UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

DIALECT, LLC,
Patent Owner.

IPR2025-01193
Patent 7,917,367 B2
Issued: March 29, 2011
Application No. 12/617,506
Filed: November 12, 2009

Title: SYSTEMS AND METHODS FOR RESPONDING TO NATURAL LANGUAGE SPEECH UTTERANCE

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,917,367

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page

Cases

Board Decisions

Statutes

Rules

LIST OF EXHIBITS

No.	Description
1001	U.S. Patent No. 7,917,367 ("**367 patent**")
1002	File History of U.S. Patent No. 7,917,367
1003	Declaration of Paul Jacobs, dated July 18, 2025 ("**Jacobs Decl.**" or "**Jacobs**")
1004	U.S. Patent Application Pub. No. 2002/0059425 to Belfiore et al. ("**Belfiore**")
1005	U.S. Patent Application Pub. No. 2004/0044516 to Kennewick et al. ("**Kennewick**")
1006	Redline showing differences between text, aside from the claims, of Kennewick (EX1005) and 367 patent (EX1001).
1007	U.S. Patent Application Pub. No. 2002/0133354 to Ross et al. ("**Ross**")
1008	Order (Construing Claim Terms), *Dialect, LLC v. Amazon.com Inc. et al.*, Civil No. 1:23cv581 (DJN) (E.D. Va. Apr. 29, 2024)
1009	Complaint, *Dialect, LLC v. Microsoft Corp.*, Case 2:24-cv-01067-JRG (E.D. Tex. Dec. 20, 2024)
1010	Akiyoshi Ochi, *et al.*, *Network Applications for Mobile Computing*, Fujitsu Sci. Tech. J., 34(1), 41-49 (Sept. 1998)
1011	Robin Garner, *Pelican DHCP Automated Self-Registration System: Distributed Registration and Centralized Management*, Proc. of the LISA 2001 15th Sys. Admin. Conf., 257-266 (Dec. 2001)
1012	Adam Rifkin & Rohit Khare, *The Evolution of Internet-Scale Event Notification Services*, Workshop on Internet-Scale Event Notification (July 13, 1998)
1013	Antonio Carzaniga, *et al.*, *Achieving Scalability and Expressiveness in an Internet-Scale Event Notification Service*,

No.	Description
	PODC '00: Proc. Nineteenth ACM Symp. Principles of Distributed Computing, 219-227 (2000)
1014	Frederick Jelinek, *The Development of an Experimental Discrete Dictation Recognizer*, Proc. of the IEEE, 73(11), 1616-1624 (1985)
1015	Andreas Stolcke, *et al.*, *Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech*, Computational Linguistics, 26(3), 339-373 (Sept. 2000)
1016	E. Levin, *et al.*, *The AT&T-DARPA Communicator Mixed-Initiative Spoken Dialog System*, Sixth Int'l Conf. on Spoken Language Processing (2000)
1017	International Pub. No. WO 01/78065 to Weber et al.
1018	Appendix H to U.S. Patent No. 7,917,367
1019	Andrew S. Tanenbaum, *Computer Networks* (Pearson Education, Inc., 4th ed. 2003)
1020	L. Rau et al., *Information Extraction And Text Summarization Using Linguistic Knowledge Acquisition*, Information Processing & Management Vol. 25, No. 4, pp. 419-428 (1989)
1021	Paul S Jacobs, *Joining Statistics with NLP for Text Categorization*, Third Conference on Applied Natural Language Processing, Association for Computational Linguistics, pp. 178-185, Mar. 31, 1992-Apr. 3, 1992
1022	List of Challenged Claims - USP 7,917,367
1023	Paul S. Jacobs, *A Knowledge Framework for Natural Language Analysis*, IJCAI 87 Proceedings of the Tenth International Conference on Artificial Intelligence, 675-678 (1987)

MANDATORY NOTICES UNDER 37 C.F.R. § 42.8

1. Real Party-In-Interest

Microsoft Corporation is the sole real party-in-interest.

2. Related Matters

The ’367 patent (EX1001) is at issue in the following proceedings:

- *Dialect, LLC v. Microsoft Corporation*, Case No. 2:24-cv-01067 (E.D. Tex.) filed December 20, 2024; and
- *Dialect, LLC v. Samsung Electronics Co., Ltd. et. al.*, Case No. 2:23-cv-00061 (E.D. Tex.) filed February 17, 2023.

The following IPRs involve a patent (USP 7,640,160) to which the ’367 patent alleges priority:

- *Google LLC v. Dialect, LLC*, IPR2024-00753 (instituted)
- *Microsoft Corp. v. Dialect, LLC*, IPR2025-00655 (pending institution)

The following IPR involves a patent (USP 9,263,039) that alleges priority to the ’367 patent:

- *Microsoft Corp. v. Dialect, LLC*, IPR2025-00657 (pending institution)

3. Lead And Back-Up Counsel, And Service Information

Lead Counsel	Back-up Counsel
Andrew M. Mason, Reg. No. 64,034 andrew.mason@klarquist.com	Cameron D. Clawson, Reg. No. 73,509 cameron.clawson@klarquist.com

Lead Counsel	Back-up Counsel
	Amy Haspel, Reg. No. 78,385 amy.haspel@klarquist.com Frank Morton-Park, Reg. No. 80,750 frank.morton-park@klarquist.com
KLARQUIST SPARKMAN, LLP 121 SW Salmon Street, Suite 1600 Portland, Oregon, 97204 503-595-5300 (phone) 503-595-5301 (fax)	

Petitioner consents to service via email at the above email addresses and the email address of Msft-Dialect@klarquist.com.

Pursuant to 37 C.F.R. § 42.10(b), concurrently filed with this Petition is a Power of Attorney executed by Petitioner and appointing the above counsel.

Petitioner authorizes Account No. 02-4550 to be charged for any fees, including those enumerated in 37 C.F.R. § 42.15.

I. INTRODUCTION

Microsoft Corporation ("Petitioner") respectfully requests *inter partes* review ("IPR") of claims 11-12, 15, and 17-18 of U.S. Patent No. 7,917,367 (EX1001, "the '367 patent"), allegedly assigned to Dialect, LLC ("Patent Owner").

The challenged claims relate to synchronizing "context" across plural user devices, in a system that processes natural language inputs. Prior art Belfiore (EX1004) taught a method nearly identical to that recited in '367 patent claims 11-12 and 15, and renders those claims obvious. Prior art Kennewick, in turn, also teaches synchronizing a user's multiple devices and contains additional details on NLP processing that satisfy claims 17-18 of the '367 patent. Indeed, an ancestor patent claim that is materially indistinct from the features recited in challenged claim 17 is currently under IPR on Kennewick-based grounds. *See* IPR2024-00753. Together, Kennewick and Belfiore render obvious all challenged claims. For the reasons forth below, the Board should institute IPR and find the challenged claims unpatentable.

II. GROUNDS FOR STANDING PER 37 C.F.R. § 42.104 (a)

Petitioner certifies that the '367 patent is available for IPR and that Petitioner is not barred or estopped from requesting an IPR on the grounds presented herein.

III. IDENTIFICATION OF CHALLENGE

Petitioner requests *inter partes* review of claims 11-12, 15, and 17-18 (the "Challenged Claims") of the '367 patent, on the following statutory grounds:

	Reference(s)	Basis	Claims
Ground 1	Belfiore (EX1004)[1]	§ 103	11, 12, 15
Ground 2	Kennewick (EX1005)[2] and Belfiore	§ 103	11-12, 15, 17-18
Ground 3	Kennewick, Belfiore, Ross (EX1007)[3]	§ 103	17-18

The Petition presents evidence showing a reasonable likelihood that the Petitioner will prevail in establishing that each Challenged Claim is unpatentable.

IV. STATE OF THE ART

A. Networked Systems Regularly Used Some Form Of Device Registration To Track And Communicate With Devices

By at least 1998, it was well known that network communications involved tracking various user devices, to properly communicate with them. EX1010, 43 ("A mechanism is required to verify users attempting to access the network."), 47 ("A mobile computer…requires registration in the mainframe computer beforehand."). Sometimes referred to as registration or authentication or both, this functionality has

[1] U.S. Patent Application Pub. No. 2002/0059425.

[2] U.S. Patent Application Pub. No. 2004/0044516.

[3] U.S. Patent Application Pub. No. 2002/0133354.

long allowed networks to ensure that messages are getting to and coming from the correct devices, and only such devices. EX1019, 282 ("When a new host enters an area … his computer must register itself with the foreign agent there."); *see also id.*, 353; EX1011, 262 (describing registration process that includes verifying username and password submitted by the device before registration); *see also id.*, 257-263 (describing detailed examples of registration workflows and communications); Jacobs, ¶¶27-32.

It was also well known to "deregister" devices upon their departure from the network or when a device's registration "timed out" or expired after a certain period of time. EX1019, 283 ("Ideally, when a host leaves an area, that, too, should be announced to allow deregistration."), 354 ("mak[ing] registration valid only for a fixed time interval" to address "mobile hosts that leave without saying goodbye"); EX1011, 263 ("Automated client expiration … deletes any client whose registration has expired."); Jacobs, ¶31.

B. Event Subscription Services Were Commonly Used To Update Network Devices Or Components With New Information

Since as early as 1998, event-based notifications were commonly used to facilitate changing the state of networked devices based on certain events and, relatedly, synchronizing the state of different devices on the network. *E.g.,* EX1012, 15 (describing examples of event notifications for "maintaining awareness of people, devices”), 19); *id.*, 2 (describing use of “Event Notification Services (ENS)”). These services typically allowed clients to subscribe to events of interest and receive notifications of published events from other clients. *Id.*, 28 (describing event lifecycle as including “[e]xpression of *interest* in an event,” “[o]ccurrence of *an event*,” “[r]elation of *an event to a pattern of interest*,” and “[n]otification to *an application*”) (emphasis added); EX1013, 219, Figure 1 (event notification service with “advertise,” “subscribe,” “publish,” “notify” model.); *see also* Jacobs, ¶¶33-34.

C. Conventional Natural Language Speech Processing Used Context

POSITAs understood that natural language speech processing typically used contextual information to facilitate the recognition of the speech input. Jacobs, ¶35. As early as 1985, n-gram language models predicted a next word in a sequence based on previous word(s). *E.g.*, EX1014, 1622 (computing “probabilities” based on “the preceding two words”); *id.*, 1619 (these contextual techniques led to more “efficient” speech recognition). By at least 2005, well-known NLP systems used a given dialog’s context to improve accuracy and efficiency. *E.g.,* EX1015, 340; EX1016, 1, 3 (increasing performance by constraining grammars to “expected” inputs); Jacobs, ¶¶35-36.

It was also well known to use context-specific grammars corresponding to a topic of discussion (e.g., “news,” “weather,” “stocks,” etc.) determined from prior user input. *E.g.,* EX1017, 10:22-11:1 (“examples of contexts may be ‘news’, or 'weather', or 'stocks'."), 10:7-8 ("For example, the statement 'I would like to see movies,' would be linked to the context of 'movies' or 'film.'"), 20:4-6 ("when the next utterance is provided … it would search the enabled context-specific grammar 212 for 'movies' before searching the general grammar 214."); Jacobs, ¶37.

V. THE '367 PATENT

The '367 patent issued March 29, 2011, from an application filed November 12, 2009, and alleges priority to an application filed August 5, 2005. EX1001, 1; *see also* Jacobs, ¶38.

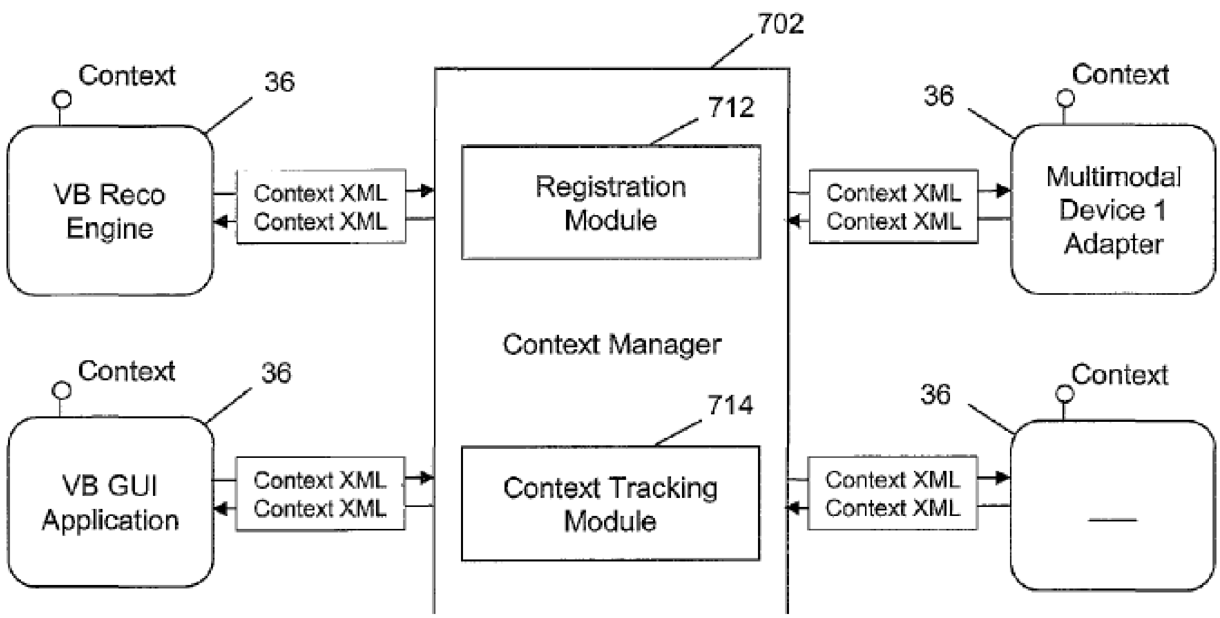
The '367 patent describes a system including a main unit, a speech unit, and a multi-modal device. EX1001, 11:36-37. The main unit may process input commands and questions to create a response string for presentation via the speech unit or multi-modal device. EX1001, 12:56-64, 13:7-22; *see also* Jacobs, ¶39.

