user data module. Ex.1003, ¶¶56, 77. This combination would have been within the skill set of a POSITA, and a POSITA would have had a reasonable expectation of success in the combination. Ex.1003, ¶56. Indeed, a POSITA would have appreciated that other implementations of the combination are possible, and that the specific implementation details would be a design choice. Ex.1003, ¶56.

5. Claim 1

[1.0] *A telephone system, comprising:*

Bodart teaches a virtual assistant 101 (“*telephone system*”) with communication functionalities “including, for example, the ability to initiate **telephone calls** and fax calls.” Ex.1005, [0027]. Fig.1 of Bodart, reproduced below, illustrates the virtual assistant 101 accessible by “cell phone 145” and “conventional telephone 146.” Ex.1005, [0030]; Ex.1003, ¶¶80-81.



Ex.1005, Fig.1 (annotated); Ex.1003, ¶80.

Thus, Bodart teaches the preamble. Ex.1003, ¶¶79-83. As demonstrated below, the combination of Bodart, Banister, and Dharmarajan discloses a system “comprising” the recited elements.

[1.1] a database; and

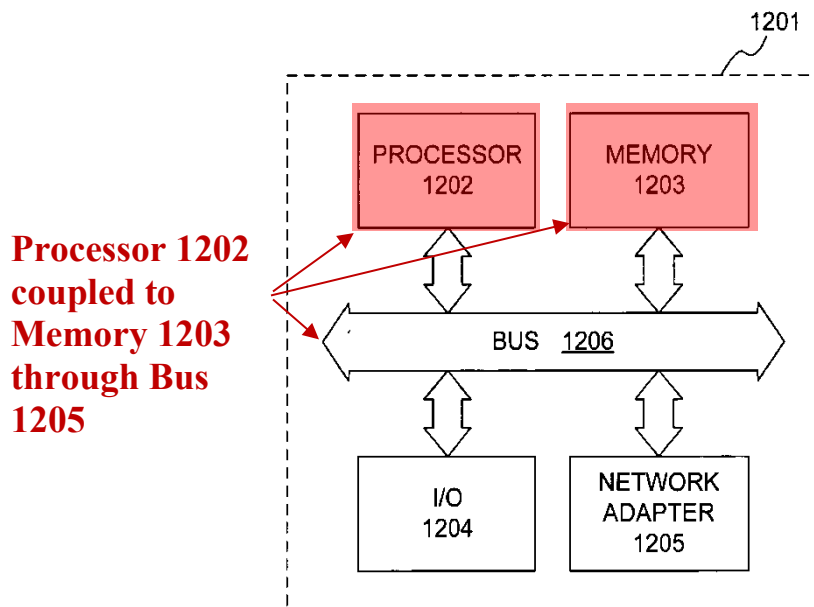
Bodart teaches that a “**user data module may store data** about a specific user,” such as “calendar appointment information, contact information, rules information, and preferences information.” Ex.1005, [0005]. “Virtual assistant 101 software further may include the ability to access **user data 120.**” Ex.1005, [0020]. Bodart further discloses looking up “phone numbers of contacts in the user’s **database.**” Ex.1005, [0057]. It would have been obvious to a POSITA that the user

data module stores contact information in a *database* which is used by the virtual assistant to look up phone numbers of contacts. Ex.1003, ¶¶84-86.

This is confirmed by Banister, which discloses storing “addresses with associated telephone numbers” in a “database.” Ex.1006, 2:1-5; Ex.1003, ¶86. Thus Bodart and Banister render obvious the virtual assistant (“*telephone system*”) comprising “*a database*.” Ex.1003, ¶¶84-87. As explained above, a POSITA would have been motivated to combine Bodart’s teaching of a virtual assistant and a database with Banister’s implementation details of storing various user information in a database. Section XI.D.4.b; Ex.1003, ¶¶28-30, 62-64, 69-71.

[1.2] *a server coupled to the database,*

Bodart’s virtual assistant is implemented on device 1201 illustrated in Fig.12 below. Ex.1005, [0075].



Ex.1005, Fig.12 (annotated); Ex.1003, ¶¶91.

In light of the disclosed structure of Bodart’s device 1201, it would have been obvious for the device 1201 to be a server. Ex.1003, ¶¶92-94. For example, Bodart teaches that “[d]evice 1201 may be any sort of hardware minimally containing the components shown here.... This includes desktop computers, laptop computers, **servers...** and so forth.” Ex.1005, [0075]; Ex.1003, ¶¶89-93.

As discussed above, the virtual assistant (“*telephone system*”) accesses user data, which it would have been obvious to store in a *database*. Section XI.D.5.[1.1]; Ex.1003, ¶86. Because Bodart describes that the virtual assistant may be implemented on the device 1201 (which may be a server), it would have been obvious to a POSITA for Bodart’s virtual assistant to be a server that is “*coupled to the database.*” Ex.1005, [0020], [0075]; Ex.1003, ¶¶94-96.

[1.3] *wherein the server is utilized by a first provider to provide a first service to a first user, wherein the first service is telephone service, and wherein the telephone service is associated with a first telephone number;*

First, Bodart teaches that “[v]irtual assistant 101 may include the ability to **receive and direct calls** and messages directed to **user 201**.” Ex.1005, [0044].

Thus, the virtual assistant 101, which includes “*a server*” as explained in [1.2], is “*utilized*” to provide “*telephone service*” (e.g., receiving and directing telephone calls) to the user 201 (“*a first user*”). Ex.1005, [0025], [0044]; Ex.1003, ¶¶97-100.

Second, a POSITA would have found it obvious that the telephone service would be provided to the user by a telephone service provider (“*a first provider*”). Ex.1016, p.2, col.2, ll.80-84; Ex.1017, p.1, col. 2, ll.4-6; Ex.1003, ¶101. For example, a POSITA would have been familiar with telephone carrier companies such as AT&T, Verizon, and Sprint, which were well-known telephone service providers. Ex.1015, [0106]; Ex.1003, ¶101.

Third, Bodart’s disclosure of the user accessing the virtual assistant by telephone renders obvious that the “*telephone service is associated with a first telephone number.*” Ex.1005, [0026]; Ex.1003, ¶¶102-104. For instance, Bodart’s disclosure that the “user can call into his or her virtual assistant” renders obvious the virtual assistant having a telephone number. Ex.1005, [0026]; Ex.1003, ¶¶102-103. Further, Bodart’s disclosure that “[v]irtual assistant 101 may include the ability to **receive and direct calls** and messages directed to **user 201**” renders

obvious the user having a telephone number. Ex.1005, [0044]; Ex.1003, ¶¶102-103. Thus, the telephone service used to access the virtual assistant is “*associated with*” a telephone number for the user, a telephone number for the virtual assistant, or both. Ex.1005, [0026], [0044]; Ex.1003, ¶103.

[1.4] *wherein the server is configured to: receive a request to provision a second service for the first user,*

The prior art renders obvious this limitation in two ways, with the “*second service*” being either an IM service or an email service. Ex.1003, ¶¶105-106, 107-111 (IM), 112-114 (email), 115.

IM example: Bodart teaches that “[t]he virtual assistant also has a rules engine enabling a user to create rules for **handling** incoming calls and **instant messages**.” Ex.1005, Abstract. Further, “a user may be able to **send messages** using an instant message client.” Ex.1005, [0025]. The handling of IMs (*e.g.*, through an IM client) teaches “*a second service*.” Ex.1003, ¶¶107-109.

Dharmarajan teaches that “when a user launches [an] instant messaging client, the user may be required to **provide a username and password**.” Ex.1007, 1:33-35. Configuring Bodart’s virtual assistant by providing the username and password for accessing the IM service, per Dharmarajan, renders obvious a “*request to provision [the] second service for the first user*.” Ex.1007, 1:33-35; Ex.1003, ¶110.

Bodart's disclosure of IM functionality in combination with Dharmarajan's disclosure of providing an IM username and password renders obvious "*wherein the server is configured to: receive a request to provision a second service for the first user.*" Ex.1005, [0025]; Ex.1007, 5:24-38; Ex.1003, ¶¶110-111. For example, Bodart discloses storing "information for ... **user accounts**, including email passwords, website passwords, and so forth." Ex.1005, [0022]. As discussed above, it would have been obvious to a POSITA that the user would configure the virtual assistant's user data module with an IM username and password. Section XI.D.4.c; Ex.1005, [0034]; Ex.1007, 5:24-38; Ex.1003, ¶¶65-71. It would have also been obvious for the user to configure the user data module with a preferred IM platform. Ex.1003, ¶110. The user configuring the virtual assistant with a preferred IM application or platform, an IM username, and a password to enable the virtual assistant to login to the IM service is a "*request to provision [the] second service.*" Ex.1005, [0022]; Ex.1003, ¶¶32, 110. Bodart's virtual assistant would "*receive [the] request to provision [the] second service,*" because the virtual assistant would store the preferred IM application or platform, username, and password provided by the user. Ex.1003, ¶¶110-111.

Email example: In an alternative example, Bodart teaches that a user, through the virtual assistant, can "search user files 123, **emails 127**, calendar 121 appointments, and so forth." Ex.1005, [0027]. Ex.1003, ¶112. To access email

service, Bodart teaches that “[a]ccounts 124 may also include ... **email** passwords.” Ex.1005, [0022]. Bodart further explains that the virtual assistant may rely on a personal information manager (PIM) such as “**Microsoft Outlook**” to access user data. Ex.1005, [0040]; Ex.1003, ¶¶112-113.

Bodart’s teaching that the virtual assistant may be configured to access emails (“*a second service*”) and store email account information (*e.g.*, email passwords) renders obvious “*wherein the server is configured to: receive a request to provision a second service for the first user.*” Ex.1003, ¶114. For example, similar to the IM service, it would have been obvious to a POSITA that the user would configure the virtual assistant’s user data module with email server settings (protocol and server name or server address) an email address and password for accessing email (“*second service*”). Ex.1003, ¶¶31, 114. The user configuring the user data module with the email account address and password, as well as the server protocol and server address, to the virtual assistant to enable the virtual assistant to login to the email service is a “*request to provision [the] second service.*” Ex.1003, ¶¶31, 114. In this way, the virtual assistant would login to an email server for email access. Ex.1005, [0022]; Ex.1003, ¶¶31, 114. Bodart’s virtual assistant would “*receive [the] request to provision [the] second service,*” and store the email address and password provided by the user. Ex.1003, ¶114.

[1.5] *wherein the second service is provided by a second provider, and wherein*

the second provider is different than the first provider;

IM example: Bodart teaches that “a user may be able to send messages using an instant message client such as America Online (AOL) Instant Messenger, Microsoft Network (MSN) Messenger, and so forth.” Ex.1005, [0034]. Thus, Bodart teaches multiple examples of “*a second provider*” (such as AOL or Microsoft) providing IM service (“*second service*”). Ex.1005, [0034]; Ex.1003, ¶¶117-118. Further, a POSITA would have found it obvious that AOL or Microsoft would be different from the telephone service provider (“*first provider*”). Ex.1003, ¶119. Therefore, Bodart renders obvious that the IM service is “*provided by a second provider, wherein the second provider is different than the first provider.*” Ex.1003, ¶¶116, 119.

Email example: Alternatively, in reference to email service, Bodart teaches using PIMs such as “Microsoft Outlook.” Ex.1005, [0040]; Ex.1003, ¶¶120-121. It would have been obvious to a POSITA that Outlook would provide the email service, or alternatively, would access a well-known email service such as Gmail or Hotmail. Ex.1005, [0036], [0040]; Ex.1007, 5:28-31; Ex.1011, pp.1-2; Ex.1003, ¶122. It would also have been obvious to a POSITA that the email service provider (e.g., Gmail) would be “*different than*” the telephone service provider (“*the first provider*”). Ex.1005, [0040]; Ex.1003, ¶122.

