

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.

DIALECT, LLC,
Patent Owner

Case IPR2024-00752
U.S. Patent No. 8,849,652 B2
Issue Date: September 30, 2014

Title: MOBILE SYSTEMS AND METHODS OF SUPPORTING NATURAL LANGUAGE
HUMAN-MACHINE INTERACTIONS

DECLARATION OF CHRISTOPHER M. SCHMANDT

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

TABLE OF CONTENTS

	Page
I. INTRODUCTION AND QUALIFICATIONS	9
A. Qualifications and Experience	9
B. Materials Considered	13
II. PERSON OF ORDINARY SKILL IN THE ART	14
III. STATEMENT OF LEGAL PRINCIPLES	16
A. Claim Construction	16
B. Obviousness (§ 103)	18
IV. THE '652 PATENT	24
A. Overview of the Specification	24
B. The Challenged Claims	27
V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIMS	29
A. Brief Summary of Prior Art	29
1. Prior Art Common to All Grounds	29
(a) Coffman (EX1003)	29
2. Additional Prior Art for Certain Dependent Claims	34
(a) Kennewick (EX1004)	34
(b) Lee (EX1006)	39
B. Ground 1: Claims 1-9, 14-18, 20-26 Are Obvious Over Coffman	41
1. Claim 1: "A system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests, the system comprising:" (1[pre])	41

TABLE OF CONTENTS
(continued)

	Page
(a) “one or more physical processors programmed with one or more computer program instructions which, when executed, cause the one or more physical processors to:” (1[a]).....	49
(i) “generate a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;” (1[a][i]).....	50
(ii) “synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;” (1[a][ii]).....	57
(iii) “receive a natural language utterance associated with a command or request;” (1[a][iii]).....	59
(iv) “determine one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and” (1[a][iv]).....	60
(v) “determine the command or request based on the one or more words and the updated context information.” (1[a][v]).....	62
2. Claim 2: “The system of claim 1, wherein the one or more physical processors are further caused to:”.....	64
(a) “prompt a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and” (2[a])	65

TABLE OF CONTENTS
(continued)

	Page
(b) “receive a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.” (2[b])	69
3. Claim 3: “The system of claim 1, wherein the first context stack includes a plurality of context entries, and wherein the one or more physical processors are further caused to:”	71
(a) “identify, from among the plurality of context entries, one or more context entries that correspond to the one or more words, wherein the updated context information includes the one or more context entries.” (3[a])	72
4. Claim 4: “The system of claim 3, wherein identifying the one or more context entries comprises:”	73
(a) “comparing the plurality of context entries with the one or more words;” (4[a]).....	74
(b) “generating one or more rank scores for individual context entries of the plurality of context entries based on the comparison; and” (4[b]).....	75
(c) “identifying, from among the plurality of context entries, the one or more context entries based on the one or more rank scores.” (4[c]).....	80
5. Claim 5: “The system of claim 4, wherein the plurality of context entries are ordered in the first context stack, and wherein the one or more physical processors are further caused to update the ordering of the plurality of context entries in the first context stack based on the one or more rank scores.”	81

TABLE OF CONTENTS
(continued)

	Page
6. Claim 6: “The system of claim 5, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:”	81
(a) “receive a subsequent natural language utterance associated with the user; and” (6[a])	82
(b) “determine one or more subsequent words of the subsequent natural language utterance by performing speech recognition on the subsequent natural language utterance;” (6[b]).....	83
(c) “identify, from among the plurality of context entries, one or more other context entries that correspond to the one or more subsequent words based on the updated ordering; and” (6[c]).....	83
(d) “determine a subsequent command or request based on the one or more subsequent words and the one or more other context entries.” (6[d]).....	84
7. Claim 7: “The system of claim 3, wherein identifying the one or more context entries comprises identifying, from among the plurality of context entries, the one or more context entries that most closely correspond to the one or more words.”	85
8. Claim 8: “The system of claim 3, wherein the one or more physical processors are further caused to:”.....	85
(a) “identify one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and” (8[a])	85
(b) “generate a response to the command or request using the one or more domain agents.” (8[b])	89