In one embodiment, a "context manager 702" communicates with "multiple mobile devices 36." EX1001, 13:23-27. The multiple devices may "register via registration module 712" and "registration may indicate events that the mobile devices 36 may subscribe to." *Id.*, 13:26-29. "[M]obile devices 36 may be informed of a context changes [sic] through context tracking module 714, thereby synchronizing the context across the registered mobile devices 36." *Id.*, 13:31-34; *see also* Jacobs, ¶40.

FIG. 7 depicts the "registration module 712" and "context tracking module 714" as black boxes within the "context manager 702" and the patent provides no meaningful detail on the structure of these modules or how they carry out their functions.

FIG. 7

700



EX1001, FIG. 7; *see also* Jacobs, ¶41.

A. Prosecution History Of The '367 Patent

Prosecution was brief. On November 1, 2010, the Office allowed all original claims without having issued an Office action or rejected any claims based on prior art. EX1002, 96, 99-101. The Examiner found that the "closest prior art of record" ("Mumick") disclosed "synchronizing plural browsers on a plurality of mobile devices" but that "the synchronization of a navigation state as taught by Mumick et al. cannot be said to teach or suggest the claimed synchronization of context across the plurality of devices." *Id.*, 100. The Examiner stated that "'context' refers to time, location, numbers, dates, categories (e.g., music, movies, television, addresses, etc.) and other context." *Id.* (citing *id.*, 32 (Appl., p. 28 ¶ 64, which corresponds to '367 patent, 14:4-31)). The Notice of Allowability cited several other references but did not compare any of them to the claims. *Id.*, 101; *see also* Jacobs, ¶42.

After applicants filed an RCE and IDS, a second Notice of Allowability issued December 20, 2010, reiterating "the reasons given in the previous Office Action." EX1002, 784-88. A Supplemental Notice of Allowability issued February 22, 2011, after an amendment adding "processor" language to claims 1-5 and 8. *Id.*, 833-835, 843-848.

B. Claims Listing

A full claims listing is provided as EX1022. Exemplary claim 11 is listed here:

[11.1] A method for processing multi-modal natural language inputs, comprising:
[11.2] registering a plurality of mobile devices with a context manager in response to a registration module associated with the context manager receiving a communication from the plurality of mobile devices;
[11.3] subscribing the plurality of mobile devices registered with the context manager to one or more context events;
[11.4] receiving, at the context manager, a context input from one or more of the plurality of mobile devices registered with the context manager, wherein the context input includes a context change event; and
[11.5] informing the plurality of mobile devices registered with the context manager of the context change event,

wherein informing the plurality of mobile devices registered with the context manager of the context change event synchronizes a context across the plurality of mobile devices.

VI. LEVEL OF SKILL IN THE ART

A POSITA at the earliest alleged priority date in August 2005 would have had a bachelor's degree in electrical engineering, computer science, computer engineering, or equivalent field, and two years of experience working with speech recognition and natural language processing systems. Additional work experience could make up for less education and vice versa. Jacobs, ¶43.

VII. CLAIM CONSTRUCTION

Unless otherwise discussed below, Petitioner applies the plain and ordinary meaning of all claim terms. *See also*, Jacobs, ¶44. Petitioner reserves the right to argue in any district court case or other proceeding that terms in the '367 patent are indefinite, and to raise additional issues of claim construction.

A. Claim Construction Order For Related '039 Patent

Certain claim terms appearing in claims 17 and 18 are similar to or the same as terms in claims of related U.S. Pat. No. 9,263,039. As shown in the table below, these terms in the '039 patent were construed as part of a district court litigation between Patent Owner and third party Amazon.com. Without conceding the correctness of those constructions, Petitioner notes that they are satisfied by the prior art presented herein. Jacobs, ¶¶44-45.

Term	Construction for '039 patent in E.D. Va.
context	"the subject matter to which a particular user input is directed and which is used to determine the meaning of the user input" (EX1008, 1)
context stack	"a data structure that stores or references contexts in a manner that can be ordered" (EX1008, 1)
context description grammar	"a data structure containing entries constituting or referencing sets of rules, wherein each of those sets describes the structure of natural language in a particular context" (EX1008, 2)

B. Means-Plus-Function Analysis

For purposes of this IPR only, Petitioner submits that no terms invoke means-plus-function analysis under Section 112(6).

To the extent Patent Owner argues that "**registration module**" invokes means-plus-function analysis[4] and that the corresponding structure is a computer programmed to register devices that communicate with the context manager (*e.g.*,

[4] For example, Patent Owner might argue that "module" is a nonce word connoting no structure, that "registration" merely recites functionality not structure, and that the surrounding claim language imparts no structure. Petitioner reserves the right to argue that "registration module" is a means-plus-function term that lacks corresponding linked structure because the '367 patent provides no linked algorithm or other structure for performing the recited "registration" function. *E.g.*, *Rain Computing, Inc. v. Samsung Elecs. Am., Inc.*, 989 F.3d 1002, 1006–07 (Fed. Cir. 2021) (finding method claims' "user identification module" a MPF term).

’367 patent, 3:34-40), the Grounds below show how the challenged claims are obvious under such a construction.

To the extent Patent Owner argues that “**context manager**” invokes means-plus-function analysis and that the corresponding structure is a computer programmed to manage context across devices, the Grounds below show how the challenged claims are unpatentable under such a construction.[5]

VIII. GROUND 1: CLAIMS 11-12, 15 ARE OBVIOUS OVER BELFIORE

Claims 11-12 and 15 are obvious over Belfiore. Indeed, Belfiore claims 21-35 describe systems and methods strikingly similar to the methods later claimed in ’367 patent claims 11-12 and 15. Jacobs, ¶¶46-87.

A. Belfiore (EX1004)

Belfiore (EX1004) published May 16, 2002, and qualifies as prior art under at least 35 U.S.C. § 102(b). EX1004, 1. Microsoft Corporation is the named assignee. *Id.* Belfiore is analogous art to the ’367 patent at least because it describes systems and methods for improved distributed computing services, including synchronizing of user and session information across mobile devices with multi-modal natural

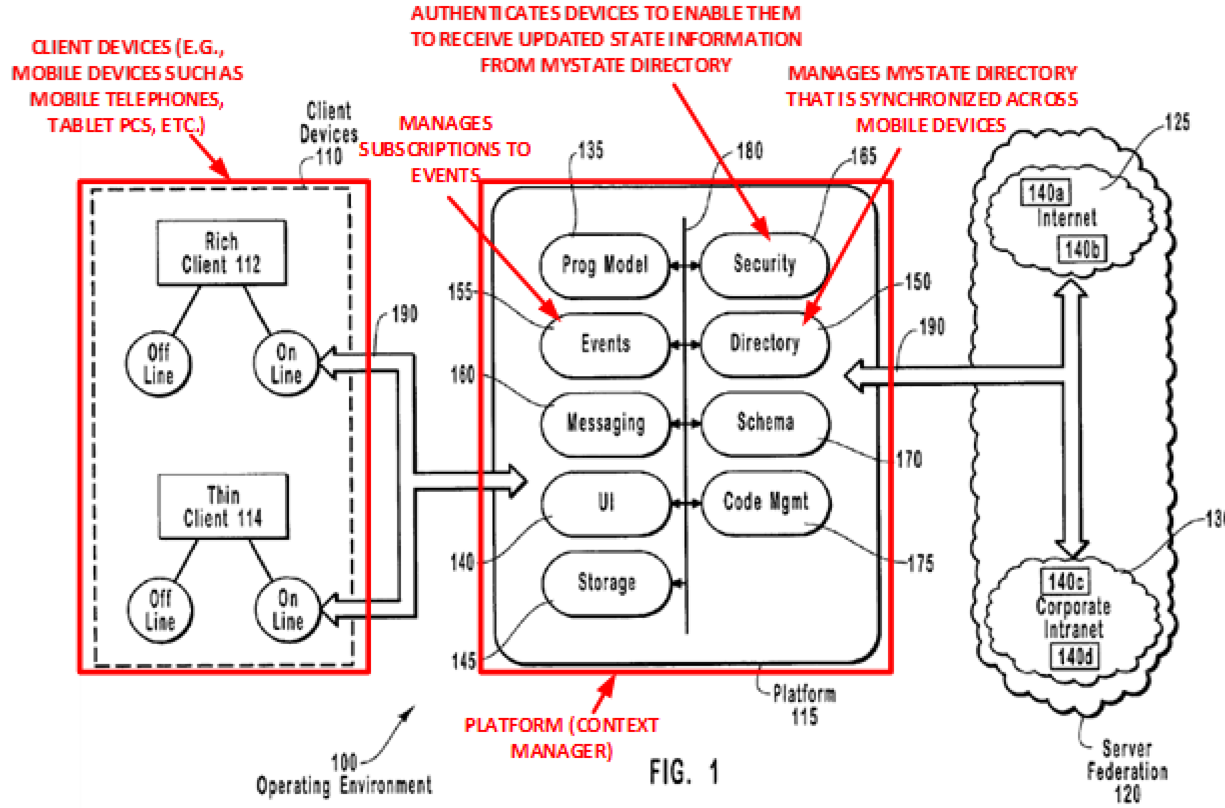
[5] Petitioner reserves the right to argue that “context manager” is a means-plus-function term that lacks corresponding linked structure.

language user interfaces. *E.g.*, *Id.*, Title, [0003], [0019], claims 21-35; Jacobs, ¶¶47-51.

Belfiore discloses "a distributed computing services platform" that *inter alia* ***"perform[s] a natural language query or command"*** (EX1004, [0104-06]) and s***ynchronizes devices*** to provide "more consistent user interaction" and "more personalized, relevant information" (*id.*, [0003], [0014], [0109]). Belfiore's platform employs a ***"multi-modal" user interface*** that provides "typed (keyboard), spoken, [and] hand-written" input/output types. *Id.*, [0118].

Belfiore teaches synchronizing session status and other information across a user's multiple devices, such that the user interface "adapts and/or changes . . . across a number of client devices 110." *Id.*, [0097]. ***Synchronization occurs through events subscribed to by a user*** (*id.*, [0020]) and Belfiore's event component "enables software components and architectures to have access to ***continually updated information about their context***" (EX1004, [0119] (emphasis added)).

Belfiore's platform 115 and its components (e.g., user interface ("UI") 140, directory 150, events 155, security 165, etc.) appear in FIG. 1:



EX1004, FIG. 1 (red annotations added); *see also* Jacobs, ¶48.

Belfiore's client devices include mobile devices, such as mobile telephones, tablet PCs, etc. EX1004, [0093], [0099]. The security component 165 authenticates each client device and then "the preference and session information are provided to the new client device." *Id.*, [0098]. The preference and session information are updated and maintained by a "directory component" using, e.g., a MyState directory, via events managed by the event component. *Id.* ("As a user works on a client device, the session status information is regularly updated in the MyState directory 905."); *see also id.*, [0203]; *id.*, [0184] ("event component 155 and messaging component

160" notify "other entities of changes and updates to the MyState directory…"); Jacobs, ¶49.

Belfiore claim 21 and its dependent claims describe synchronizing user information and session status across multiple mobile devices as part of "a method for facilitating a mid-session transition between [a] first client device and [a] second client device." Jacobs, ¶¶54-55. Claim 21 recites these steps:

> retrieving state information stored on at least one server;
> interacting in a session with the first client device resulting in an update of the state information;
> storing the updated information on the at least one server;
> retrieving the updated state information when the session is continued on the second client device; and
> interacting in the continued session with the second client device based on the updated state information.

EX1004, cl. 21. Dependent claims specify that the first and second client devices are both mobile telephones (cl. 34-35) and authenticating the user on each mobile device (cl. 22-23).