[1.6] responsive to receiving the request: create a mapping from the first

telephone number to a second identifier (ID), wherein the second ID is utilized by the second provider to identify the first user;

IM example: As discussed at Section XI.D.5.[1.4], Bodart and Dharmarajan render obvious a user providing an IM username and password into the user data module. Ex.1003, ¶¶107-111. The IM username (*see* Ex.1007, 1:33-35) renders obvious a “*second ID [] utilized by the second provider to identify the first user.*” Ex.1003, ¶¶123-124.

Bodart describes functionalities of the virtual assistant that combine telephone service associated with a telephone number, and a second communication service associated with a different electronic ID. Bodart teaches that, in response to an incoming phone call, the “virtual assistant 101 may **put [the call] on hold** and attempt to reach user 201 via phone or **instant message.**” Ex.1005, [0060]. Bodart further teaches that the user data module may store user account information “including email passwords, website passwords, and so forth.” Ex.1005, [0022]; Ex.1003, ¶¶125-126.

Bodart teaches the server sending an IM to a user receiving a phone call, but provides limited details on how this is done. Ex.1003, ¶¶126-127. Banister and Dharmarajan provide implementation details for providing access to electronic communications by phone. Ex.1006, Abstract; Ex.1007, 1:33-38; Ex.1003, ¶127. For instance, Banister describes “converting a textual portion of an electronic mail

message into audio and delivering the audio through a telephone call to a recipient.” Ex.1006, Abstract. Banister explains that “an [email] **address** specified by an originator of a message may be ... used to **look-up a telephone number.**” Ex.1006, 4:14-17. To facilitate the look-up, a “mail server may maintain a list or **database of addresses with associated telephone numbers.**” Ex.1006, 2:1-5; *see also*, 7:8-10; Ex.1003, ¶127. Thus, Banister teaches “*a mapping from the first telephone number to*” an email address. Ex.1003, ¶127. While Banister primarily describes email messages, Dharmarajan explains that logging into an IM service involves an IM username and password. Ex.1007, 1:33-38; Ex.1003, ¶¶31-32, 128.

As discussed above in Section XI.D.4, a POSITA implementing Bodart’s IM functionalities for notifying a user of an incoming call would have found it obvious to consider other references like Banister and Dharmarajan, which simply provide the implementation details that Bodart expects. Ex.1003, ¶128. In this regard, it would have been obvious to incorporate Banister’s database of telephone numbers mapped to electronic addresses to facilitate Bodart’s functionalities that involve two different types of communication identifiers (e.g., IM username and telephone number). Ex.1005, [0002]; Banister, Abstract; Dharmarajan, 1:33-38; Ex.1003, ¶128. The mapping between the user’s telephone number and IM username would facilitate the virtual assistant notifying the user by IM about the incoming call. Ex.1005, [0060]; Ex.1003, ¶128. This renders obvious “*create a mapping from the*

first telephone number to a second identifier (ID)." Ex.1003, ¶128. Further, it would be obvious to a POSITA for the virtual assistant to create the mapping between the telephone number and the IM username when the user configures the user data module with the IM login credentials (*"responsive to receiving the request"*). Ex.1003, ¶¶28-30, 129.

Email example: Bodart's teachings of storing email passwords in the user data module also renders obvious *"receive a request to provision a second service for the first user."* See Section XI.D.5.[1.4]; Ex.1003, ¶¶130-132. It would have been obvious to a POSITA that the user would provide an email address associated with the email password. Ex.1003, ¶131. The stored email address in the user account information is the *"second ID [] utilized by the second provider to identify the first user."* Ex.1003, ¶¶130-131.

Further, Bodart contemplates a communication method for **"having a user's email read to her by phone."** Ex.1005, [0002]. Banister teaches such a technique, including a method *"for converting a textual portion of an electronic mail message into audio and delivering the audio through a telephone call to a recipient."*

Ex.1006, Abstract; Ex.1003, ¶131. Banister explains that *"an address specified by an originator of a message may be ... used to **look-up a telephone number** when the message is received at electronic mail server 104."* Ex.1006, 4:14-17. To facilitate the look-up, a *"mail server may maintain a list or **database of addresses***

with associated telephone numbers.” Ex.1006, 2:1-5. Thus, Banister teaches using “*a mapping from the first telephone number to*” the email address. Ex.1003, ¶131.

As discussed above in Section XI.D.4, it would have been obvious to a POSITA in view of Banister for Bodart’s virtual assistant to “*create a mapping from*” the telephone number to the email address when the user configures the virtual assistant with email account information (“*in response to the request*”). Ex.1003, ¶¶131-132. Storing directories of telephone numbers and associated email addresses and other addresses was well-known. *See* Ex.1010, Fig.2; Ex.1012, pp. 112-118; Ex.1006, 2:1-5; Ex.1003, ¶64. The mapping would enable the virtual assistant to read emails to the user by phone, as explained above. Ex.1005, [0060]; Ex.1003, ¶133; Section XI.D.4.a-b. In this way, when the virtual assistant receives an email addressed to the user, the virtual assistant would look up the telephone number associated with the email address in the user data module. The virtual assistant would call the user at the telephone number to read the email to the user by phone. Ex.1003, ¶133.

Thus, Bodart, Banister, and Dharmarajan render obvious this limitation. Ex.1003, ¶¶123-134.

[1.7] [*responsive to receiving the request:*] activate the second service with the second provider utilizing the second ID as a user ID for the first user; and

As discussed at Section XI.D.5.[1.4], Bodart teaches email and IM services, either of which corresponds to the claimed “*second service*.” Ex.1003, ¶106.

IM example: Dharmarajan’s disclosure of the IM username (*see* Ex.1007, 1:33-35) teaches the “*second ID*.” *See* Section XI.D.5.[1.6]; Ex.1003, ¶124. Dharmarajan explains that a user may provide an IM username and password “to gain access to the server computer that provides the instant messaging service.” Ex.1007, 1:33-38; Ex.1003, ¶136.

It would have been obvious to a POSITA that Bodart’s virtual assistant would login to an IM service (“*activate the second service*”) when the user configures the IM username and password (“*responsive to receiving the request*”), for example, to confirm that the username and password are correct. Ex.1003, ¶¶31-32, 137. Gaining access to the IM service would be an obvious part of the virtual assistant “handling incoming... instant messages” and “send[ing] messages using an instant message client.” Ex.1005, Abstract, [0034]; Ex.1007, 9:57-61; Ex.1003, ¶¶137-138. It would also have been obvious that the IM service provider would utilize the IM username as “*a user ID*” for the user. Ex.1005, [0034]; Ex.1003, ¶¶135, 138.

Email example: The stored email address in the account information is the “*second ID*.” Section XI.D.5.[1.6]. Similar to the IM example, it would have been obvious to a POSITA that Bodart’s virtual assistant would login to an email service

(“*activate the second service*”). Ex.1003, ¶139. It would also have been obvious to a POSITA that the email service provider would utilize an email address as “*a user ID*” for the user because that is how email services commonly worked. Ex.1005, [0040]; Ex.1003, ¶140. It would have been obvious for the virtual assistant to login to the email server when the user provides their email address and password into the user data module (“*responsive to receiving the request*”) to verify the login credentials. Ex.1003, ¶¶31-32, 135, 140-141.

[1.8] [*responsive to receiving the request:*] associate the second service with the first telephone number, wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.

Bodart teaches email and IM services, either of which corresponds to the claimed “*second service*.” Section XI.D.5.[1.4]; Ex.1003, ¶106.

IM example: Bodart discloses that “virtual assistant may **put [a call] on hold** and attempt to reach user 201 via... **instant message**.” Ex.1005, [0060]. Thus, Bodart teaches using IM service (“*the second service*”) to reach the user when the virtual assistant intercepts a call to a telephone number associated with the user or virtual assistant (“*the first telephone number*”). Ex.1005, [0060]; Ex.1003, ¶145. Bodart teaches that these functionalities may be facilitated by a stored “set of rules...setting forth the logic used by virtual assistant to aid the user.” Ex.1005, [0021], [0057], [0060]. It would have been obvious to a POSITA for the virtual assistant to use stored rules associating the IM service with “*the first telephone*

number” to automate the IM functionality described above. Ex.1005, [0060]; Ex.1003, ¶¶142-145.

Bodart further explains that “a user may be able to send messages using an instant message client.” Ex.1005, [0034]. Dharmarajan explains that an “instant messaging client” performs a login “to gain access to the server computer that provides the instant messaging service.” Ex.1007, 1:35-38. It would have been obvious to a POSITA that Bodart’s account information would include the name or address of the IM server computer (“*routing and interworking data necessary to utilize the second service*”) for gaining access to the IM service. Ex.1005, [0034], [0060]; Ex.1003, ¶¶146-147. The name or address of the IM server computer would be stored in the account information with the telephone number of the virtual assistant or user (“*the first telephone number*”), and therefore would be “*associated with*” the telephone number. Ex.1005, [0022]; Ex.1003, ¶147.

Email example: Bodart contemplates “having a user’s email read to her by phone.” Ex.1005, [0002]. In another example, Bodart discloses that “virtual assistant 101 may attempt **to send** a text or **email message** to user 201 to determine the user’s availability to take a particular **call**.” Ex.1005, [0064]. Thus, Bodart teaches providing email access by phone using a telephone number associated with the user or virtual assistant (“*the first telephone number*”). Ex.1003, ¶148. If the user calls into the virtual assistant to access email by phone,

the number the user calls is “*the first telephone number.*” Ex.1003, ¶¶102-103, 149. Alternatively, if the virtual assistant calls the user at his/her phone number, the user’s phone number is “*the first telephone number.*” Ex.1003, ¶¶102-103, 149. In either case, Bodart’s virtual assistant system uses the phone number to provide access to email services, which renders obvious to “*associate the second service with the first telephone number.*” Ex.1003, ¶149.

It would have also been obvious to a POSITA that Bodart’s account information would include the name or address of the email server computer (“*routing and interworking data necessary to utilize the second service*”) for accessing the email service. Ex.1005, [0022], [0044], [0072]; Ex.1003, ¶¶150-151. Bodart explains using email services. Ex.1005, [0027], [0040], [0044]. A POSITA would have understood that the email server’s address would be used to access the email service. Ex.1012, pp.104-105; Ex.1022, 5:27-44; Ex.1003, ¶150. It would have been obvious to store the server’s address in the user data module with the telephone number of the user or the virtual assistant. Ex.1003, ¶150. Therefore, it would have been obvious that the server’s address would be “*associated with*” the telephone number of the virtual assistant or user (“*the first telephone number*”). Ex.1005, [0022]; Ex.1003, ¶¶150-151.

6. Claim 2

[2.0] The telephone system as recited in claim 1, wherein the server is further

configured to associate a plurality of services with the first telephone number,

As explained above at Section XI.D.5.[1.0], the functionalities of the virtual assistant 101 are accessible to the user by telephone. Ex.1005, [0030]; Ex.1003, ¶¶152-153. As further explained above at Section XI.D.5.[1.8], the prior art renders obvious associating a *plurality of services*, such as IM and email services, with a user's telephone number ("*first telephone number*"). Ex.1003, ¶¶142-151.

Bodart's virtual assistant is capable of accessing a wide variety of "*services*," such as "search[ing] for the text string among all documents" (Ex.1005, [0066]), sending a document "by facsimile" (Ex.1005, [0068]), "controlling functions within a house" (Ex.1005, [0074]), and "checking of voicemail, the monitoring of a calendar, the viewing of email, and access to additional user data." Ex.1005, [0072]. To perform such tasks, Bodart discloses that the "virtual assistant 101 may receive information from **various sources 130** using one or more computer networks." Ex.1005, [0028]. The virtual assistant 101 also accesses information from third-party applications such as "**Google Maps**" and "**iCal**." Ex.1005, [0040], [0046], [0047]; Ex.1003, ¶¶152-156.

The virtual assistant's functions are accessible via a telephone interface. Ex.1005, [0030]. This includes services like Google Maps, iCal, IM, etc. Because these other services can be accessed by telephoning the virtual assistant, it would have been obvious that the virtual assistant "*associate[s]*" these services "*with the*

first telephone number.” See Ex.1005, [0030], [0033] (“a user may control virtual assistant 101 using an audio interface 203 using a conventional telephone”); Section XI.D.5.[1.6]; Ex.1003, ¶¶153-154, 156. For instance, it would have been obvious that Bodart’s virtual assistant would be configured to provide access to email, IM, or mapping information to the user based on the user’s telephone number, the virtual assistant’s telephone number, or both. Ex.1003, ¶¶155, 157; Section XI.D.4.a-b. Thus, Bodart renders obvious that the server providing the virtual assistant is “*configured to associate a plurality of services with the first telephone number.*” Ex.1003, ¶¶152, 157.

[2.1] wherein the plurality of services are provided by a plurality of providers different than the first provider, and wherein the plurality of services are utilized by the first user.

First, Bodart explains that “**information sources 130** ... may reside on remote networks, interconnected for example, via the Internet.” Ex.1005, [0028]. The information sources include date and time, weather, geographic location, and road traffic information, among others. Ex.1005, [0029]. The virtual assistant 101 may use third party applications to obtain information such as “Google Maps” and “iCal.” Ex.1005, [0040], [0046]-[0047]. Thus, Bodart teaches the information sources 130 are “*provided by a plurality of providers.*” Ex.1003, ¶¶158-159. Further, it would have been obvious that the providers of the information sources (e.g., Google Maps, iCal) would be “*different than*” the telephone service provider

(“*first provider*”). Ex.1003, ¶160.

Second, Bodart teaches that “a user may **query** virtual assistant with regard to various aspects of user data 120 and/or **information sources 130**.” Ex.1005, [0027]. Thus, Bodart teaches that the information sources “*are utilized by the first user*” via the virtual assistant. Ex.1003, ¶¶161-162.

7. Claim 6

[6.0] The telephone system as recited in claim 1, wherein the second service is a conferencing application, voice attendant application, or text messaging application.

As explained above at Section XI.D.5.[1.4], both the IM service and email service teach “*the second service*.” Each of IM service and email service is a “*text messaging application*.” Ex.1003, ¶¶106, 164.

In another example, Bodart teaches that “[i]nteractive voice response (IVR) technologies may enable speech processing 113 abilities for virtual assistant 101.” Ex.1005, [0026]. Bodart describes the virtual assistant using “speech processing” to “search user files 123, emails 127, calendar 121 appointments, and so forth.” Ex.1005, [0027]. IVR or speech processing functionality for issuing commands is a “*voice attendant application*.” Ex.1003, ¶¶163, 165-166.

8. Claim 7

All limitations of claim 7, except the preamble [7.0] which is addressed below, are substantively identical to limitations of claims already addressed above,

as correlated in the table below:

Claim 7 Limitation	Corresponding Claim Limitation
[7.1]	[1.2] and [1.3]
[7.2]	[1.4] and [1.5]
[7.3]	[1.6]
[7.4]	[1.7]
[7.5]	[1.8]

The combination of Bodart, Banister, and Dharmarajan renders obvious these limitations for the same reasons discussed above. Ex.1003, ¶¶167-173.

[7.0] A method comprising:

Bodart teaches “[s]ystems and **methods** for assisting a user with a variety of tasks.” Ex.1005, Abstract; *see also*, Ex.1005, Figs. 4, 6, 8, [0009], [0013], [0061]. Thus, Bodart teaches the preamble. Ex.1003, ¶168. The combination of Bodart, Banister, and Dharmarajan discloses a method “*comprising*” the recited elements. Ex.1003, ¶¶167-168.

9. Claims 8 and 12

Claims 8 and 12 are substantively identical to claims 2 and 6, respectively, already addressed above. The combination of Bodart, Banister, and Dharmarajan renders obvious these claims for the same reasons discussed above. Ex.1003,

¶¶174-176.

10. Claim 13

All limitations of claim 13, except the preamble [13.0] which is addressed below, are substantively identical to limitations of claims already addressed above, as correlated in the table below:

Claim 13 Limitation	Corresponding Claim Limitation
[13.1]	[1.2] and [1.3]
[13.2]	[1.4] and [1.5]
[13.3]	[1.6]
[13.4]	[1.7]
[13.5]	[1.8]

The combination of Bodart, Banister, and Dharmarajan renders obvious these limitations for the same reasons discussed above. Ex.1003, ¶¶177-185.

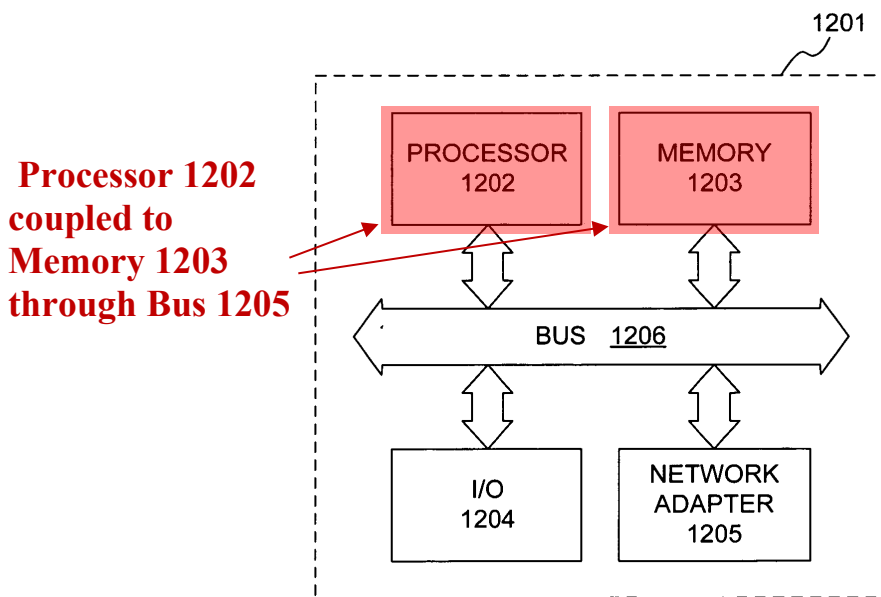
[13.0] A non-transitory computer readable storage medium comprising program instructions, wherein when executed the program instructions are operable to:

Bodart describes a “virtual assistant 101,” which may be implemented as “software.” Ex.1005, [0020]; Ex.1003, ¶177. A POSITA would have been familiar with computer software, including the fact that computer software is typically implemented as executable program instructions stored on a non-transitory computer readable medium. Ex.1003, ¶178. Thus, to the extent the preamble is

limiting, Bodart’s discussion of virtual assistant software renders it obvious.

Ex.1003, ¶178.

Fig.12 of Bodart (below) further illustrates an “operating environment in which one or more aspects of the invention may be implemented.” Ex.1005, [0075]. The device 1201 of Fig.12 includes a memory 1203 in communication with a processor 1202. Ex.1005, Fig.12; Ex.1003, ¶179.



Ex.1005, Fig.12 (annotated); Ex.1003, ¶179.

Bodart further teaches that the virtual assistant 101 can include a rules engine that “may be compared with a software compiler which translates user input (i.e. computer source code) into **a series of instructions which a computer processor can understand.**” Ex.1005, [0024]. Bodart also explains that the virtual assistant may connect to the information sources and communication interfaces remotely via

a network. Ex.1005, [0028], [0030], [0074]; Ex.1003, ¶¶180-181.

Bodart's further teachings of processor 1202, memory 1203, compiling computer-understandable instructions, and remote connection to information sources and communication interfaces further render obvious the preamble.

Ex.1005, [0075]; Ex.1003, ¶182.

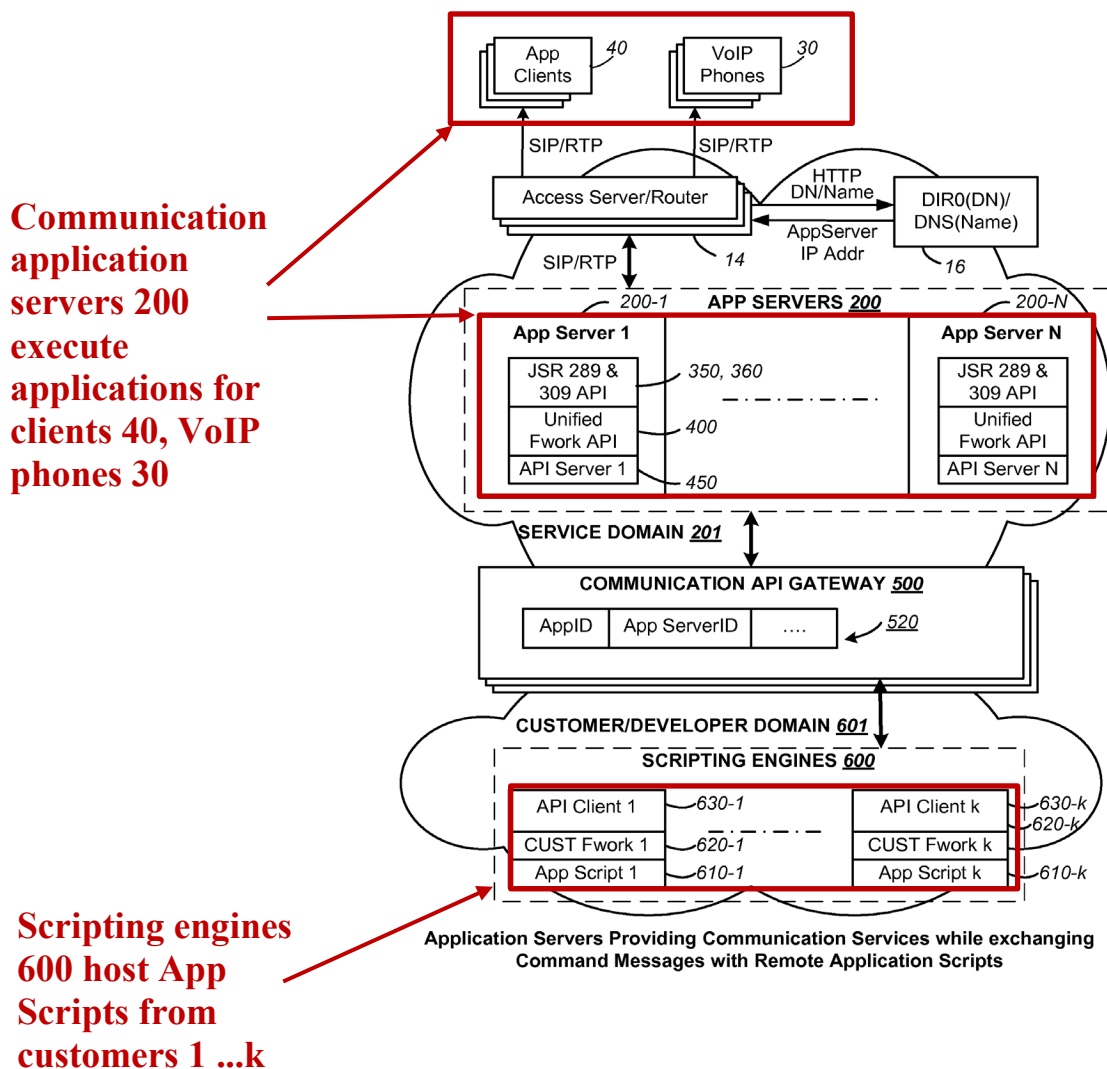
11. Claim 14

Claim 14 is substantively identical to claim 2, already addressed above. The combination of Bodart, Banister, and Dharmarajan renders obvious this claim for the same reasons discussed above. Ex.1003, ¶¶188-189.