B. <u>Obvious Implementations Of Belfiore's Teachings</u>

One obvious implementation of Belfiore follows from claim 21 and its dependent claims, as informed by Belfiore's figures and other disclosures. Jacobs, ¶¶52-53, 56-57. That implementation includes:

- At least **two mobile client devices**, such as the "first mobile telephone" and "second mobile telephone" of claims 34 and 35. *See also* EX1004, [0041], [0044], [0099].
- The mobile client devices providing user interaction (*id*., claims 21-23) through a **multi-modal natural language interface**, such as a "unified command line" that carries out "a natural language query or command" (*id*., [0104-05]), where "all forms of input—typed (keyboard), spoken, handwritten may be received by the user interface" (*id.*, [0106]). *See also id.,* [0104] (preferably, "the unified command line would be provided in some form on every type of device").
- A plurality of "**servers**" (*id.*, claim 21).
- A server **storing "state information"** (*id*., claim 21), including "user interface information" (*id*., claim 27), user "**preference information**" (*id*., claim 38; [0019]), and "**session status**" information (*id*., [0097]), preferably in a user's MyState directory (*id*., [0098]).
- A "**platform" that "facilitates communication** between" the mobile client devices and the servers (*e.g.*, *id*., FIG. 1; [0016]), including updating MyState information on the server (*id*., claims 21, 39) and retrieving information from the server to the client devices (*id*., claims 21, 28, 38).

- A security component with authentication module, for **authenticating** a user on **each mobile client device** (*id*., claims 22-23) based on **receiving** from the device **a message** with the user's **Global ID** and/or **"authentication credentials"** (*id*., [0199-202], FIGS. 4, 10).
- A **directory component for *inter alia* keeping track of a user's currently authenticated devices** (claims 22-23), e.g., by adding information on a device a user has logged into and removing that information if a user logs out of the device (EX1004, [0023], [0098]), thus allowing for communications with a user's currently authenticated devices (claim 21).
- An **event component** used to **subscribe each of a user's current devices** (or the software running thereon) to a group subscription providing updates via events indicating a change in information on any device. *E.g.*, EX1004, [0133], [0124], [0136]; *see also id.*, [0114] (event triggered "when an individual logs onto a particular device").
- **Synchronizing state information** between authenticated mobile client devices, e.g., via Belfiore's group subscription. A POSITA would readily appreciate Belfiore's group event subscriptions as an elegant way taught to **update state information** at the server based on a user session at one mobile device (claim 21) and **retrieve updated** state

information to other user devices so that a user may continue the session on those devices (claim 21). Indeed, Belfiore teaches that "events within the platform are used to synchronize." EX1004, [0020]. Using events to carry out state synchronization achieves Belfiore's stated purpose of "facilitating a mid-session transition between" devices (claim 21) and storing updated information to the server and retrieving that information to the second device (claim 21). *See also id.*, [0098] ("As a user works on a client device, the session status information is regularly updated in the MyState directory 905."). Jacobs, ¶56.

The following annotated Belfiore FIG. 4 illustrates this exemplary implementation:

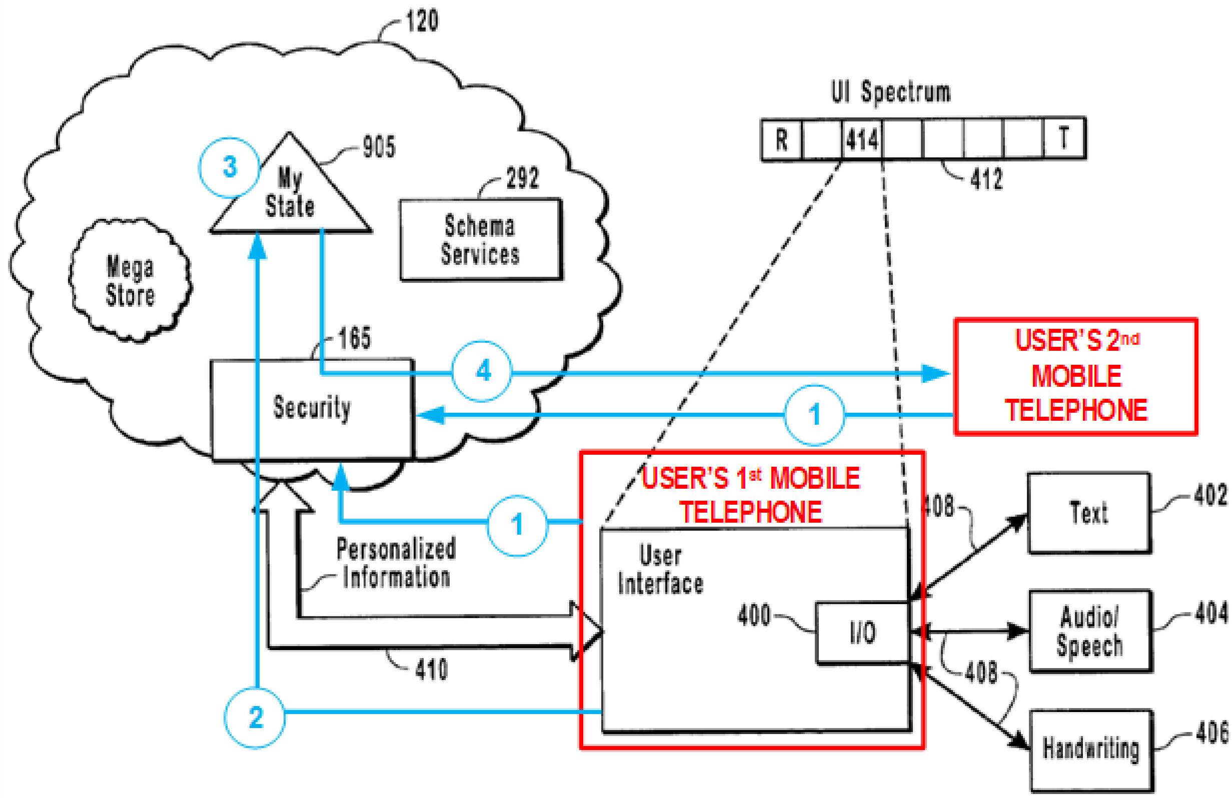


FIG. 4

Circled numbers show process flow for synchronizing context across devices:

1. Devices transmit user log-in/authentication information to platform's security component, which authenticates and registers each device (e.g., claims 22, 23)

2. Update session and other user information, based on user interactions with first mobile telephone (e.g., claims 21, 34)

3. Store the updated user information on the server, e.g., in the user's My State directory (e.g., claim 21)

4. Inform second mobile telephone of the changed user information, thus synchronizing the user's two mobile devices (claims 21, 35)

EX1004, FIG. 4 (red and blue annotations added); *see also* Jacobs, ¶¶52-53.

Belfiore either expressly contemplates these features/steps in a single embodiment, as recited in claim 21 and its dependents, or teaches using them in providing a platform that facilitates the mid-session device-to-device transitions described in Belfiore claim 21. Jacobs, ¶¶57. As explained below, these and other obvious variants of Belfiore render obvious challenged claims 11-12, and 15.

C. Ground 1 Claim Mappings

As set out below, Belfiore renders obvious claims 11-12 and 15.

1. Claim 11

a) Element [11.1] (Preamble)

[11.1][6] A method for processing multi-modal natural language inputs, comprising:

If the preamble is limiting, Belfiore discloses it. Belfiore discloses user interaction through a "user interface (UI)" that "is multi-modal, intelligent and responsive." EX1004, [0019]. This multi-modal interface provides "natural language communication." *Id.*, [0096]; *see also id.*, [0104] (describing "unified command line" used to "perform a natural language query or command)" *Id.*, [0104]; [0105] (describing processing of "natural language command"), [0106] (describing

[6] Reference numbers in the format "[claim#.element#]" added throughout.

parsing the "query or command" using [a]ll levels of intelligence"), [0092] ("speech-to-text technology""), [0093]; Jacobs, ¶59.

The "multi-modal" interface's multiple types of user input include "keyboard entries," "audio/voice input," and "stylus/touch input." EX1004, [0019]; *see also id.*, [0090]; [0091] (describing "multi-modal input and output" such as "text," "audio/speech," "stylus/touch," and "handwriting input."). Belfiore FIG. 4 shows exemplary user inputs 402, 404, 406:

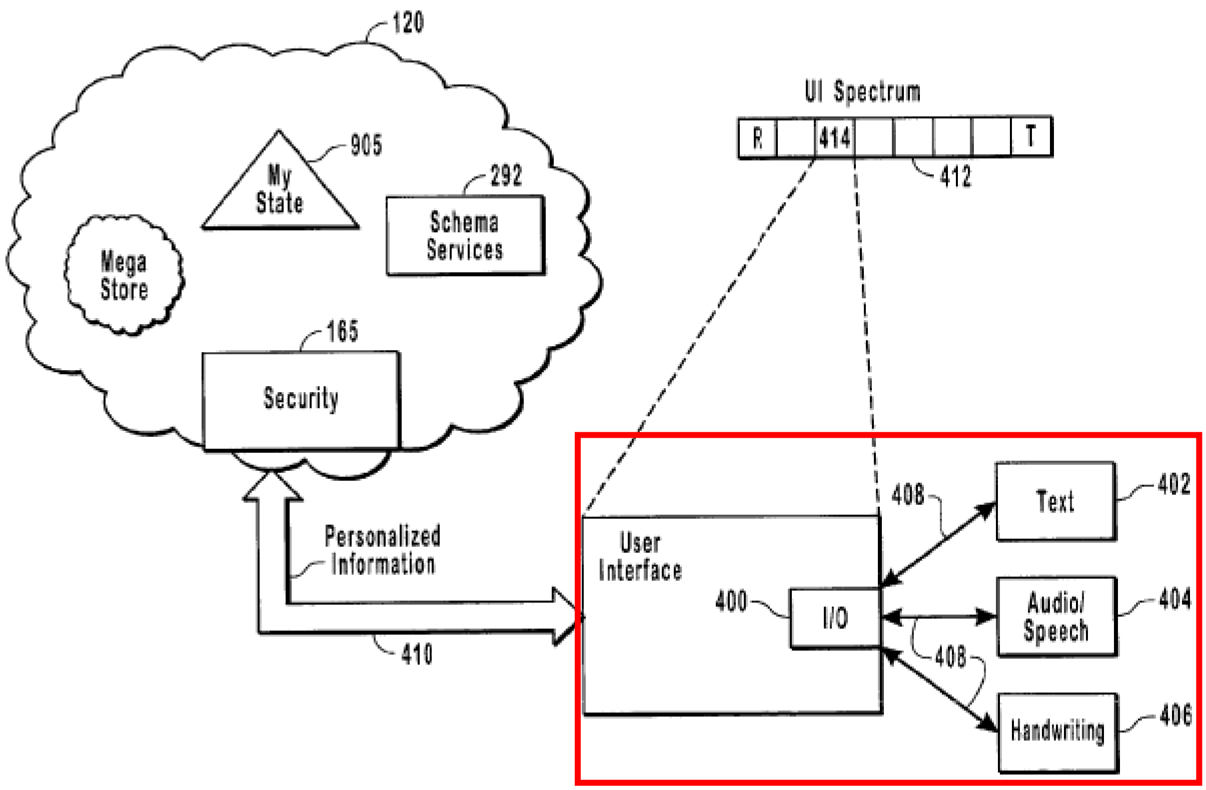


FIG. 4

EX1004, FIG. 4 (red annotations added); *compare with* EX1001, 2:44-54 (describing "multimodal input such as text-based commands and questions and/or voice commands and questions …"); *see also* Jacobs, ¶¶60-61; *supra* Section VIII.B.

b) <u>Element [11.2]</u>

[11.2] registering a plurality of mobile devices with a context manager in response to a registration module associated with the context manager receiving a communication from the plurality of mobile devices;

Belfiore discloses and teaches this element. Belfiore's "distributed computing services platform" or just "platform" is the claimed "context manager" at least because it uses events, MyState, and other features to manage context as a user carries out a session on one or more devices. *Supra* Section VIII.B; EX1004, claims 21-23, [0199-202], FIGS. 4, 10; *see also id.*, [0016]; Jacobs ¶¶62-67. Belfiore claims 34-35 describe a method with two "mobile telephones" and Belfiore elsewhere teaches mobile client devices such as cellular phones, pagers, and other hand-held devices. *E.g.*, EX1004, [0044] ("client devices 110 include" "cellular phones and pagers" as well as "hand-held or Tablet PCs"); *id.*, claims 21, 34-35; [0045-47], FIG. 1; Jacobs, ¶67.