E. Ground 2: Claims 1, 2, 6-8, and 12-14 are obvious under 35 U.S.C. § 103 over de Castro

1. De Castro

De Castro describes “a plurality of communication application servers 200” that “allow[] application clients 40 ... to access the applications hosted in the communication application servers.” Ex.1008, [0115]. Further, “a plurality of scripting engines 600” are used to “host and process the customers’ application scripts 610.” Ex.1008, [0119]. VoIP phones 30, application clients 40 (i.e., users), application servers 200, and scripting engines 600 are illustrated in Fig.14 below. Ex.1003, ¶¶190-192.

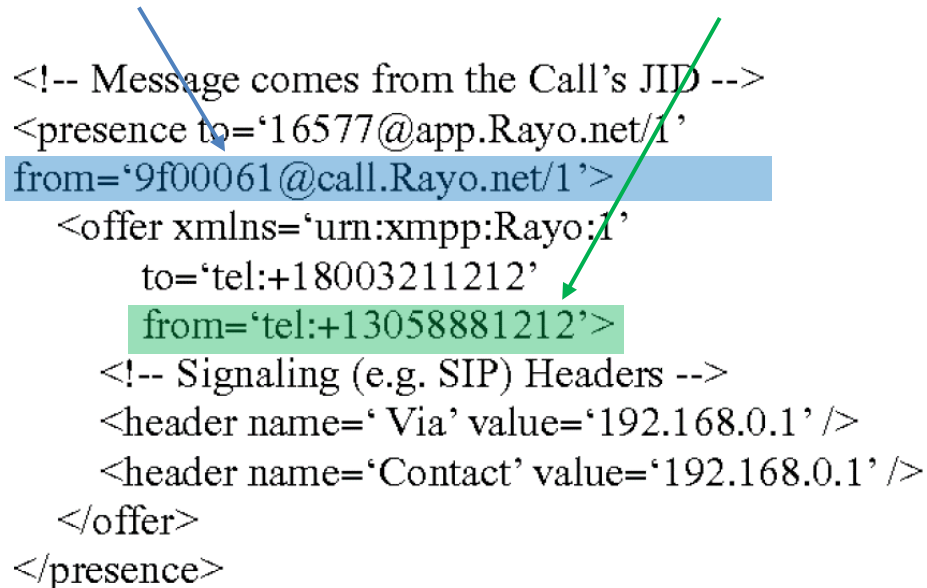


Ex.1008, Fig.14 (annotated); Ex.1003, ¶192

An example messaging API protocol used by the communication application servers (App servers 200) is the “Rayo™ protocol.” Ex.1008, [0138]-[0139]. The Rayo protocol “primarily deals with calls” which “are offered to Rayo clients via XMPP [extensible messaging and presence protocol] using a Jabber Identifier (JID).” Ex.1008, [0139]; Ex.1003, ¶193.

The “JID allow[s] a two way conversation between the Rayo client and the server that’s handling the call signaling and media.” Ex.1008, [0139]. In the excerpt below, de Castro illustrates the JID associated with a user’s telephone number so that a Rayo Client can control a call. Ex.1008, [0141]; Ex.1003, ¶194.

**The Rayo protocol includes associating a JID
(top) with a user’s telephone number (bottom)**



```
<!-- Message comes from the Call's JID -->
<presence to='16577@app.Rayo.net/1'
from='9f00061@call.Rayo.net/1'>
  <offer xmlns='urn:xmpp:Rayo:1'
    to='tel:+18003211212'
    from='tel:+13058881212'>
    <!-- Signaling (e.g. SIP) Headers -->
    <header name='Via' value='192.168.0.1' />
    <header name='Contact' value='192.168.0.1' />
  </offer>
</presence>
```

Ex.1008, [0141] (annotated); Ex.1003, ¶194

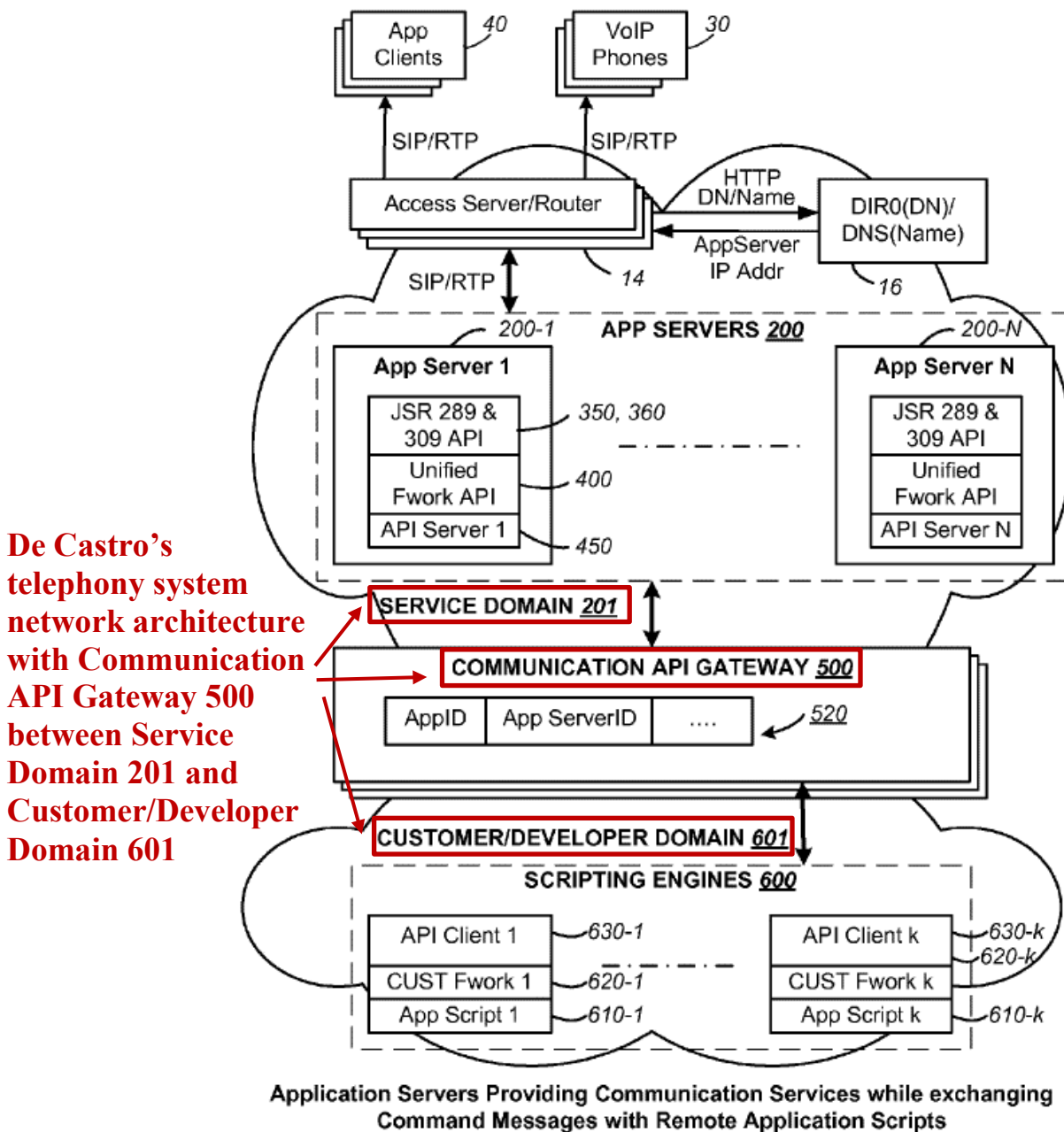
De Castro is analogous art to the '254 patent and is from the same field of endeavor. Ex.1003, ¶195. De Castro describes a “unified framework” that is used to provide application clients access to a variety of communication applications. Ex.1008, Abstract. Similarly, the '254 patent provides a “unified services enabler platform” to access “a variety of services and applications from multiple different ecosystems.” Ex.1001, Abstract; Ex.1003, ¶195. De Castro is also reasonably

pertinent to the particular problem of the '254 patent: it allows service providers using electronic IDs (e.g., Jabber ID) to provide services to clients by telephone. Ex.1001, Abstract, 11:33-42; Ex.1008, [0138]-[0141]; Ex.1003, ¶195.

2. Claim 1

[1.0] *A telephone system, comprising:*

De Castro “relates to telecommunication and a networked computer **telephony system.**” Ex.1008, [0002]. As illustrated in Fig.14 below, de Castro’s telephony system “network architecture” includes a “communication API gateway 500” between a “service domain 201” and a “customer/developer domain 601.” Ex.1008, [0114], [0120]; Ex.1003, ¶¶196-198.



Ex.1008, Fig.14 (annotated)

The service domain 201 has “a plurality of communication applications servers 200” that “provide telephony and communication services to applications.” Ex.1008, [0115]. Further, the “API Gateway 500 is deployed on the network to

service as a messaging broker between the plurality of customers with application scripts in the customer/developer domain 601 and the group of communication application servers 200 in the service domain 201.” Ex.1008, [0120]; Ex.1003, ¶¶197-198.

Thus, de Castro’s telephony system including service domain 201, API gateway 500, and customer/developer domain 601 teaches a “*telephone system*.” Ex.1003, ¶199. As demonstrated below, de Castro discloses a system “*comprising*” the recited elements.

[1.1] *a database; and*

De Castro teaches that the API gateway 500 “includes a **registry 520**” (“*database*”). Ex.1008, [0129]; Ex.1003, ¶¶201-204. A POSITA would have found it obvious for a registry to be, or to include, a database because the registry stores a variety of data and associations among the stored data. Ex.1013, 4:34-37; Ex.1003, ¶¶201-204.

De Castro explains that registry 520 “registers the communication applications servers 200 in the service domain and the customers’ applications in the customer/developer domain 601.” Ex.1008, [0130]. Registry 520 also “log[s] the properties and status associated with each communication application server 200 and the properties and status associated with each API client[] 630.” Ex.1008, [0130]. Fig.19 of de Castro below shows the registry 520 with logged properties

and statuses. Ex.1003, ¶¶202-204. It would have been obvious to a POSITA that the registry 520 is a *database*; alternatively, it would have been obvious to implement registry 520 as a *database* because a database was a commonly used tool for storing information like that stored by the registry. Ex.1013, Abstract, 6:55-7:3; Ex.1014 11:8-12; Ex.1003, ¶¶205-207.

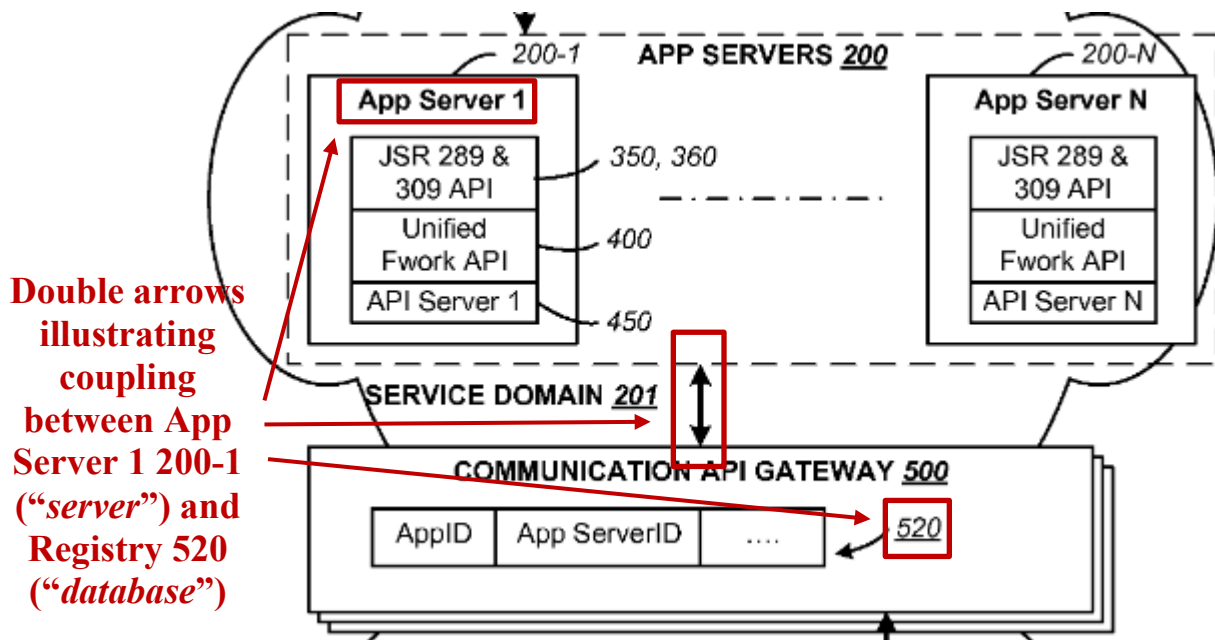
Registry <u>520</u>					
SERVICE DOMAIN			USER DOMAIN		
App ServerID	App Server Properties/ Status		App ID	App Properties/ Status	

Ex.1008, Fig.19

Thus, de Castro's registry 520 renders obvious "*a database*." Ex.1003, ¶¶200, 205-207.

[1.2] *a server coupled to the database,*

As explained at Section XI.E.5.[1.1], the registry 520 renders obvious "*a database*." De Castro teaches that in "the service domain 201 is a plurality of **communication application servers 200, such as App Server 1** to App Server N." Ex.1008, [0115]; Ex.1003, ¶¶209-210. De Castro's Fig.14 shown below illustrates App Server 1 200-1 ("*server*") in communication with (double arrows) Registry 520 ("*a server coupled to the database*"). Ex.1008, Fig.14; Ex.1003, ¶¶208-214.



Ex.1008, Fig.14 (partial annotated)

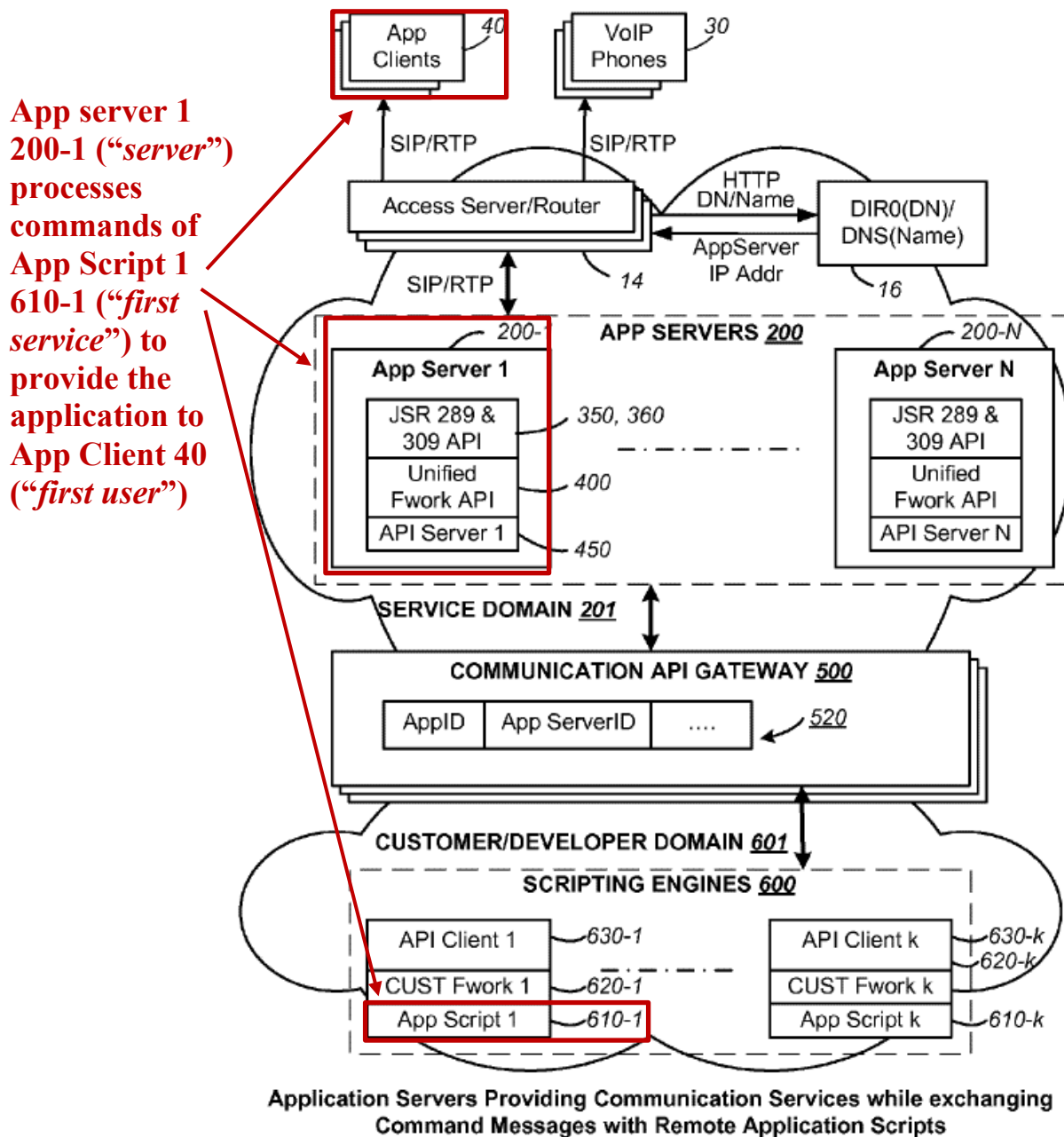
[1.3] wherein the server is utilized by a first provider to provide a first service to a first user, wherein the first service is telephone service, and wherein the telephone service is associated with a first telephone number;

First, de Castro teaches that “customer 1” (“a first provider”) has “the scripting engines 600 host and process an application script such as **App Script 1 610-1**” (“a first service”). Ex.1008, [0119]; Ex.1003, ¶215. Further, de Castro teaches that App Server 1 (“server”) provides “application clients 40” (“a first user”) with access to the application of app script 1 (“the server ... provide[s] a first service to a first user”). Ex.1003, ¶¶215-217. For instance, “the **communication application servers 200** provide telephony and communication services to applications and allows **application clients 40** such as, for example, VoIP phones to **access the applications** hosted in the communication application

servers,” including the application of App Script 1. Ex.1008, [0115], [0119]; Ex.1003, ¶217.

De Castro explains that to provide application clients 40 with an application service, “API client 1 630-1 [] interprets App Script 1 and renders it in terms of the commands of the finite set defined at the API server 450 of the communication application server 200.” Ex.1008, [0119]. Then, “API client 630 then packages the commands as messages in accordance with the messaging API protocol to be sent to a connected communication application server 200 such as App server 1 200-1.” Ex.1008, [0119]; Ex.1003, ¶216.

Fig.14 below illustrates app server 1 providing application client 40 with the application of app script 1.



Ex.1008, Fig.14 (annotated); Ex.1003, ¶219

Thus, de Castro teaches that App Server 1 is used by customer 1 to provide the application of App Script 1 to an application client 40, which renders obvious “wherein the server is utilized by a first provider to provide a first service to a first user.” Ex.1003, ¶¶217-220.

Second, de Castro teaches “providing a **telephony-enabled service** via a message-based API interface” (“*the first service is telephone service*”). Ex.1008, [0002]; Ex.1003, ¶¶221-222. For instance, App Script 1 can be a telephony-enabled service, as “application clients such as **VoIP phones 30 [are] able to access a specific communication application by calling a dialed number.**” Ex.1008, [0116], [0119]; Ex.1003, ¶¶222-223.

Third, it would have been obvious to a POSITA that “*the telephone service is associated with a first telephone number,*” such as the dialed number or the dialing-user’s number. Ex.1008, [0116], [0141]; Ex.1003, ¶¶224-227.

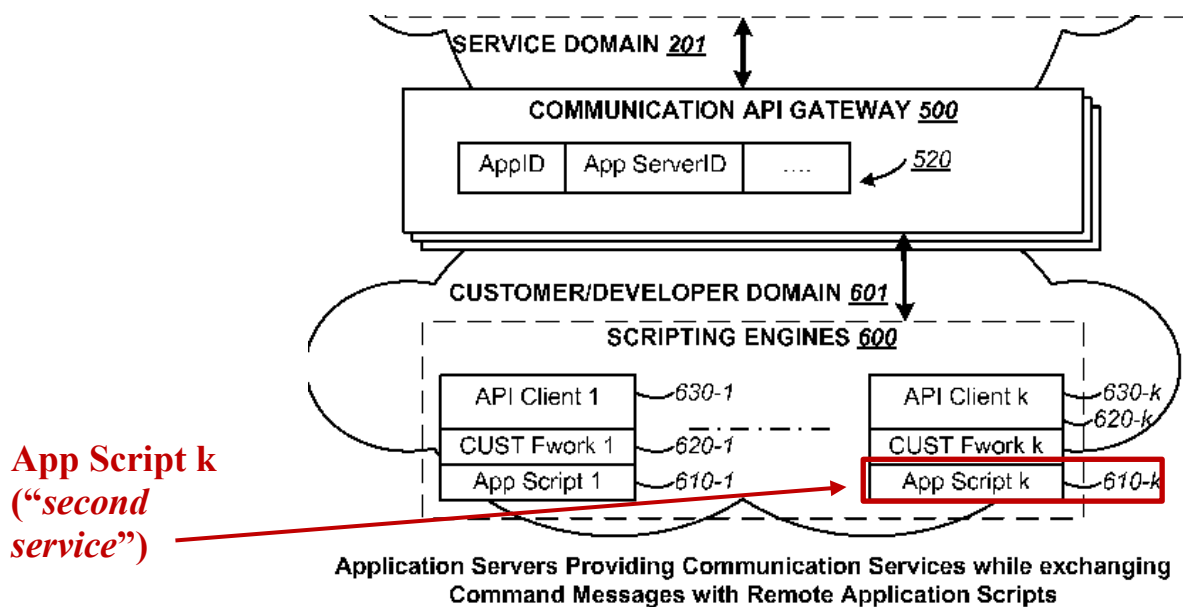
A POSITA would have understood that where “**application clients ... access a specific communication application** by calling a dialed number,” each client has a telephone number (“*first telephone number*”) that is associated with the communication application. Ex.1008, [0116]; Ex.1003, ¶¶225-226. For instance, de Castro describes a telephony application that receives an incoming call from a specific telephone number: “Incoming Calls <!-- Message comes from the Call's JID --> ... to=**tel:+18003211212** **from=**tel:+13058881212****>.” Ex.1008, [0141].

Thus, de Castro teaches user VoIP phones accessing a telephony-enabled application, such as a call service associated with the dialing-user’s telephone number. This disclosure renders obvious “*wherein the first service is telephone service, and wherein the telephone service is associated with a first telephone*

number.” Ex.1003, ¶¶226-228.