Belfiore satisfies the claimed "registering a plurality of mobile devices …" at least through its authentication process. Belfiore teaches authenticating multiple mobile devices (e.g., two mobile telephones) with a context manager (its platform) after the platform's security component receives a communication from the devices, e.g., a message with a user's Global ID and/or log-in credentials. *Supra* Section VIII.B; EX1004, claims 21-23, [0199-202], FIGS. 4, 10; *see also* Jacobs, ¶¶63-66. This results in the platform associating the devices with a given user, thus allowing the platform to appropriately communicate with the devices. *E.g.*, *supra* Section VIII.B; Jacobs, ¶68. The authentication occurs in response to communications from the mobile devices being received at a registration module associated with the context manager, e.g., at least the platform's "security component" receives ID and/or log-in messages from the mobile devices. *E.g.*, EX1004, [0098]. After completing this authentication, the platform may provide authenticated devices with a user's "preference and session information," e.g., when the user switches from device to device. *Id.*, claims 21-23, [0098], FIGS. 1, 10; *supra* Section VIII.B. This authentication involves:

- The security component receiving messages from the device, such as log-in/authentication/Global ID information. *E.g.*, EX1004, [0109], [0200-202], FIGS. 4, 10.
- The security component establishing the device as "authenticated," so that the platform can provide updated session status and other user information to the device. *E.g.*, EX1004, claims 21-23, [0202]; *supra* Section VIII.B.

- Recording on the platform, such as in a directory, information indicating that the device has been authenticated by the user, to facilitate communications with that device, such as providing updated session status and other user information. *Supra* Section VIII.B.

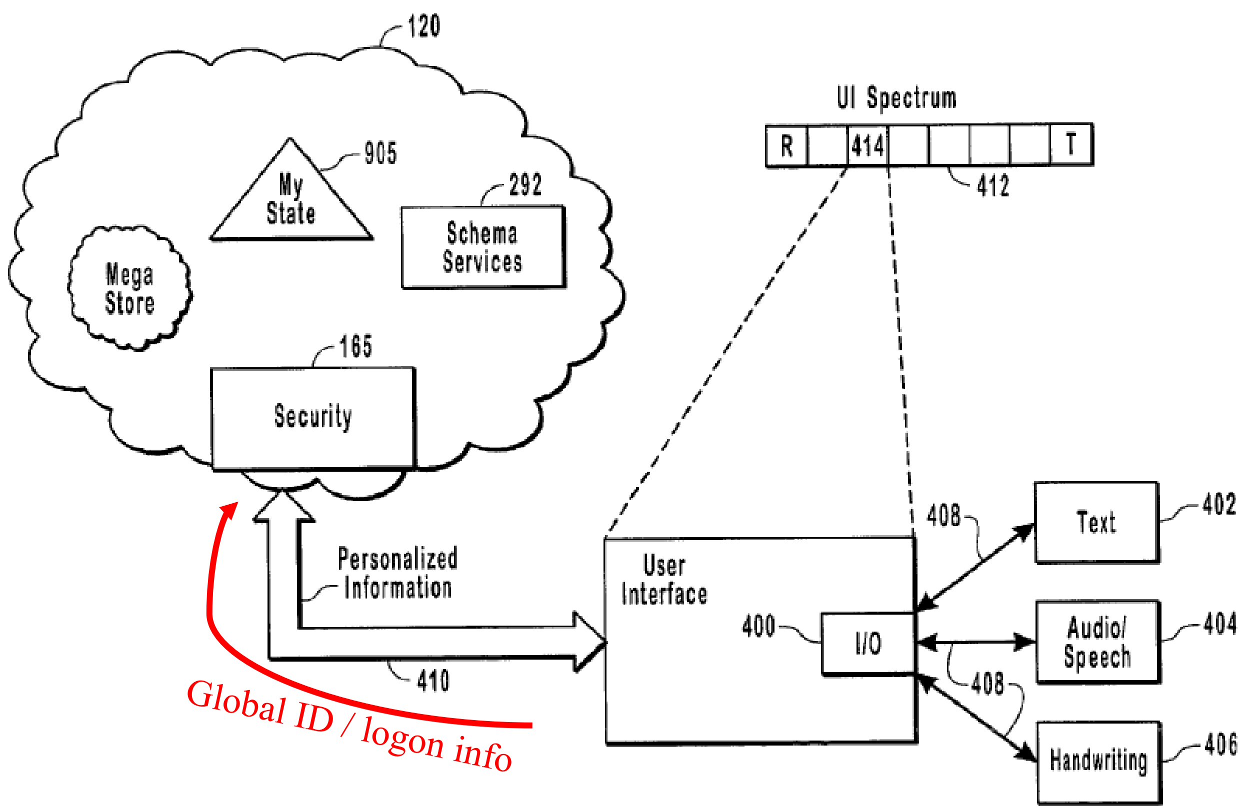


FIG. 4

EX1004, FIG. 4 (red annotations added); *see also supra* Section VIII.B.

The log on / authentication of a given device through the security component registers the device with the platform at least because it allows the platform to securely provide user preference and state information to the device. EX1004, [0175]; *see also id.*, [0109], [0175]; Jacobs, ¶¶69-72.

Consistently, in district court Dialect alleges the claimed "registering …" is satisfied by associating devices with an account and by "Sign in" on a device. EX1018, 8 (mapping this element to description to "*register* a plurality of mobile devices with a context manager *by associating the mobile devices with the user account*") (emphasis added); *see also id.*, 8-9 (mapping "registering" to account "Sign in").

Further confirming that Belfiore satisfies the claimed "registering …," it would have been obvious to implement Belfiore such that user login on a given mobile client device leads to registering that device with the platform. Jacobs, ¶73. Through this registration, the platform would remain aware of the device's authenticated status and able to effectively communicate with the device. This was state of the art for any networked system (particularly for mobile devices), and a POSITA would have thus found it obvious (if not necessary) to register each user device. *Supra* Section IV.A; *see also* EX1019, 282 (describing how a new "mobile host" "must register itself"); *id.*, 282-83, 352-54. Device registration would beneficially allow communications with the mobile devices and facilitate the post-authentication functionality specified by Belfiore, such as receiving / providing preferences, session-status, and other user-associated information from / to authenticated devices (and only authenticated devices). *E.g.*, EX1004, [0098]; *id.*, claims 21-23; *supra* Section VIII.B (explaining obvious implementation in which group of authenticated devices provided updates to one another); Jacobs, ¶73.

For example, registering a device would facilitate that device accessing information from a user's MyState directory (e.g., preferences, session status). *E.g.,* EX1004, [0203]; *see also id.*, claim 21; Jacobs, ¶74. Registration would also beneficially facilitate the system receiving important information back from the device, as also taught by Belfiore. *E.g.*, EX1004, [0098] ("As a user works on a client device, the session status information is regularly updated in the MyState directory 905."), [0184] (describing notifying others of "changes and updates to the MyState directory"), claim 21.

A POSITA would have had a reasonable expectation of success in implementing the Belfiore system to register client devices upon login from those devices because device registration was state of the art. *Supra* Section IV.A. Belfiore already teaches authenticating such devices and providing information to and from such devices, and further teaches Directory Components and other functionality for maintaining device registration information for a given user. Jacobs, ¶¶75-77. For example, in the context of its event services, Belfiore confirms what any skilled artisan would expect, namely that "[w]hen a new machine enters the cloud" it would rely on the exchange and availability of addressable information for that new machine, such as a "routing configuration that instructs the forwarding service where the events should be sent" (i.e., a registration of the new machine). EX1004, [0137]; *see also id.*, [0022], [0273], [0175] ("After the user is authenticated, the directory component 150 may be used to determine where the user's information is stored."). One logical location for this client device information would be within the platform's Directory structure and, in particular, within a given user's MyState directory. Belfiore teaches that the MyState directory contains other regularly updated user information. EX1004, [0182]. POSITAs would have thus naturally used the MyState directory to also store the list of registered client devices, and to update that directory as the user logged into / logged out of different devices. Doing so would allow a particular user's information to be provided to those devices that had been authenticated with a given user's Global ID, as taught by Belfiore. It would also ensure that such information was provided *only to* those devices properly authenticated with the user's Global ID. Implementing the MyState Directory to maintain this registration information would have required minimal effort as Belfiore already teaches maintaining similar user information in that directory. Jacobs, ¶76.

That skilled artisans knew how to implement Belfiore to register each authenticated client device is confirmed by the '367 patent itself, which provides no explanation of why or how registration is performed. *In re Fox*, 471 F.2d 1405, 1407 (C.C.P.A. 1973).

Belfiore discloses its platform components as software that runs on computers to carry out certain functions. EX1004, [0041-42], [0046], [0199]. Thus, it discloses or teaches one or more computers programmed to manage context and register devices (as described in the preceding paragraphs), satisfying the claimed "registration module" and "context manager" to the extent Patent Owner argues a means-plus-function construction of these terms. *Supra* Section VII.B.

c) **Element [11.3]**

[11.3] subscribing the plurality of mobile devices registered with the context manager to one or more context events;

As explained above, obvious implementations of Belfiore would have subscribed plural mobile devices registered with the platform to context events. *Supra* Section VIII.B; *see also* Jacobs, ¶78. Belfiore teaches events that allow its system components "to have access to continually updated information about their context." EX1004, [0119]; *see also id.*, [0123]. Components both "publish[]" events (serving as "event source[s]") and "subscribe[]" to events (serving as "event sink[s]"). EX1004, [0119]; *see also id.*, FIGS. 1, 5. An "event component"—much like the '367 "event manager"— "facilitates the distributed communication of events between" the different events sources and sinks. EX1004, [0119]; *see also id.*, [0125]. "[E]vents can be distributed, registered and accessed across the server federation 120 and client devices"). *Id.*, [0124].

Belfiore also teaches maintaining consistent "state and context across a number of client devices," so that users may switch between devices mid-session. EX1004, [0097]; *see also id.*, claim 21; *supra* Section VIII.B; Jacobs, ¶79. For example, "[a]s a user works on a client device," "preference and session information" may be regularly updated in a user's MyState directory and then provided to other user devices. EX1004, [0098]. Devices are synchronized in this fashion only after being authenticated (registered) by a user. *Id.*; *id.*, [0109], claims 22-23. Belfiore further specifies using its event component to notify devices of MyState updates. *E.g.*, EX1004, [0184].

Given these teachings, a POSITA would have implemented Belfiore to subscribe each of a user's authenticated mobile client devices to MyState update events so that "state and context" were consistent across devices. *Id.*, [0097], claims 21-23 & 34-35; *see also id.*, [0133] (describing "event subscription"); *id.*, [0184]. These subscriptions would beneficially provide context updates, e.g., changes to session status, to other devices logged into by the user, thus synchronizing "state and context across a number of client devices." *Id.*, [0097]; *id.*, [0020] ("Events within the platform are used to synchronize."); *see also supra* Section VIII.B.

These subscribed-to events are "context events" at least because they relate to changes in context, such as a user's preference or session status. *See also id.*, [0119] (describing how "event system enables" "continually updated information" about "context"); *infra* Section VIII.C.1.e); Jacobs, ¶80. The '367 patent itself never uses the term "context event" outside the claims.

d) Element [11.4]

[11.4] receiving, at the context manager, a context input from one or more of the plurality of mobile devices registered with the context manager, wherein the context input includes a context change event; and

As just explained, Belfiore teaches using events to provide updates from a user's authenticated devices to the user's MyState directory and other devices. *Supra* Section VIII.C.1.c). These events are "context inputs" that include "context change events," as they include information on updates (i.e., changes), to "the user's state and context." EX1004, [0097]; *see also id.*, [0085] (describing actions taken based on a "users preferences of [sic] context"); [0119] ("The event system enables software components and architectures to have access to continually updated information about their context."); *id.*, claim 21. As the event component and MyState directory both are part of the Belfiore platform (*e.g.*, EX1004, FIG. 1), the events received by these entities are received "at the context manager," namely Belfiore's platform. *Supra* Section VIII.C.1.b); *see also supra* Section VIII.B. Belfiore thus satisfies Element 11.4. Jacobs, ¶81.

Additionally, the events of Belfiore align with the description of context in the '367 patent, which states that "the location data can be used by the invention as part of the context" and further describes "identify[ing] context such as time, location,…" EX1001, 7:48-49, 14:14-15. Belfiore similarly explains providing "events, such as an update on the general location of a user" (EX1004, [0020]) or "a stream of events about a user's location, activity, and active devices" (*id.*, [0123]). *See also id.*, [0123] ("[A] stream of events…can be analyzed by a Bayesian model to provide a set of probabilities about a user's goals."). When a user moves from an old location to a new location, any event (e.g., observation of state) reflecting the new location would be a context change event (e.g., from the old location to the new location). *Id.*, [0123]. Thus, a POSITA would recognize the platform of Belfiore as receiving context input from the client devices, the context input including context change events. Jacobs, ¶¶82-83.

e) <u>Element [11.5]</u>

> [11.5] informing the plurality of mobile devices registered with the context manager of the context change event,
>
> wherein informing the plurality of mobile devices registered with the context manager of the context change event synchronizes a context across the plurality of mobile devices.