[1.4] wherein the server is configured to: receive a request to provision a second service for the first user,

First, de Castro’s telephone system provides users access to a plurality of “application scripts 610” including “App Script 1” (“*first service*”) and “App Script k” (“*second service*”). Ex.1008, Fig.14, [0119]. De Castro illustrates App Script 1 and App Script k in Fig.14 below. Ex.1008, Fig.14; Ex.1003, ¶¶229-230.



Ex.1008, Fig.14 (partial annotated), Ex.1003, ¶230

Second, de Castro teaches one of the application servers 200 (“*server*”) receiving, from an application client (“*first user*”), a call to access to one of the communication applications (“*request to provision a second service*”): “application clients such as VoIP phones 30, [are] able to **access a specific communication**

application by calling a dialed number or an URL.” Ex.1008, [0116]; Ex.1003, ¶231. In response to the call from a client 40, the access server router 14 “return[s] the address of the communication application server executing the specification communication,” and the App server 200 registers the app server address for the call to provision an App Script for the client 40 dialing in. Ex.1008, [0116], [0124], [0126], Fig.16; Ex.1003, ¶¶231-232. A POSITA would have understood that the App Server receiving the user’s call to provide access to a communication application teaches the app server “*receiv[ing] a request to provision a second service for the first user.*” Ex.1008, [0116]; Ex.1003, ¶¶231-235. Thus, de Castro teaches that the App servers 200 receive a routed call to provide a client 40 access to an App Script, which renders obvious this limitation. Ex.1003, ¶235.

[1.5] *wherein the second service is provided by a second provider, and wherein the second provider is different than the first provider;*

De Castro teaches that “**customer 1**” (“*a first provider*”) uses “the scripting engines 600 [to] host and process an application script such as App Script 1 610-1” (“*first service*”). Section XI.E.2.[1.4]; Ex.1008, [0119]. De Castro’s telephone system includes a plurality of “application scripts 610” including “App Script 1” and “**App Script k**” (“*second service*”). Section XI.E.2.[1.4]; Ex.1008, Fig.14, [0119]; Ex.1003, ¶236.

De Castro explains that its telephone system architecture provides “a

messaging broker between the **plurality of customers** with application scripts and the group of communication application servers.” Ex.1008, [0039]. De Castro further explains that, in its system, a “**customer using the communication services can code the application script in a preferred programming language** using a custom framework providing a set of **customer-specific libraries.**”

Ex.1008, [0041]; Ex.1003, ¶237. It would have been obvious to a POSITA that this plurality of customers, with each customer (“*provider*”) using a preferred programming languages and customer-specific library to code the various application scripts, refers to different customers (“*the second provider is different than the first provider*”). Ex.1008, [0041]; Ex.1003, ¶¶237-239. Thus, de Castro teaches a plurality of different customers providing users access to a plurality of different application scripts, which renders obvious this limitation. Ex.1003, ¶239.

[1.6] responsive to receiving the request: create a mapping from the first telephone number to a second identifier (ID), wherein the second ID is utilized by the second provider to identify the first user;

First, de Castro describes commands of a communication application responding to “[i]ncoming calls” (“*responsive to receiving the request*”). Ex.1008, [0141]. For instance, “inbound calls...are offered to Rayo clients via XMPP using a Jabber Identifier (JID).” Ex.1008, [0139]. De Castro describes offering an inbound call from a telephone number (“*the first telephone number*”) to a Rayo client using a “from” JID (“*second ID*”): “Incoming Calls <!-- Message comes

from the Call's JID --> ... **from='9f00061@call.Rayo.net/1'>** ...

from='tel:+13058881212'>." Ex.1008, [0141]; Ex.1003, ¶¶240-242. Offering the inbound call to a Rayo client from the telephone number by associating the telephone number with a JID teaches to "*create a mapping from the first telephone number to a second identifier.*" Ex.1003, ¶243.

De Castro also teaches that the commands described above are handled through the API server 450 of a App server 200: "The following are examples of the commands supported by the unified framework API 400 through the **API server 450**" Ex.1008, [0138]; Ex.1003, ¶244. Thus, de Castro teaches that the App server 200 ("*server*") is configured to "*create a mapping from the first telephone number to a second identifier (ID),*" "*responsive to receiving the request.*" Ex.1003, ¶¶245-246.

Second, it would have been obvious that the JID "*is utilized by the second provider to identify the first user.*" De Castro teaches, in the context of the Rayo protocol, that "[e]ach call is ... represented by its own unique JID." Ex.1008, [0139]; Ex.1003, ¶247. A POSITA would have understood that the JID representing the call is "*utilized ... to identify*" the end user (e.g., client 40). For example, XEP-0327 ("the Rayo Specification") explains that the actors (e.g., client, server, call) in a Rayo deployment scheme "are represented by XMPP entities with a JID of their own." Ex.1009, [5.2]; Ex.1003, ¶¶247-248. For

instance, for a Rayo client, the Rayo Specification discloses that “any JID” may represent the Rayo client. Ex.1009, [5.2]; Ex.1003, ¶247. As illustrated below, de Castro shows that an app server offers a call from a telephone number (“tel:+13058881212”) using a JID (“9f00061@call.Rayo.net/1”).

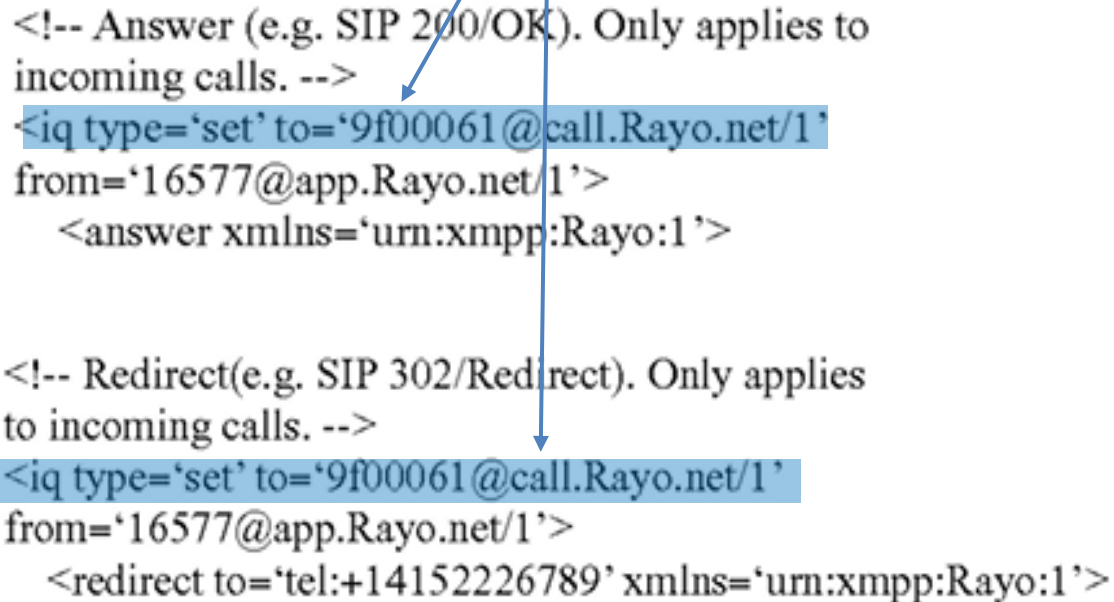
Rayo server offers a call from a user's telephone number (bottom) using a JID (top).

```
<!-- Message comes from the Call's JID -->  
<presence to='16577@app.Rayo.net/1'  
from='9f00061@call.Rayo.net/1'>  
  <offer xmlns='urn:xmpp:Rayo:1'  
    to='tel:+18003211212'  
    from='tel:+13058881212'>
```

Ex.1008, [0141] (annotated); Ex.1003, ¶248

Once the app server has offered the call to the Rayo client, the Rayo client controls the call with commands (e.g., Answer, Redirect) using the call’s JID (“9f00061@call.Rayo.net/1”):

**Rayo client issues commands for the call using
JID only.**



```
<!-- Answer (e.g. SIP 200/OK). Only applies to  
incoming calls. -->  
<iq type='set' to='9f00061@call.Rayo.net/1'  
from='16577@app.Rayo.net/1'>  
  <answer xmlns='urn:xmpp:Rayo:1'>  
  
<!-- Redirect(e.g. SIP 302/Redirect). Only applies  
to incoming calls. -->  
<iq type='set' to='9f00061@call.Rayo.net/1'  
from='16577@app.Rayo.net/1'>  
  <redirect to='tel:+14152226789' xmlns='urn:xmpp:Rayo:1'>
```

Ex.1008, [0141] (annotated); Ex.1003, ¶249

A POSITA would have thus understood that the JID of de Castro “*identif[ies] the first user.*” Ex.1003, ¶250. For instance, a POSITA would have understood that the JID addresses commands from the Rayo client (e.g., Answer) to the user in the call session. Ex.1003, ¶250. Further, it would have been obvious that the scripting engine 600-k would utilize a JID representing a call for controlling a call for the “*first user.*” Ex.1003, ¶251. Thus, it would have been obvious for the customer (“*second provider*”) associated with scripting engine 600-k to utilize the JID “*to identify*” the client 40 (“*first user*”). Ex.1003, ¶251.

[1.7] [responsive to receiving the request:] activate the second service with the second provider utilizing the second ID as a user ID for the first user; and

First, de Castro explains that the App server executes commands generated from the App Scripts 610: “[w]hen a command carried in a message is received at the communication application server 200, the API server 450 will...allow the command to be queued with the associated call (actor) to be executed.” Ex.1008, [0122]. The App server registers Call IDs of incoming calls “into a call ID registry 262 and start a call actor 1 270-1” Ex.1008, [0124]; Ex.1003, ¶252. “As command messages from the API clients 630 (see FIG. 14) are received by the application server 200, they are identified with a call actor and queued in an associated message queue 272, which are then applied to the call in order.” Ex.1008, [0124]; Ex.1003, ¶¶252-254. A POSITA would have understood that registering a call ID of an incoming call and starting an associated call actor for an App Script such as App Script 610-k (“*the second service*”) would include “*activat[ing]*” the App Script 610-k. Ex.1003, ¶¶255-256.

Second, it would have been obvious to a POSITA that the app server would execute App Script 610-k “*utilizing the second ID as a user ID for the first user.*” Ex.1003, ¶257. For instance, de Castro teaches that a Rayo client (“*second provider*”) uses the JID to direct commands from the communication application to the user in the call. Ex.1008, [0141]; Section XI.E.2.[1.6]; Ex.1003, ¶257. It would have been obvious to a POSITA that the customer associated with scripting engine 600-k would similarly use the JID to direct commands from the App Script 610-k

to the client 40 (“*the first user*”) during a call session. Ex.1008, Fig.14, [0119]; Ex.1003, ¶257.

[1.8] [responsive to receiving the request:] associate the second service with the first telephone number, wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.

De Castro teaches a server executing a communication application (“*second service*”) using a JID: “Each call is in turn represented by its own unique JID.” Ex.1008, [0139]; Ex.1003, ¶258. As discussed at Section XI.E.2.[1.6], de Castro teaches that the JID is mapped to the telephone number (“*the first telephone number*”) of the client accessing the communication application. Ex.1003, ¶259. Thus, the App server associates the telephone number with the JID, which is associated with the communication application. Ex.1003, ¶259. Therefore, de Castro teaches that the App server “*associate[s]*” the communication application “*with the first telephone number.*” Ex.1003, ¶¶259-260. The second service, accessible by a phone call, is associated with the phone number because the server maintains the call with the client. Ex.1008, [0139]; Ex.1003, ¶¶259-260.