As explained for Element 11.3, Belfiore teaches using events to provide updates from a user's authenticated devices to the user's MyState directory *and other devices*. *Supra* Section VIII.C.1.c). And as explained for Element 11.4, these events are "context inputs" that include "context change events." *Supra* Section VIII.C.1.d). Thus, these events inform the user's authenticated mobile devices (i.e., the user's registered devices) of context change events. This ensures that all devices have "consistent" user MyState information— such as the "user's state and context" and "session status information" and "preferences" (EX1004, [0097-98]), thus synchronizing a context across the plurality of mobile devices. As Belfiore states, "[e]vents within the platform are used to synchronize"). *Id.*, [0020] *see also supra* Section VIII.B; Jacobs, ¶84. Belfiore thus satisfies Element 11.5.

2. **Claim 12**

[12] The method of claim 11, wherein a context tracking module associated with the context manager informs the plurality of mobile devices registered with the context manager of the context change event.

Belfiore discloses a context tracking module of the context manager (i.e., events component 155 of Belfiore's platform) that informs registered mobile devices (e.g., client devices authenticated with the platform and subscribed to event notifications) of a context change event. As explained for Element 11.5, Belfiore's context events notifications are provided by the platform's events component 155. *Supra* Section VIII.C.1.e); *see also* EX1004, [0119-20], [0103] (triggering events based on inferring user context of performing a word processing activity). The events component 155 includes, among other things, event storage and logging mechanisms that facilitate tracking the context events (*id*., [0126]) and also utilizes directories such as the MyState directory to track and distribute context events (*supra* Section VIII.C.1.e)). Jacobs, ¶85. Thus, Belfiore at least renders obvious claim 12.

3. <u>Claim 15</u>

[15] The method of claim 11, further comprising removing, by the context manager, one or more of the plurality of mobile devices from the plurality of mobile devices registered with the context manager.

Belfiore teaches removing, by the context manager (the platform) one or more of the mobile devices (mobile client devices) from being registered (e.g., authenticated) with the platform. For example, Belfiore discloses adapted directories 905 and 910, which include information for identities of a user. EX1004, [0183-84]. These adapted directories control access of entities to the directories (and notifications of changes and updates to the directories). *Id.* Belfiore further describes terminating a relationship between an entity and the directory using the event component 155. *Id.*, [0184]. It would have been at least obvious to implement Belfiore such that the termination of the relationship resulted in the entity no longer being authenticated or registered with the platform, at least because terminating a relationship between the entity and the directory would beneficially remove access of the entity to the directory. *Id.*, [0184]; Jacobs, ¶86. Belfiore describes the term "entity" as including machines and hardware and/or software components, thus encompassing mobile devices such as the client devices 110. EX1004, [0200]; Jacobs, ¶87. Thus, a POSITA would recognize the termination of the relationship between the entity and the directory in Belfiore as removing one or more of the plurality of mobile devices from the plurality of mobile devices registered with the context manager. *Id*.

Furthermore, as described above in Section VIII.C.1.b., access to the user preference information and state information in the adaptive directories is provided after a user logs onto a client device and authenticates to a security component. EX1004, [0109], [0175]. However, as further described in Belfiore, a user may log onto another user's device as a guest user, suggesting a transient relationship between the guest user and the device in which a user would subsequently log out. *Id.*, [0251] ("She [Alex] borrows Sandra's Tablet PC and logs in as a guest user […] the storage, directory and security components described above allow Alex to authenticate herself on a new client device and have all of her personalized information available to her through the client device."), *see also Id.*, [0197] ("Many consumers also use multiple service providers with whom they have widely varying business relationships and implicit trust (subscription, transient transactions, none).").

A POSITA would thus understand or find it obvious to implement Belfiore such that devices (such as the borrowed device) would no longer be registered once the user logs off. EX1019, 283 ("Ideally, when a host leaves an area, that, too, should be announced to allow deregistration..."). Jacobs, ¶86. For example, a POSITA would have been motivated to remove the registration of the borrowed device to protect the security and privacy of the guest user by ensuring that the owner/primary user of the device is not able to access the personal information of the guest user. *Id.* Even if the user does not perform a specific log off operation, as described above in Section IV.A., it was well known for registrations to only be provided as a lease, which expires after a designated time or is only valid for a specified time interval (and thereafter removing the registration of the device). EX1011, 257 ("all devices must be registered with the DHCP server(s) in order to be granted a lease."), 263 (describing registration lease / expiration times); EX1019, 354 (describing "impolite mobile hosts that leave without saying goodbye" and need "to make registration valid only for a fixed time interval.""). Jacobs, ¶86. Thus, a POSITA would have an expectation of success in performing the removal of the device from registration since there were already known lease mechanisms that would be usable in systems such as Belfiore's platform. *Id.* Accordingly, for this additional reason, a POSITA would recognize Belfiore as disclosing or rendering obvious removing one or more of the plurality of mobile devices from the plurality of mobile devices registered with the context manager. Jacobs, ¶¶86-87.

IX. GROUND 2: CLAIMS 11, 12, 15, 17, AND 18 ARE OBVIOUS OVER KENNEWICK (EX1005) AND BELFIORE (EX1004)

As explained below, claims 11, 12, 15, 17, and 18 are obvious over Kennewick and Belfiore. Jacobs, ¶¶88-136.

A. Kennewick

Kennewick (EX1005) published March 4, 2004, from an application filed June 3, 2003, and is prior art under at least Sections 102(a)-(b). Kennewick and the ’367 patent share a common named inventor, Robert A. Kennewick, and a common title “Systems and Methods for Responding to Natural Language Speech Utterance.” *Compare* EX1001, 1 *with* EX1005, 1. Furthermore, the ’367 patent includes many of Kennewick’s disclosures nearly verbatim. *See* EX1006. As a result, overlapping disclosures and Figs. 1-6 of Kennewick and the ’367 patent describe substantially the same subject matter.

Kennewick discloses a natural language processing system (*e.g.*, EX1005, [0010]) that users interact with “on one or more devices,” such as “a mobile or portable device” (*e.g.*, *id.*, [0043]).

It further discloses synchronizing user and session context across a user’s multiple devices: “history and profile information for the user may be synchronized between the multiple devices on a periodic basis or other basis.” *Id.*, [0043].

Kennewick also discloses recognizing words and phrases using information in "dictionary and phrase tables" (*e.g.*, *id.*, [0147]) and use of a "context stack" in "scoring" contexts, where the "weight of each context for the scoring is based on the relevance of one context to another and the age of the contexts" (*e.g.*, *id.*, [0166]).

B. <u>Combining The Teachings Of Kennewick And Belfiore</u>

A POSITA implementing the teachings of Kennewick would have looked to Belfiore for its teachings on (i) the synchronization of context and other user information across a user's different devices and (ii) multi-modal user interfaces. Jacobs, ¶93.

1. <u>Device Synchronization</u>

POSITAs would have been motivated to include Belfiore's teachings on device authentication and event subscriptions, to facilitate the synchronization described in Kennewick. Jacobs, ¶94. For example, Kennewick teaches synchronizing a user's "history and profile information" between the user's many devices. *E.g.*, EX1005, [0043] ("[H]istory and profile information for the user may be synchronized between the multiple devices."). Given Kennewick's lack of express detail, POSITAs would have turned to other references teaching methods for synchronizing information on devices used in NLP systems, and naturally encountered Belfiore. A POSITA would have been motivated to incorporate Belfiore's teachings on events in order to synchronize "state, session and preference status of a user" across devices. *E.g.*, EX1004, [0019]; *see also id.*, [0020] ("Events within the platform are used to synchronize."); *supra* Section VIII.B. This would achieve Kennewick's goal of device synchronization and provide benefits taught by Belfiore, including that a user could "switch from one device to another mid-session with all state and preference information remaining consistent." *E.g.*, EX1004, [0019]; *see also id.*, [0097-98]; *supra* Section VIII.B.

Moreover, a POSITA would have been motivated to use Belfiore's device authentication to ensure that information was only synchronized between authorized devices of a given user and not synchronized indiscriminately or exposed to "unscrupulous individuals." EX1004, [0197]; *see also id.*, [0198] (identifying "authentication" as a security tool), [0200] (explaining how the "authentication module" controls access to "hardware and/or software modules and components" and prevents entities from "hiding" their identity); Jacobs, ¶95.

POSITAs would have reasonably expected success in using Belfiore's event component and related functionality to synchronize context information within a Kennewick-like system. Jacobs, ¶96. Belfiore teaches that events are "used to synchronize" (EX1004, [0020]) and that its "event system enables software components and architectures to have access to continually updated information about their context." *Id.*, [0119]. As both Kennewick and Belfiore teach distributed NLP systems, a POSITA would have recognized that incorporating Belfiore's event-based synchronization would have required only basic software modifications such that authenticated mobile devices both provided and received events indicating changes to a given user's session status and preferences. *Id.*, [0119]; *see also id.*, [0097-98] (describing use of MyState directory to facilitate providing "preference and session information" to a user's authenticated devices). Additionally, Belfiore explains that the different components of its platform may be implemented on "general purpose computing devices, including various hardware components, such as personal computers, servers, laptops, hand-held devices, cell phones or the like." *Id.*, [0041].

2. Multi-Modal Interfaces

POSITAs also would have been motivated to incorporate Belfiore's teachings on multi-modal user interfaces, to beneficially allow users to interact using not only "'traditional' methods such as keyboard entries" but also "audio/voice input" and stylus/touch input." EX1004, [0019]. Belfiore expressly teaches that these multi-modal inputs "allow[] flexibility in receiving input from a user and providing output to the user" (*id.*, [0118]) and provide different types of interactions "based on the user's state and proximity (*id.*, [0095]). *See also id.*, [0091]; Jacobs, ¶97.

POSITAs would have had a reasonable expectation of success in providing multi-modal inputs such as text and speech, at least because Kennewick teaches interfacing to its NLP system using devices with multiple input/output modalities, such as "a PDA or other portable computer device" and "a wireless telephone." EX1005, [0012], [0043]; Jacobs, ¶98. POSITAs would understand that these devices allowed for at least both voice or speech input/output, as well as text-based input/output. And Belfiore's teachings on how to implement multi-modal natural language communication were compatible with Kennewick's processing of natural language speech inputs. *E.g.*, EX1004, [0096] (discussing *inter alia* "analysis of an input string," "matching via logical form matching," and "generation of an appropriate language-like sentence along with the results."); EX1005, [0185] (describing "high level process 300 for receiving natural language speech-based queries and/or commands and generating a response according to an embodiment of the invention"). Jacobs, ¶98.

Further, a POSITA would have found it obvious to incorporate the speech user interface device of Kennewick to allow for multi-modal input as taught by Belfiore because, similarly to the mobile devices described in Kennewick as including speech interface devices for processing user speech input, the multi-modal client devices of Belfiore are described as having a user interface that allows for speech input to enable a user to interact with the client device to ask questions or provide commands using natural language. EX1004, [0091], [0104]; EX1005, [0003], [0043] ("If the invention is used with a mobile or portable device that has position location capability, the location data can be used by the invention as part of the context for user questions. A user may use the system on one or more devices."); Jacobs, ¶99. Incorporating the multi-modal functionality of Belfiore would involve the straightforward application of conventional techniques and technologies. For example, Belfiore discloses natural language speech input processing techniques such as analyzing an input string and performing logical form matching, which are described in more detail in Kennewick. EX1004, [0092], [0098], [0106]; EX1005, [0010], [0088], [0149-56].

3. **The Combined Kennewick-Belfiore System**

One obvious implementation of this Kennewick-Belfiore system (hereinafter, "Kennewick-Belfiore") would incorporate:

- Kennewick's teachings on a "speech-based information query, retrieval, presentation and command environment" (EX1005, [0009]), adapted to provide **multi-modal natural language processing** of not just speech but also text- or touch-based inputs, as taught by Belfiore (EX1004, [0091-92], FIG. 4).
- **Context-aware natural language speech processing** that "determines a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile 110, and dialog history" (EX1005, [0152]), e.g., through *inter alia*:
  - "Dialog history" that is "maintained in a context stack." *Id.*, [0166].

- Use of grammar expressions "for evaluation of a context." *Id.*, [0108].
- Use of "a scoring system" to determine a "most[] likely context or domain" *Id.*, [0153].
- Use of updated "probabilities or possibilities" to score "the possible contexts." *Id.*, [0152].

- Multiple user **mobile devices** that:
  - Are **synchronized** with one another, as taught by both Kennewick and Belfiore, upon each being **authenticated** by the same user, as taught by Belfiore. *E.g.*, EX1005, [0043-44]; Ex1004, [0019-20], [0097-98], [0168] (enumerating client devices), claim 21, claims 22-23 (authenticating), claims 34-35 (first and second mobile phones).
  - Provide **multi-modal user interfaces**, as taught by Belfiore. *E.g.*, EX1004, [0091-92], FIG. 4.
- An adaptive directory, such as a **MyState directory**, as taught by Belfiore, that stores information about a given "person or entity" (EX1004, [0182]), including "**session status information**" that "is regularly updated" as "a user works on a client device" (*id.*, [0098]). *See also id.*, [0109].
- An **event component**, through which a user's mobile devices **subscribe to event notifications** in order to "synchronize" information

and context across a user's authenticated devices, as taught by Belfiore. EX1004, [0020]; *see also id.*, [0019]; [0119]; [0184]. The event component may be incorporated with or separate from an event manager as taught by Kennewick. *E.g.*. EX1005, [0084] ("The event manager 100 may mediate interactions between other components of the main unit 98.").