Further, it would have been obvious to a POSITA that “*the first telephone number is associated with routing and interworking data necessary to utilize the second service.*” Ex.1003, ¶261. De Castro teaches that “the call manager register [sic] the Call ID into a call ID registry 262 and start[s] a call actor 1 270-1, and similarly for call 2, . . . , and call i.” Ex.1008, [0124]. De Castro also teaches that

“[i]nbound calls ... are offered to Rayo clients via XMPP using a [JID].” Ex.1008, [0139]; Ex.1003, ¶261. A POSITA would understand that a phone call from the public switched telephone network (PSTN) is associated with the client’s phone number, while the command messages received by the App server from the scripting engines use the client’s JID. Ex.1003, ¶261. It would have been obvious that the App servers would store an association between the client 40’s phone number and the JID in the call ID registry 262. Ex.1003, ¶261. In this way, the app servers can identify the correct phone call corresponding to the JID indicated in the command messages from the developers. Ex.1003, ¶261.

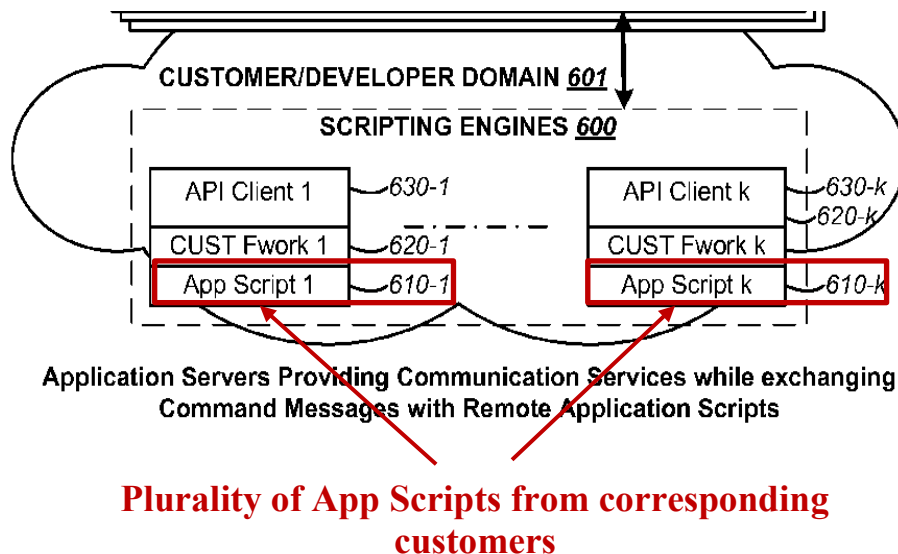
Thus, de Castro renders obvious associating a telephone number and JID in a call ID registry to obtain routing information between the JID and telephone number. Ex.1003, ¶262. The stored association would be “*routing and interworking data necessary to utilize the second service.*” Ex.1003, ¶262. Taken together, de Castro renders obvious this limitation Ex.1003, ¶262.

3. Claim 2

[2.0] The telephone system as recited in claim 1, wherein the server is further configured to associate a plurality of services with the first telephone number,

First, as explained above at Section XI.E.2.[1.8], de Castro renders obvious that the App server 200 associates an App Script 610 with the client 40’s telephone number (“*the first telephone number*”). Ex.1008, [0141]; Ex.1003, ¶263.

Second, de Castro teaches that “a plurality of scripting engines 600 ... host and process the customers' **application scripts 610** [*“plurality of services”*].” Ex.1008, [0119]; Ex.1003, ¶264.



Ex.1008, Fig.14 (partial annotated), Ex.1003, ¶264

Third, it would have been obvious to a POSITA that the App server 200 would be “*configured to associate*” the App Scripts 610 with the client’s 40 telephone number. Ex.1003, ¶265. For example, for the same reason it would have been obvious to associate the App Script 610-k (“*the second service*”) with the client 40’s telephone number, it would have been obvious that the app server 200 associates two or more App Scripts (e.g., 610-1 ... 610-k) with the client 40’s telephone number. Ex.1003, ¶265. For example, the same client 40 may, at different times, call different phone numbers that each invoke a different App

Script. Ex.1003, ¶265. It would have been obvious that the App server 200 would similarly associate each of these invoked App Scripts with the client 40's telephone number when they are executed. Ex.1003, ¶265. Thus, it would have been obvious to a POSITA that the App server 200 would “*associate a plurality of services with the first telephone number.*” Ex.1003, ¶265.

[2.1] wherein the plurality of services are provided by a plurality of providers different than the first provider, and wherein the plurality of services are utilized by the first user.

As explained above at Section XI.E.6.[2.0], de Castro teaches that “a plurality of customers” have their respective App Scripts (“*plurality of services*”) hosted on the scripting engines 600. Ex.1008, [0110], [0129]; Ex.1003, ¶266. A POSITA would have understood that the App Scripts (“*plurality of services*”) are “*provided by*” the plurality of customers (“*plurality of providers*”). Ex.1003, ¶266. Because there are multiple customers, it would have been obvious to a POSITA that at least one of the customers would be different from first customer (“*the first provider*”) associated with the App Script 610-1. Ex.1003, ¶267.

Further, de Castro teaches:

the communication application servers 200 ... allows [sic]
application **clients** 40 ... to access the **applications** hosted
in the communication application servers.

Ex.1008, [0115]; Ex.1003, ¶268.

As explained at Section XI.E.2.[2.1] above, it would have been obvious that the same client 40 (“*the first user*”) would call different telephone numbers corresponding to different App Scripts. Ex.1003, ¶269. Thus, it would have been obvious that a plurality of the communication applications embodied in App Scripts 610 would be “*utilized by the first user.*” Ex.1003, ¶¶268-269.

4. Claim 6

[6.0] The telephone system as recited in claim 1, wherein the second service is a conferencing application, voice attendant application, or text messaging application.

De Castro teaches that the applications supported by the unified communication framework includes interactive voice response (IVR) and “Conferencing” applications. Ex.1008, [0075]; Ex.1003, ¶270. De Castro further describes the IVR concept and other “voice-enabled self-help applications.” *See* Ex.1008, [0104], [0008]-[0010]; Ex.1003, ¶¶270-271. Thus, it would have been obvious that the App Script 610-k includes “*a conferencing application*” or a “*voice attendant application.*” Ex.1003, ¶¶270-271.

5. Claim 7

All limitations of claim 7, except the preamble [7.0] which is addressed below, are substantively identical to limitations of claims already addressed above, as correlated in the table below:

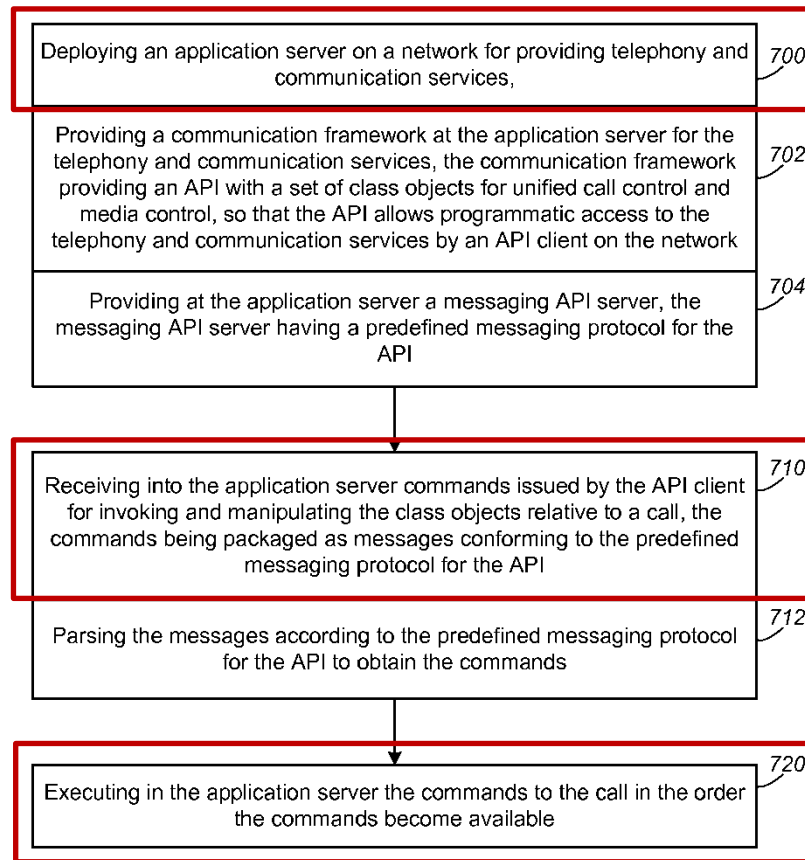
Claim 7 Limitation	Corresponding Claim Limitation
[7.1]	[1.2] and [1.3]
[7.2]	[1.4] and [1.5]
[7.3]	[1.6]
[7.4]	[1.7]
[7.5]	[1.8]

De Castro renders obvious these limitations for the same reasons discussed above. Ex.1003, ¶¶274-278.

[7.0] A method comprising:

De Castro discloses that a server performs the method of Fig.20 to provide services to clients or users: “Fig.20 is a flow diagram of a **method** of operating the communication application server.” Ex.1008, [0131]. The method shown in Fig.20, reproduced below, includes the aspects of the network architecture of Fig.14 described above in [1.2]-[1.8]. Ex.1008, Fig.20; Ex.1003, ¶272.

**Method
includes steps
for executing
customer
applications
using an
application
server**



Ex.1005, Fig.20 (annotated), Ex.1003, ¶272.

Thus, de Castro teaches the preamble. Ex.1003, ¶¶272-273.

6. Claims 8 and 12

Claims 8 and 12 are substantively identical to claims 2 and 6, respectively, already addressed above. De Castro renders obvious these claims for the same reasons discussed above. Ex.1003, ¶¶279-281.

7. Claim 13

All limitations of claim 13, except the preamble [13.0] which is addressed below, are substantively identical to limitations of claims already addressed above,

as correlated in the table below:

Claim 13 Limitation	Corresponding Claim Limitation
[13.1]	[1.2] and [1.3]
[13.2]	[1.4] and [1.5]
[13.3]	[1.6]
[13.4]	[1.7]
[13.5]	[1.8]

De Castro renders obvious these limitations for the same reasons discussed above. Ex.1003, ¶¶286-290.

[13.0] A non-transitory computer readable storage medium comprising program instructions, wherein when executed the program instructions are operable to:

De Castro teaches that “the communication application server 200 shown in Fig.14 supports a Java virtual machine that **maintains a number of software modules** 250....” Ex.1008, [0123]; Ex.1003, ¶¶282-283. It would have been obvious to a POSITA for de Castro’s software modules to be embodied in a “*non-transitory computer readable storage medium comprising program instructions*” because that is how computer software was commonly stored. Ex.1003, ¶284. Thus, it would have been obvious to a POSITA that de Castro’s App server would include a “*non-transitory computer readable storage medium comprising program instructions*” to operate the software modules of the App server and to perform the

functions of the App server as described above at Sections XI.E.2.[1.0]-[1.8].

Ex.1003, ¶285.

8. Claim 14

Claim 14 is substantively identical to claim 2, already addressed above. De Castro renders obvious this claim for the same reasons discussed above. Ex.1003, ¶¶291-292.

XII. DISCRETIONARY DENIAL IS INAPPROPRIATE

A. No basis for 35 U.S.C. § 325(d) denial

None of Bodart, Banister, Dharmarajan, or de Castro were cited or considered by the Examiner during prosecution of the '254 patent.

Accordingly, discretionary denial under 35 U.S.C. § 325(d) is not appropriate because the first part of the *Advanced Bionics* framework is not met. *Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH*, IPR2019-01469, Paper 6 at 8 (PTAB Feb. 13, 2020) (precedential).