As explained below, such an implementation of Kennewick-Belfiore, and obvious variants thereof, render all challenged claims obvious. Jacobs, ¶¶100-101.

C. Ground 2 Claim Mappings

1. Claim 11

a) Element [11.1] (Preamble)

If the preamble is limiting, Kennewick-Belfiore satisfies it. Jacobs, ¶102. As explained above, it would have been obvious to implement Kennewick-Belfiore to process multi-modal natural language inputs, as expressly taught by Belfiore. *Supra* Section IX.B. Kennewick discloses a "a complete speech-based natural language query and response environment." *E.g.*, EX1005, [0009-10]. Belfiore teaches a "user interface (UI)" that "is multi-modal, intelligent and responsive." EX1004, [0019]; *see also id.*, [0104-06]; *see also supra* Section VIII.C.1.a).

b) <u>Element [11.2]</u>

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on device authentication and communications, which either satisfy this element or render it obvious. *Supra* Section IX.B; *see also supra* Section VIII.C.1.b); Jacobs, ¶103. Furthermore, Kennewick teaches that its system may be deployed in multiple devices that are all connected to a network. EX1005, [0044].

c) <u>Element [11.3]</u>

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on subscriptions to event notifications relating to updates to an adaptive directory, which either satisfy this element or render it obvious. *Supra* Section IX.B; *see also supra* Section VIII.C.1.c); Jacobs, ¶104.

d) <u>Element [11.4]</u>

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on receiving updates to an adaptive directory, which either satisfy this element or would have rendered it obvious. *Supra* Section IX.B; *see also supra* Section VIII.C.1.d). When implemented within a Kennewick-like system, Belfiore's teachings result in receiving a context input from one or more of the plurality of mobile devices registered with the context manager, wherein the context input includes a context change event. *Id.*; *see also* Jacobs, ¶105.

e) **Element [11.5]**

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on event notifications that provide mobile devices with continually updated information about their context and enable a user interface to adapt and/or change based on the user's state and context across multiple devices, which either satisfy this element or render it obvious. *Supra* Section IX.B; *see also supra* Section VIII.C.1.e); Jacobs, ¶106. Furthermore, Kennewick discloses synchronizing history and profile information between multiple devices, where the history and profile information are updated automatically as a user uses the system. EX1005, [0029], [0043]. The combined Kennewick-Belfiore implementation informs the plurality of mobile devices registered with the context manager of the context change event, thereby synchronizing a context across the plurality of mobile devices.

2. **Claim 12**

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on event notifications with logging mechanisms, which either satisfy this element or render it obvious. *Supra* Section IX.B; *see also supra* Section VIII.C.2. Furthermore, Kennewick discloses maintaining a context stack, which a POSITA would understand as tracking the contexts because the context stack includes contexts that are associated with ages, meaning the system is aware of respective timing of multiple contexts. EX1005, [0166]; Jacobs, ¶107. The Kennewick-Belfiore implementation thus includes a context tracking module associated with the context manager informing the plurality of mobile devices registered with the context manager of the context change event.

3. <u>Claim 15</u>

Kennewick-Belfiore satisfies this element, as it utilizes Belfiore's teachings on device log-on, authentication, and event subscription and notification, which at least render obvious removing mobile devices that a user has logged out of. *Supra* Section IX.B; *see also supra* Section VIII.C.3; Jacobs, ¶108. So implemented, the Kennewick-Belfiore implementation's context manager (platform) removes one or more of the plurality of mobile devices from the plurality of registered mobile devices. *Id.*

4. <u>Claim 17</u>

[17.1] The method of claim 11, further comprising: receiving a natural language utterance at a multi-modal voice user interface associated with at least one of the plurality of mobile devices registered with the context manager;

Kennewick-Belfiore satisfies this element. As explained above, the combination utilizes Kennewick's teachings and disclosures on natural language speech processing, accessed via mobile devices such as mobile telephones that are registered with the platform for event-based synchronization, as taught in more detail by Belfiore. *Supra* Section IX.B. Kennewick discloses receiving "a natural language utterance" from a user at a system comprising "a speech unit interface device that receives spoken natural language queries, commands and/or other utterances from a user." EX1005, [0010]; *see also id.*, [0012] (the system comprises "a PDA or other portable computer device"). Thus, in the Kennewick-Belfiore system, natural language utterances would be received at the multi-modal voice user interfaces of the mobile devices associated with / authorized by a given user. *See also supra* Section IX.C.1.b); Jacobs, ¶109.

[17.2] identifying, using a knowledge-enhanced speech recognition engine associated with the at least one mobile device, one or more contexts that completely or partially match one or more text combinations contained in text transcribed from the natural language utterance, wherein identifying the matching contexts includes comparing the text combinations against one or more grammar expression entries in a context description grammar and one or more expected contexts stored in a context stack synchronized through the context manager;

Kennewick-Belfiore satisfies this element. As explained above, the combination utilizes Kennewick's teachings and disclosures on natural language speech processing. *Supra* Section IX.B. Kennewick's parser corresponds to the claimed "knowledge-enhanced speech recognition engine." *See also* Jacobs, ¶110.

The written description of the '367 patent describes a "knowledge-enhanced speech recognition system" (or "engine") in terms of the functionality it provides, and largely tracks the language of claim 17. EX1001, 14:6-19. The '367 patent states that the knowledge-enhanced speech recognition engine "may be used to determine the intent of the request and/or to correct false recognitions," that it "may access a set of expected contexts that are stored in a context stack to determine a most likely context" and that it "may use context specific matchers that are able to identify context such as time, location, numbers, dates, categories (e.g., music, movies, television, addresses, etc.) and other context." EX1001, 14:9-17. These descriptions are consistent with the term itself, which suggests software that uses additional/contextual information in connection with performing speech recognition/understanding—for example, to recognize the intent of a user's request (and thus determine that the speech was likely recognized correctly) or to determine that the recognized speech may be incorrect. Petitioner notes that, as shown in the passages just cited, the '367 patent uses permissive language (e.g., "may") when describing the knowledge-enhanced speech recognition engine's functionality and does not require it to be able to perform all, or even any, of the specific functions described. In any event, as explained below, Kennewick's parser provides the same functionality. *See also* Jacobs, ¶111.

To start, Kennewick's parser is used to determine the intent of a request in the transcribed natural language utterance. It does this by examining the words of the transcribed utterance, as well as by using user profile information and dialog history, to determine a "context." EX1005, [0152] ("The parser 118 determines a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile 110, and dialog history."). The context in Kennewick is described as "what is the subject the query and/or command directed to," among "other parameters used in defining the query and/or command." *Id*., [0185]. "For example," Kennewick explains, "a question with the keywords 'temperature' implies a context value of weather for the question." *Id.*, [0152]. In other words, in this example, the parser determines that the intent of a user's question with the keyword "temperature" is to ask about the weather. By determining the user's intent, the parser can identify the appropriate application and query to execute in order to try to provide the user with the requested information. *Id*. ("The context of a question or command determines the domain and thereby, the domain agent 156, if any, to be evoked."); *id*., [0092] ("Based on keywords in the questions and commands and the structures of the questions and commands, the parser invokes the required agent[s]."). And by being able to identify the context, the parser also provides an indication that the speech was recognized correctly. And these disclosures show that Kennewick's contexts are the subject matter to which a particular user input is directed and are used to determine the meaning of the user input. *Supra* Section IX.A, IX.B; Jacobs, ¶112.

In further alignment with the knowledge-enhanced speech recognition engine described in the '367 patent, Kennewick's parser is also used to correct false recognitions. For example, Kennewick discloses that the parser "uses a scoring system to determine the mostly likely context or domain for a user's question and/or command." EX1005, [0153]. But "[i]f the confidence level of the score is not high enough to ensure a reliable response, the system 90 may ask the user to verify whether the question and/or command is correctly understood." *Id*. Kennewick further explains: "If the confidence level of the domain or context score is not high enough to ensure a reliable response, the system can request that the user verify the question or command is correctly understood. In general, the question may be phrased to indicate the context of the question including all criteria or parameters. If the user confirms that the question is correct the system may proceed to produce a response. Otherwise, either the user can rephrase the original question, perhaps adding additional information to remove ambiguity, or the system may ask one or more questions to attempt to resolve the ambiguity or other actions may taken." *Id.*, [0031]. By causing the system to prompt the user to verify whether the input was correctly understood, the parser provides the ability to correct false recognitions. Jacobs, ¶113.

Next, a POSITA would have understood and found it obvious that Kennewick's parser would access a set of expected contexts that are stored in a context stack to determine a most likely context. Jacobs, ¶114. Kennewick discloses the "set of expected contexts" in a few different ways. First, Kennewick states that, as part of determining the most likely context, multiple "possible contexts are scored and the top one or few are used for further processing." EX1005, [0152]. Therefore, because these "possible contexts" are recognized as possibly matching the user's intent, and thus reflect an expectation that at least one or some will be contexts of the user's input, this collection of "possible contexts" is the first way that Kennewick discloses the "set of expected contexts." Kennewick further discloses that the possible contexts are stored in a context stack, describing an example of how "the contents of the context stack," such as keywords and criteria corresponding to past contexts, may be used in determining the context of a user's next request (e.g., "a following-on question"):

> [A voice query language] provides a grammar to clearly specify the keyword used to determine the context and a set of one or [sic] criteria or parameters…. The voice query language may be sensitive to the contents of the context stack. Thus, a following-on question can be asked using an abbreviated grammar, since key words and criteria can be inherited from the stack. For example, the user can simply asked [sic] about another keyword if the criteria of the question remain constant.

Id., [0144-45]; *see also id.*, [0166] ("Part of the dialog history is maintained in a context stack.").) In the excerpt above, the context keyword(s) stored in the context stack and associated with an earlier question is an example of an "expected context."

A POSITA would understand the context stack to be stored in a data structure, as Kennewick describes storing "[u]ser specific data, parameters, and session and history information that may determine the behavior of agents 106" in "one or more user profiles 110," which are further described as being stored in a database. *Id.*, [0085], [0186] ("These queries may then be sent to one or more agent domains, such as a domain which may access a database containing the user's profile..."). A POSITA would further recognize that the contexts in the context stack of Kennewick can be ordered because the contexts are described as having weights or "[o]ther scoring variables that can be associated through the context stack." *Id.*, [0166]; Jacobs, ¶114. The set of contexts associated with earlier question(s) is another way that Kennewick discloses and renders obvious the "set of expected contexts."

Kennewick's parser, like the '367 patent's "knowledge-enhanced speech recognition" engine, also uses "context specific matchers that are able to identify context such as time, location, numbers, dates, categories (e.g., music, movies, television, addresses, etc.) and other context." EX1001,14:13-17; *see also* Jacobs, ¶115. For example, Kennewick describes "specific parameters or values (criteria) used to determine context for questions and commands." EX1005, [0093]. These criteria, like the "context specific matchers" of the '367 patent, include time, location, dates and other information. *Id.*, [0154] ("For a weather context, examples of criteria include, location, date and time. Other criteria can include command criteria (i.e., yes/no, on/off, pause, stop), and spelling."); *see also id.*, [0189] (describing parameters and criteria).

Therefore, because Kennewick's parser includes the specific functionality ascribed to a "knowledge-enhanced speech recognition" engine in the '367 patent's written description—and more generally is software that uses additional/contextual information in connection with performing speech recognition/understanding—Kennewick's parser discloses and renders obvious the claimed "knowledge-enhanced speech recognition engine." Jacobs, ¶116.

Kennewick discloses "**identifying … one or more contexts that completely or partially match**…**text transcribed from the natural language utterance**," describing matching tokens (e.g., words) in a recognized utterance to contexts—such as the text "temperature" to the context "weather." EX1005, [0152]. Petitioner notes that, although in this example, the keyword "temperature" does not verbatim match the word "weather" for a weather context, nothing in claim 17 requires this sort of narrow, verbatim matching. Claim 17 merely recites "one or more contexts that completely or partially match one or more text combinations," (EX1001, 31:20-21) which would be satisfied by recognizing that a text combination is associated with a context, and therefore matches the context. Jacobs, ¶¶117-118.

In fact, both Kennewick, as shown in the excerpt above, and the '367 patent state how "[t]he parser 118 determines a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile 110, and dialog history." EX1005, [0152]; EX1001, 21:9-11. A POSITA would have therefore understood that the text "temperature" "matches" the "weather" context in the above example. Jacobs, ¶119. Even if claim 17 required a verbatim match, a POSITA would have understood and found it obvious, based on Kennewick's teachings, to identify a context that matches verbatim a text combination in the transcription. For example, it would have been obvious that a keyword "weather" would identify a "weather" context because the keyword completely overlaps with the context descriptor. It would not make sense, for example, for a user's question, "what's the weather?" to not match a weather context for processing the question. A POSITA would have been motivated to include verbatim matching in the system of Kennewick for at least the reason that verbatim matching represents the strongest probability of keyword matching and thereby improves the confidence and reliability of the parser by enabling a default or baseline case of context determination. A POSITA would have had a reasonable expectation of success in implementing verbatim matching in the system of Kennewick because it would require no additional hardware or software modification to detect verbatim matches using the prior or fuzzy possibility algorithms already taught by Kennewick. Jacobs, ¶119.

With respect to "text combinations," in addition to matching a single word, such as keyword "temperature" mentioned in the excerpt above, Kennewick also makes clear that "text combinations" are matched, in at least two distinct ways. Jacobs, ¶120.

First, Kennewick discloses examples of not only single-word keywords, but also multi-word keywords, such as "get the value" and "my stock portfolio" in the user utterance, "please get the value of my stock portfolio." EX1005, [0186] ("For instance, suppose a user is interested in retrieving the value of her stock portfolio. The user may utter 'please get the value of my stock portfolio.' The system 90 may review this request together with stored data such as the user's profile and determine keywords such as 'get the value' and 'my stock portfolio.'"); *see also id.*, [0187] ("The user may utter 'please record my favorite TV program.' The system 90 may review this request together with stored data such as the user's profile and determine keywords such as 'record' and 'my favorite TV program.'"), [0098] ("new keywords for a domain, which can include names of politicians, athletes, entertainers, names of new movies or songs, etc. who have achieved recent prominence."). A POSITA would have understood and found it obvious to identify context(s) by matching multi-word keywords (i.e., "text combinations") in a recognized utterance—for example, to identify a stock portfolio context (and ultimately invoke a stock portfolio agent) by matching the text combinations "get the value" and/or "my stock portfolio." *Id.*, e.g., *id.*, [0186] ("a domain [agent] which accesses stock pricing sources to determine the answers to these questions"); Jacobs, ¶121.

Second, Kennewick makes clear that the evaluation of a context can involve matching not only keyword(s) in the transcription, but criteria or parameters in the transcription as well. EX1005, [0031]; Jacobs, ¶122. For example, Kennewick discloses that "[i]n general, the question may be phrased to indicate the context of the question including all criteria or parameters." EX1005, [0031]. The criteria or parameters can be "used to determine the context for questions and commands." *Id.*, [0093] ("The system agent 150 manages the criteria handlers 152, which handle specific parameters or values (criteria) used to determine context for questions and commands.").

Kennewick further discloses identifying a context by comparing the text combinations against grammar expression entries (i.e., grammar expressions) in a context description grammar (i.e., regular grammar). EX1005, [0108], [0152]; Jacobs, ¶123. For example, Kennewick explains that the matching process described above uses a grammar. EX1005, [0108] (explaining that an agent "passes a regular grammar expression to the parser 118 for evaluation of a context or question."). Kennewick further describes how, in one example, a grammar is used "to clearly specify the keyword used to determine the context and a set of one or [sic] criteria or parameters." *Id.*, [0144] ("[T]he voice query language helps users clearly specify the keywords or contexts of the question or command along with the parameters or criteria. The language provides a grammar to clearly specify the keyword used to determine the context and a set of one or [sic] criteria or parameters.").

Kennewick also discloses that the grammar would include multiple grammar expressions. EX1005, [0108]; Jacobs, ¶124. For example, Kennewick explains how a "content package" for an agent "include[s] questions or commands" and that each question or command has an associated regular grammar expression. *Id.* ("The agent **156** passes a regular grammar expression to the parser **118** for evaluation of a context or question. An initial or default context is typically supplied for each command or question. The command or question includes a grammar for the management and evaluation of the context stack.").

Therefore, a collection of grammar expressions used to identify a context in Kennewick discloses "grammar expression entries in a context description grammar" as claimed. The collection of Kennewick discloses "a context description grammar" because it comprises a grammar that describes context(s). *Id.* This is consistent with the descriptions of a "context description grammar" in the written description of the '367 patent. EX1001, 13:58-60 ("The text combination may be compared against entries in a context description grammar that is associated with each agent 106."), 16:55-63 ("Content packages include questions or commands. Each command or question or group of commands or questions includes contexts used for creation of one or more requests. ...."). One of ordinary skill would understand the collection of grammar expressions in Kennewick to be stored as entries in a data structure, as Kennewick describes the collection as being a category of data used by agents, the parser, etc., where such data is further described as being stored in a database. EX1005, [0107-8], [0085], [0186]. One of ordinary skill would further recognize that the grammar expressions in Kennewick constitute or reference sets of rules describing the structure of natural language in a particular context, as Kennewick describes using grammar expressions "for evaluation of a context or question," where the grammar expressions describe structures (e.g., syntax, tokens) of natural language for a given context. *Id*., [0108], [0155-56]; Jacobs, ¶125.

Thus, the "regular grammar expression" in Kennewick serves the same purpose as the "context description grammar expression" in the '367 patent: it is passed to the parser for evaluation of a context or question. EX1005, [0108]; Jacobs, ¶126. With respect to evaluating a context with the context description grammar, the written description of the '367 patent largely tracks the language of claim 17, stating that "[t]he text combination [of a text message] may be compared against entries in a context description grammar that is associated with each agent 106. If a match is identified between an active grammar in the context description grammar and the command and/or request, then the match may be scored." EX1001, 13:58-62. Although Kennewick does not also include a nearly-verbatim passage to this one, Kennewick does make clear, and it would have been obvious to a person of ordinary skill in the art, that using a "regular grammar" for evaluation of a context would involve comparing grammar expression entries in the regular grammar to text combinations in a transcribed utterance. This is because, as explained above, Kennewick describes matching text in a transcription, such as keywords and criteria, to grammar expressions to identify context(s) for scoring. EX1005, e.g., [0144] ("[T]he voice query language helps users clearly specify the keywords or contexts of the question or command along with the parameters or criteria. The language provides a grammar to clearly specify the keyword used to determine the context and a set of one or [sic] criteria or parameters."), [0152] ("The parser 118 determines a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile 110, and dialog history.").

Thus, based on Kennewick's descriptions a person of ordinary skill in the art would have found it obvious, for example, that matching "temperature" to identify a "weather" context or matching "stock portfolio" to identify a "stock portfolio" context would involve matching grammar expressions in a grammar. Jacobs, ¶127.

A POSITA would have understood and found it obvious that Kennewick's parser would access one or more expected contexts that are stored in a context stack to determine a most likely context. Jacobs, ¶¶128-129. Kennewick discloses the "one or more expected contexts" in a few different ways. First, Kennewick states that, as part of determining the most likely context, multiple "possible contexts are scored and the top one or few are used for further processing." EX1005, [0152]. Therefore, because these "possible contexts" are recognized as possibly matching the user's intent, the "possible contexts" are a first way that Kennewick discloses the "one or more expected contexts." Kennewick further discloses that possible contexts matching a transcription are stored in a context stack, describing an example of how "the contents of the context stack," such as keywords and criteria corresponding to past contexts, may be used in determining the context of a user's next request (e.g., "a following-on question"):

> [A voice query language] provides a grammar to clearly specify the keyword used to determine the context and a set of one or [sic] criteria or parameters….The voice query language may be sensitive to the contents of the context stack. Thus, a following-on question can be asked using an abbreviated grammar, since key words and criteria can be inherited from the stack. For example, the user can simply asked [sic] about another keyword if the criteria of the question remain constant.

EX1005, [0144-45]; *see also id.*, [0166] ("Part of the dialog history is maintained in a context stack."). In the excerpt above, the context keyword(s) stored in the context stack and associated with an earlier question is an example of an "expected context." The one or more contexts associated with earlier question(s) is another way that Kennewick discloses and renders obvious the "one or more expected contexts." As Kennewick makes clear, the transcribed text of a user's utterance (e.g., text of "a following-on question") ("text combinations") can be compared to context keywords in the context stack ("against one or more expected contexts stored in a context stack").

A POSITA would have found it obvious to synchronize the context stack through the context manager (i.e., platform) of Belfiore, because the platform of Belfiore is described as being used to adapt and synchronize a user interface across client devices based on context. EX1004, [0098]. Accordingly, a POSITA would have been motivated to look to Belfiore to provide more detailed teachings relating to the configuration and communications involved in maintaining the synchronization of the context across mobile devices. Jacobs, ¶130.

[17.3] scoring each of the identified matching contexts; and

Kennewick-Belfiore satisfies this element. As explained above, the combination utilizes Kennewick's teachings and disclosures on natural language speech processing, including its teachings on context scoring. *Supra* Section IX.B; *see also* Jacobs, ¶¶131-132.

Kennewick discloses "scoring each of the identified matching contexts." As discussed above, Kennewick describes identifying possible contexts that match a user's utterance "by applying prior probabilities or fuzzy possibilities to keyword matching, user profile 110, and dialog history." EX1005, [0152]. "Based on these probabilities or possibilities the possible contexts are scored and the top one or few are used for further processing." *Id.*; *see also id.*, [0153] ("The parser 118 uses a scoring system to determine the mostly likely context or domain for a user's question and/or command."), [0030] ("The system may determine the mostly likely context or domain for a user's question or command, for example, by using a real-time scoring system or other technique.").

[17.4] selecting the matching context having a highest score as the most likely context for the natural language utterance.

Kennewick-Belfiore satisfies this element. As explained above, the combination utilizes Kennewick's teachings and disclosures on natural language speech processing, including on context scoring and selection. *Supra* Section IX.B; *see also* Jacobs, ¶¶133-134.

Kennewick discloses selecting (i.e., using for further processing) the matching context having a highest score (i.e., "top one or few") as the most likely context for the natural language utterance. EX1005, [0152-53]. For example, as discussed above, Kennewick discloses that "the possible contexts are scored and the top one or few are used for further processing." *Id.*, [0152]. Kennewick further describes how "[t]he parser 118 uses a scoring system to determine the mostly likely context or domain for a user's question and/or command" and that "[b]ased on this scoring, the system 90 invokes the correct agent." *Id.*, [0153]. A POSITA would have therefore understood and found it obvious that the matching context "having a highest score" would determine the "most likely context"—first, because

Kennewick discloses selecting only the context with the "top" score as one example, and second, because it would be quite sensible to select the context having a highest score as a mostly likely context even where "the top…few are used for further processing." *Id.*, [0152].

5. **Claim 18**

[18] The method of claim 17, wherein the context change event received at the context manager includes a communication from the at least one mobile device that indicates a change in context to the most likely context.

Kennewick-Belfiore satisfies this element. As explained above, the combination utilizes Belfiore's teachings and disclosures on use of event subscriptions to synchronize context across a user's devices. *Supra* Section IX.B; *see also* Jacobs, ¶¶135-136.

As explained above, Belfiore discloses receiving a context input from a mobile device, wherein the context input includes a context change event including changes to the "session status" and context of a given user on given device. *Supra* Sections IX.C.1.d) and VIII.C.1.d); *see also* EX1004, [0020], [0068], [0097], [0109], [0119-20], [0123], [0182], [0184], [0203]. For example, Belfiore teaches that its event component "enables software components and architectures to have access to continually updated information about their context." *Id.*, [0119]. Kennewick's "most likely context" (e.g., "top one or few" scored contexts) relates to the user's session status and context at least because it reflects the course and state of the user's NLP speech session. EX1005, [0151-52]. Thus, within the Kennewick-Belfiore system, the session status / context change of identifying a "most likely context" would be communicated by the event component to the platform (e.g., for storing in the user's MyState directory, as taught by Belfiore) and then also to the user's other authenticated devices, including mobile devices. *Supra* Sections VIII.C.1.c)-e); *see also* EX1004, [0097-98], [0123], [0184].

X. GROUND 3: CLAIMS 17 AND 18 ARE OBVIOUS OVER KENNEWICK, BELFIORE, AND ROSS (EX1007)

To the extent Patent Owner argues that Kennewick somehow does not teach verbatim matching and thus does not satisfy element 17.2's "identifying...one or more contexts that completely or partially match one or more text combinations contained in text transcribed from the natural language utterance," Ross provides more express teachings that confirm the obviousness of claims 17-18. Jacobs, ¶¶137-146.

A. Ross

Ross (EX1007) is a published patent application that published September 19, 2002, and qualifies as prior art at least under 35 U.S.C. § 102(b). Ross is analogous art to the '367 patent at least because it describes techniques for using a grammar to identify a context for processing a user command or question and techniques for choosing an application to invoke to process a command or question in a user's transcribed utterance. EX1001, 21:21-26; EX1007, e.g., [0010], [0013]; Jacobs, ¶138.