B. Discretionary denial under the *Fintiv* factors is not appropriate

The six factors for considering denial under § 314 strongly favor institution. *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (Mar. 20, 2020) (precedential). The district court case is in its early stages. Patent Owner filed its complaint on May 20, 2022 (Ex.1026) and served Cisco on May 27, 2022. Patent Owner provided its preliminary infringement contentions on September 22, 2022.

Ex.1025. Petitioner has since worked diligently to prepare this Petition, which is being filed well within the one-year timeframe allowed by Congress.

1. Whether evidence exists that a stay may be granted if a proceeding is instituted

A motion to stay has not yet been filed and the Board should not infer or attempt to predict the outcome if such a motion is later filed. *Sand Revolution II LLC v. Continental Intermodal Group – Trucking LLC*, IPR2019-01393, Paper 24 at 7 (June 16, 2020) (informative); *see also Dish Network L.L.C. v. Broadband iTV, Inc.*, IPR2020-01359, Paper 15 (Feb. 12, 2021) (“It would be improper to speculate, at this stage, what the Texas court might do regarding a motion to stay...”). Thus, this factor is neutral.

2. Proximity of the court’s trial date to the board’s projected statutory deadline for a final written decision

The operative Docket Control Order sets July 13, 2023, for the *Markman* hearing and January 22, 2024, for Jury Selection. Ex.1024, p. 3. The Final Written Decision is expected in this case in August 2024. This factor nevertheless weighs slightly against discretionary denial, or at the very least is neutral, because the Petitioner has worked expeditiously to prepare and file this petition just over four months after receiving Patent Owner’s infringement contentions on September 22, 2022. Ex.1025; *Fintiv*, Paper 11 at 11-12.

Furthermore, the current average time-to-trial for the district court hearing the co-pending case is 24.5 months. Ex.1023, p. 35; *In re Apple Inc.*, 979 F.3d 1332, 1350 (Fed. Cir. 2020). Thus, a trial in the co-pending case would be expected in June 2024. A Final Written Decision in this IPR is expected by August 2024.

As such, this factor weighs slightly against discretionary denial, or at the very least is neutral. *See Sand Revolution II, LLC v. Continental Intermodal Grp.—Trucking LLC*, IPR2020-01393, Paper 24 (June 16, 2020) (panel granting institution on rehearing based on finding that uncertainty over district court’s trial date weighed against discretionary denial).

3. Investment in the parallel proceeding

The co-pending litigation is in its early stages, and the investment in it has been minimal. Fact discovery is not set to close until September 5, 2023, and expert discovery is not set to close until October 10, 2023. Ex.1024. The Docket Control Order sets July 13, 2023 as the date for the *Markman* Hearing. Ex.1024, pp. 3-4. Thus, this factor weights against discretionary denial. *See PEAG LLC v. Varta Microbattery GMBH*, IPR2020-01214, Paper 8 at 17 (Jan. 6, 2021) (finding that since no claim construction hearing had yet been held and discovery was not completed, the little investment in the parallel proceeding weighed against discretionary denial).

4. Overlap between issues raised in the petition and in the parallel proceeding

There will be no overlap of prior art issues. Petitioner stipulates that, if the Board institutes trial, Petitioner will cease asserting in the co-pending litigation the combination of references on which trial is instituted for the claims, to the extent Petitioner even asserts the same combination in the district court. This factor weighs against discretionary denial. *See Verizon v. Huawei*, IPR2020-01079, Paper 10 at 38 (Jan. 14, 2021) (finding a similar stipulation “mitigates the concern about overlapping issues” and weighs “against discretionary denial of the Petition.”).

5. Whether the petitioner is a defendant

Petitioner is a defendant in the litigation. Ex.1026. This is true of most Petitioners in IPR proceedings. While the *Fintiv* case indicates that a difference between the district court defendant and the petitioner may weigh against discretionary denial, nothing within the *Fintiv* case suggests that the same party between the proceedings weighs in favor of discretionary denial. *See HP Inc. v. Slingshot Printing LLC*, IPR2020-01084, Paper 13 at 9 (having the “same parties as parallel proceeding” makes factor 5 “neutral”). Accordingly, this factor is neutral and should not be a basis for denying institution.

6. Other circumstances that impact the Board’s exercise of

discretion, including the merits

As discussed above, the prior art presented in this Petition renders the Challenged Claims unpatentable as obvious. In view of the striking similarities between the '254 patent claims and both the Bodart and de Castro primary prior art references, this petition “presents compelling evidence of unpatentability.” Director Vidal’s Guidance Memorandum, Ex.1027, 2. This factor weighs against discretionary denial.

As such, the *Fintiv* factors are either neutral or weigh against discretionary denial. Because this Petition was filed more than three months before the statutory bar date, institution should not be denied on discretionary factors.

C. Discretionary denial under *General Plastic* is not appropriate

The '254 patent has not been challenged in any prior IPR petition, so no *General Plastic* discretionary institution factors apply here. *See General Plastic Indus. Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19 at 16 (Sept. 6, 2016) (Section II.B.4.i. precedential).

XIII. CONCLUSION

Petitioner has established a reasonable likelihood that the Challenged Claims are unpatentable.

Respectfully submitted,

Dated: February 2, 2023
HAYNES AND BOONE, LLP
2323 Victory Avenue, Suite 700
Dallas, Texas 75219
Customer No. 27683

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456

XIV. MANDATORY NOTICES

A. Real party-in-interest

Pursuant to 37 C.F.R. §42.8(b)(1), Petitioner certifies the real party-in-interest is Cisco Systems, Inc.

B. Related matters

Pursuant to 37 C.F.R. §42.8(b)(2), to Petitioner's best knowledge, the '254 patent is or was involved in the following cases:

Case Heading	Number	Court	Filed
<i>Zilkr Cloud Technologies, LLC v. Cisco Systems, Inc.</i>	2-22-cv-00166	E.D. Tex.	May 20, 2022
<i>Zilkr Cloud Technologies, LLC v. RingCentral, Inc.</i>	3-21-cv-02807	N.D. Tex.	Nov. 11, 2021

C. Lead and back-up counsel and service information

Lead Counsel

Theodore M. Foster
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (303) 382-6205
Fax: (214) 200-0853
ipr.theo.foster@haynesboone.com
USPTO Reg. No. 57,456

Back-up Counsel

David L. McCombs
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (214) 651-5533
Fax: (214) 200-0853
david.mccombs.ipr@haynesboone.com
USPTO Reg. No. 32,271

Eugene Goryunov
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (312) 216-1630
Fax: (214) 200-0853
eugene.goryunov.ipr@haynesboone.com
USPTO Reg. No. 61,579

Samuel M. Drezdron
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: 972-739-6918
Fax: 214-200-0853
samuel.drezdron.ipr@haynesboone.com
USPTO Reg. No. 67,085

Brett Bostrom
HAYNES AND BOONE, LLP
2323 Victory Ave. Suite 700
Dallas, TX 75219

Phone: (972) 739-6973
Fax: (202) 654-4501
brett.bostrom.ipr@haynesboone.com
USPTO Reg. No. 73,009

Please address all correspondence in this proceeding to lead and back-up counsel. Petitioner consents to service in this proceeding by email at the addresses above.

APPENDIX OF THE CHALLENGED CLAIMS

Claim limitation identifier	Corresponding claim language
[1.0]	<i>A telephone system, comprising:</i>
[1.1]	<i>a database; and</i>
[1.2]	<i>a server coupled to the database,</i>
[1.3]	<i>wherein the server is utilized by a first provider to provide a first service to a first user, wherein the first service is telephone service, and wherein the telephone number is associated with a first telephone number;</i>
[1.4]	<i>wherein the server is configured to: receive a request to provision a second service for the first user,</i>
[1.5]	<i>wherein the second service is provided by a second provider, and wherein the second provider is different than the first provider;</i>
[1.6]	<i>responsive to receiving the request: create a mapping from the first telephone number to a second identifier (ID), wherein the second ID is utilized by the second provider to identify the first user;</i>

[1.7]	<i>activate the second service with the second provider utilizing the second ID as a user ID for the first user; and</i>
[1.8]	<i>associate the second service with the first telephone number, wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.</i>
[2.0]	<i>The telephone system as recited in claim 1, wherein the server is further configured to associate a plurality of services with the first telephone number,</i>
[2.1]	<i>wherein the plurality of services are provided by a plurality of providers different than the first provider, and wherein the plurality of services are utilized by the first user.</i>
[6.0]	<i>The telephone system as recited in claim 1, wherein the second service is a conferencing application, voice attendant application, or text messaging application.</i>
[7.0]	<i>A method comprising:</i>
[7.1]	<i>providing a first service to a first user by a first provider, wherein the first service is telephone service, and wherein the first user has a first telephone number;</i>

[7.2]	<i>receiving a request to provision a second service for the first user, wherein the second service is provided by a second provider, and wherein the second provider is different than the first provider;</i>
[7.3]	<i>responsive to receiving the request: creating a mapping from the first telephone number to a second identifier (ID), wherein the second ID is utilized by the second provider to identify the first user;</i>
[7.4]	<i>activating the second service with the second provider utilizing the second ID as a user ID for the first user; and</i>
[7.5]	<i>associating the second service with the first telephone number, wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.</i>
[8.0]	<i>The method as recited in claim 7, further comprising associating a plurality of services with the first telephone number,</i>

[8.1]	<i>wherein the plurality of services are provided by a plurality of providers different than the first provider, and wherein the plurality of services are utilized by the first user.</i>
[12.0]	<i>The method as recited in claim 7, wherein the second service is a conferencing application, voice attendant application, or text messaging application.</i>
[13.0]	<i>A non-transitory computer readable storage medium comprising program instructions, wherein when executed the program instructions are operable to:</i>
[13.1]	<i>provide a first service to a first user by a first provider, wherein the first service is telephone service, and wherein the first user has a first telephone number;</i>
[13.2]	<i>receive a request to provision a second service for the first user, wherein the second service is provided by a second provider, and wherein the second provider is different than the first provider;</i>
[13.3]	<i>responsive to receiving the request: create a mapping from the first telephone number to a second identifier (ID),</i>

	<i>wherein the second ID is utilized by the second provider to identify the first user;</i>
[13.4]	<i>activate the second service with the second provider utilizing the second ID as a user ID for the first user; and</i>
[13.5]	<i>associate the second service with the first telephone number, wherein the first telephone number is associated with routing and interworking data necessary to utilize the second service.</i>
[14.0]	<i>The non-transitory computer readable storage medium as recited in claim 13, wherein the program instructions are further operable to associate a plurality of services with the first telephone number,</i>
[14.1]	<i>wherein the plurality of services are provided by a plurality of providers different than the first provider, and wherein the plurality of services are utilized by the first user.</i>

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioner hereby certifies, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 13,585. Pursuant to 37 C.F.R. § 42.24(d), this word count excludes the table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing.

Dated: February 2, 2023

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456

CERTIFICATE OF SERVICE

The undersigned certifies that, in accordance with 37 C.F.R. § 42.6(e) and 37 C.F.R. § 42.105, service was made on Patent Owner as detailed below.

Date of service February 2, 2023

Manner of service PRIORITY MAIL EXPRESS

Documents served Petition for *Inter Partes* Review Under 35 U.S.C. § 312 and 37 C.F.R. § 42.104 of U.S. 9,210,254;
Petitioner's Exhibit List;
Exhibits 1001-1027.

Persons served Conley Rose, P.C.
4965 Preston Park Blvd., Suite 195E
Plano, TX 75093

/Theodore M. Foster/
Theodore M. Foster
Lead Counsel for Petitioner
Registration No. 57,456