Ross describes a system for determining the appropriate speech-enabled application to handle a user's spoken utterance in a multi-context environment. EX1007, [0010] ("determining a speech-enabled application to receive a spoken utterance in a multi-context speech enabled environment."); *see also* Jacobs, ¶139. The system uses a "context manager" that maintains a list of application contexts, each associated with a grammar. EX1007, [0033] ("The context manager 50 includes a context list 62. The context list includes contexts 70 (e.g., 70-1, 70-1, 70-3, etc.) for speech enabled applications 26, which represent the grammars for the applications 26."). When a user speaks, the system receives a recognition message and compares the utterance against the grammars in the context list to determine which application can process the utterance. *Id.*, [0037] ("the context manager 50 receives the recognition message 68 and selects a context 70 for a speech enabled application 26 by checking the speech representation against contexts 70, starting at top of context list 62 to determine the highest priority context 70 that can process the speech representation."). Ross provides examples where words or phrases in utterances are matched verbatim against words or phrases in grammars of different contexts. *E.g., id.*, [0040-53] (describing matching an utterance of "print it" to an electronic mail application or calendar application grammar that allows phrases including "print it").

B. The Combined Kennewick-Belfiore-Ross Implementation

A POSITA implementing Kennewick-Belfiore as described above would have been motivated with a reasonable expectation of success to include Ross's teachings on comparing transcribed user utterances to grammar expressions in a grammar to identify a context. Kennewick generally teaches and discloses comparing text combinations against grammar expression entries in a context description grammar. *Supra* Section IX.C.4; *see also supra* Section IX.B.3. Ross, in turn, describes a remarkably similar technique but in much more detail. A POSITA would have thus naturally turned to these more detailed teachings in Ross—and been motivated to apply those detailed teachings—when implementing Kennewick's teachings and disclosures. Jacobs, ¶140. POSITAs would have had a reasonable expectation of success in doing so given that Ross merely provides more details on a "comparing" technique highly similar to that taught and disclosed in Kennewick. Jacobs, ¶140. A POSITA would have thus understood these Ross teachings as highly compatible with Kennewick and seen no issues in applying those teachings when implementing Kennewick. This would have involved the straightforward application of conventional techniques and technologies. Jacobs, ¶140 (citing EX1007, [0060] ("In a preferred embodiment, the grammar used for each speech-enabled application 26 may be a Backus Naur Form (BNF) grammar as shown above.").

The combination would have predictably resulted in Kennewick's parser identifying context(s) matching text combination(s) contained in a transcribed user utterance by "comparing the text combinations against the [sic] grammar expression entries in the [sic; a] context description grammar" as claimed. Jacobs, ¶141.

C. Ground 3 Claim Mappings

As explained above in Ground 2, Kennewick-Belfiore renders obvious claims 17-18. *Supra* Sections IX.B.3, IX.C.4-5; Jacobs, ¶142. To the extent Patent Owner argues that element 17.2 requires a verbatim match and that Kennewick somehow does not render obvious a method that involves such matching, Ross confirms the obviousness of the claims.

Ross describes techniques for determining a context associated with a user's spoken command or question in order to determine an application to invoke to process the command or question, just like Kennewick and the '367 patent. EX1007, [0010], [0013]. For example, Ross describes a technique where determining a context involves evaluating grammars associated with potential contexts against a user's recognized speech input—Figure 4 of Ross illustrates a general overview:

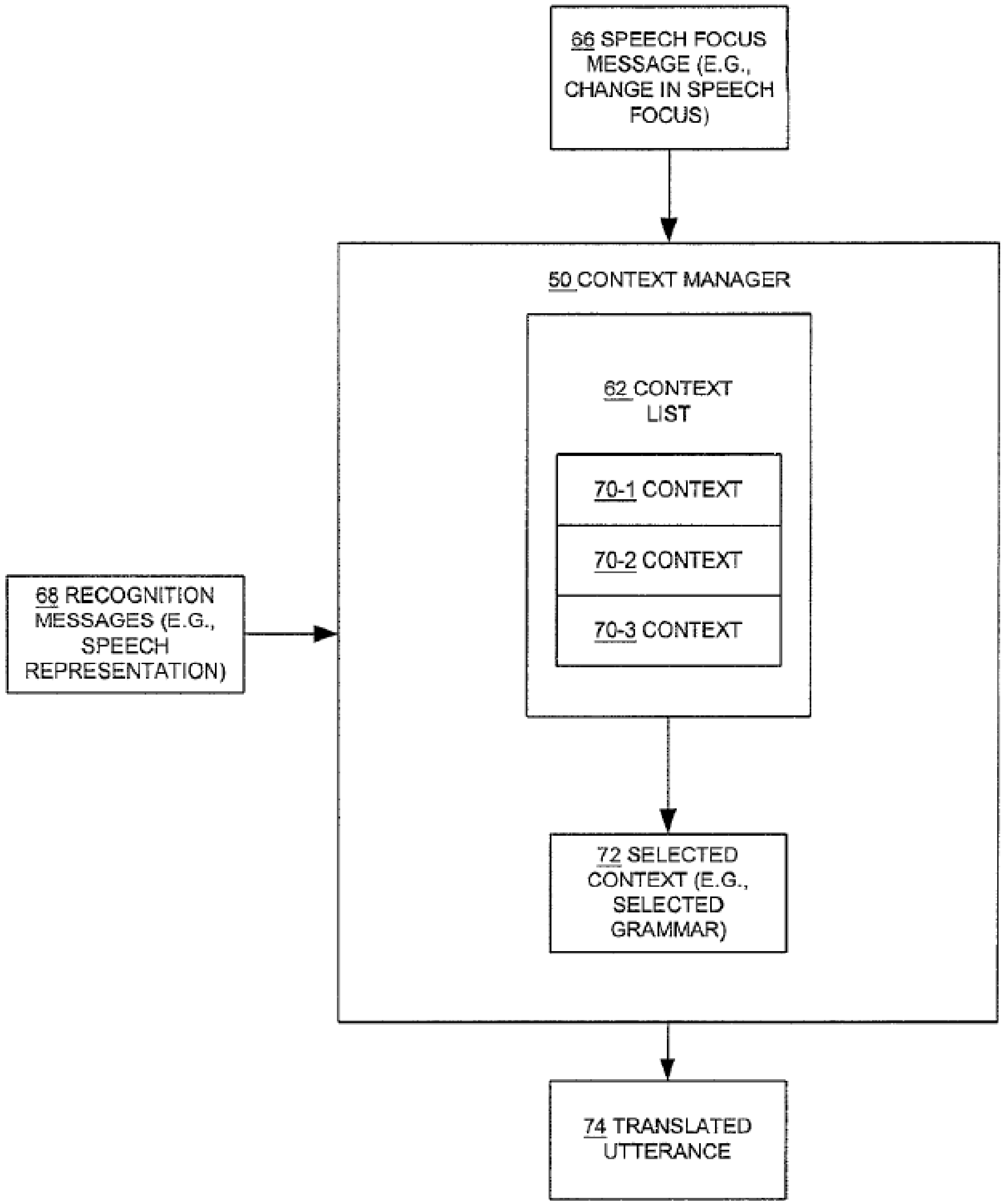

66 SPEECH FOCUS MESSAGE (E.G., CHANGE IN SPEECH FOCUS)
50 CONTEXT MANAGER
62 CONTEXT LIST
70-1 CONTEXT
70-2 CONTEXT
70-3 CONTEXT
68 RECOGNITION MESSAGES (E.G., SPEECH REPRESENTATION)
72 SELECTED CONTEXT (E.G., SELECTED GRAMMAR)
74 TRANSLATED UTTERANCE

FIG. 4

EX1007, Fig. 4. As shown above, Ross includes a context manager 50 that includes a context list 62. EX1007, [0033]. Context list 62, in turn, "includes contexts 70 (e.g., 70-1, 70-2, 70-3, etc.) for speech-enabled applications 26, which represent grammars for the applications 26." EX1007, [0033]. A user's transcribed utterance (a "recognition message") is tested against the grammars in the context list in order to determine an application to invoke:

> The context manager 50 maintains the priority and state of the various grammars in the context list 62 in the system…. Recognition messages 68 from the speech engine interface 30 are tested by the context manager 50 against the active grammars in the context list 62 in priority order. When a successful match is found, the corresponding translation 74 is dispatched to the script engine 38 for execution, and the priority of the matching grammar (i.e., the selected context 72) is raised.

EX1007, [0034]; *see also* Jacobs, ¶143.

In the following descriptions, Ross further makes clear that "test[ing]…against the active grammars" and finding a successful "match" involves matching text combinations in a user's transcribed utterance to grammar expressions in a grammar. Ross explains that "a grammar is defined for each application 26" and describes two example grammars, one for an electronic mail application:

```
<mail> = do I have any messages |
    open <message> |
    create a message |
    send <message>|
    print <message>.
<message> = the? <nth> message | it | this.
<nth> = first | second | third | fourth | fifth | . . .
```

(EX1007, [0035], [0040]) and one for a calendar application:

```
<appointment> = do I have any appointments |
    open <appointment> |
    create an appointment |
    print <appointment>.
<appointment> = the? <nth> appointment | it | this.
```

(EX1007, [0046]). A grammar expression can generally comprise word tokens, rule references, and grammar operators. Jacobs, ¶144. For example, in the grammar for the electronic mail application shown above, the line "<message> = the? <nth> message | it | this," defines a rule named "message." Jacobs, ¶144. The rule specifies a grammar expression that includes word tokens (e.g., "the," "message"), a reference to another rule ("<nth>") and a grammar operator ("?"). *Id*. As shown above, each grammar includes a number of grammar expressions.

Therefore, as Ross explains, the grammar for the electronic mail application allows spoken phrases such as "open the first message," "create a message," "send this," and "print it" to be matched, while the grammar for the calendar application matches phrases such as "open the first appointment," "create an appointment," "print the fourth appointment," and "print it." EX1007, [0041]-[0051]. The phrases that can be matched can be easily recognized by substituting the term in angle brackets with text (or expression) to which it can correspond. For example, in the grammar for the electronic mail application, "<message>" can correspond to "the? <nth> message," "it," or "this." Therefore, the grammar expression "print <message>" can match "print it" (as well as "print this" or "print the first message" and so on). Ross discloses, therefore, that a text combination in a user's transcribed utterance, such as "print it," matches the "print <message>" expression defined by the "mail" rule as well as the "it" expression defined by the "message" rule in the grammar for the electronic mail application. Jacobs, ¶145. With respect to the calendar application grammar, "print it" would also match the "print <appointment>" and "it" expressions in the "appointment" rules. *Id*.

Using the example electronic mail and calendar application grammars shown above, Ross further describes how identifying context(s) involves matching transcribed text to the grammar expressions in the grammars:

> […] the speech center system 20 is listening for any of the phrases accepted by either grammar. If the speech center 20 hears a phrase such as "print the first message" or "print the first appointment," the context manager 50 can readily figure out the intended target application 26 for the uttered sentence. Only one grammar will accept the phrase, which thus indicates the selected context 72 for that phrase[…]. If the sentence is "print it" however, both grammars are capable of accepting the utterance. [...] The context manager 50 tests the utterance against these grammars (indicated by the contexts 70 in the context list 62) in priority order, and passes the commands on to the first application 26 that has a grammar that will accept the phrase.

EX1007, [0052-53]; *see also* Jacobs, ¶146.

XI. NO OBJECTIVE INDICIA OF NON-OBVIOUSNESS

Petitioner is not aware of any evidence of objective indicia of non-obviousness having a nexus to the challenged claims. The district court complaint mentions alleged licenses; praise of technology, generally; and use of voice technology by others. EX1009, ¶¶39, 41, 43, 44. But there is no evidence of any license, much less to the challenged patent, and no evidence tying the alleged praise and use to any patent claim, much less to any claim of the challenged patent.

XII. CONCLUSION

Based on the Grounds presented herein, Petitioner respectfully requests that the Board institutes IPR of claims 11-12, 15, and 17-18 of the '367 patent.

Respectfully submitted,

Dated: July 21, 2025

By: /Andrew M. Mason/

Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner

CERTIFICATE OF COMPLIANCE WITH
TYPE-VOLUME LIMITATION PURSUANT TO 37 C.F.R. § 42.24

This brief complies with the type-volume limitation of 37 C.F.R. § 42.24(a)(1)(i).

The brief contains 13,936 words, excluding the parts of the brief exempted by 37 C.F.R. § 42.24(a).

The brief has been prepared in a proportionally spaced typeface using Microsoft Word for O365 in a 14-point Times New Roman font.

Dated: July 21, 2025

By: /Andrew M. Mason/

Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner

CERTIFICATE OF SERVICE
IN COMPLIANCE WITH 37 C.F.R. § 42.6(e)(4)

The undersigned certifies that the **PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,917,367** and **Exhibits 1001 – 1023** were served on July 21, 2025, via **Express Mail** on the Patent Owner at the following address of record as listed on the USPTO Patent Center:

Dialect, LLC
c/o The Law Offices of David A. Gerasimow, P.C.
211 W. Wacker Drive, Suite 1717
Chicago, IL 60606

By: /Andrew M. Mason/
Andrew M. Mason, Reg. No. 64,034
andrew.mason@klarquist.com
KLARQUIST SPARKMAN, LLP
One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Tel: 503-595-5300
Fax: 503-595-5301

Counsel for Petitioner