TABLE OF CONTENTS

(continued)

Page

9.	Claim 9: “The system of claim 8, wherein each of the one or more domain agents comprises domain knowledge associated with a particular domain, the domain knowledge comprising: one or more of (i) a keyword; (ii) a link to an information source; (iii) a list of responses associated with a plurality commands or requests; (iv) a substitution list used to format the plurality of commands or requests; or (v) content including dictionaries, encyclopedias, or almanacs.”	91
10.	Independent Claim 14	92
11.	Claim 15	94
12.	Claim 16	94
13.	Claim 17	95
14.	Claim 18	96
15.	Independent Claim 20	97
16.	Claim 21	99
17.	Claim 22	100
18.	Claim 23	101
19.	Claim 24	101
20.	Claim 25	102
21.	Claim 26	103
C.	Ground 2: Claims 10-13, 19 and 27 Are Obvious Over Coffman and Kennewick.	104
1.	Claim 10: “The system of claim 8, wherein the response comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.”	104
2.	Claim 11: “The system of claim 8, wherein the one or more physical processors are further caused to:”	109

TABLE OF CONTENTS
(continued)

	Page
(a) “obtain information relating to a license agreement that is associated with at least one of the one or more domain agents;” (11[a])	109
(b) “determine, based on the information related to the license agreement, that use of the at least one of the one or more domain agents to process the command or request is permitted; and” (11[b])	113
(c) “use, based on the determination that the use is permitted, the at least one of the one or more domain agents to generate the response.” (11[c])	114
3. Claim 12: “The system of claim 1, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:” (12[pre]).....	114
(a) “obtain one or more of a cognitive model or an environmental model associated with the user, wherein the cognitive model comprises information relating to one or more interactions between the user and the system, and the environmental model comprises information indicative of how noisy an environment surrounding the user is,” (12[a]).....	115
(b) “wherein the one or more words are determined further based on one or more of the cognitive model or the environmental model.” (12[b])	121
4. Claim 13: “The system of claim 1, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:” (13[pre]).....	122
(a) “obtain a first cognitive model that comprises information relating to one or more interactions between the user and the system; and” (13[a])	122

TABLE OF CONTENTS
(continued)

	Page
(b) “obtain a second cognitive model that comprises information relating to one or more interactions between the system and a plurality of users of the system,” (13[b]).....	123
(c) “wherein the one or more words are determined further based on the first cognitive model and the second cognitive model.” (13[c])	125
5. Claim 19	126
6. Claim 27	127
D. Ground 3: Claims 13 and 19 Are Obvious Over Coffman, Kennewick and Lee.	128
VI. NO SECONDARY CONSIDERATIONS OF NONOBVIOUSNESS	132
VII. CONCLUSION	133

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

I, Christopher M. Schmandt, declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

A. Qualifications and Experience

1. I retired several years ago after a 40-year career at the Massachusetts Institute of Technology (“MIT”); for most of that time I was employed as a Principal Research Scientist at the Media Laboratory. In that role I also served as faculty for the MIT Media Arts and Sciences academic program. I was a founder of the Media Laboratory, a research lab which now spans two buildings.

2. I received my B.S. degree in Electrical Engineering and Computer Science from MIT in 1978, and my M.S. in Visual Studies (Computer Graphics) also from MIT. I was employed at MIT since 1980, initially at the Architecture Machine Group which was an early computer graphics and interactive systems research lab. In 1985, I helped found the Media Laboratory and continued to work there until retirement. I was director of a research group titled “Living Mobile.” My research spanned distributed communication and collaborative systems, with an emphasis on multi-media and user interfaces, with a strong focus on speech-based systems. I have over 70 published conference and journal papers and one book in the field of speech technology and user interaction.

3. For the first fifteen years of my career, my research emphasized speech

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

recognition and speech user interfaces. I built the first conversational computer system utilizing speech recognition and synthesis (“Put That There”) starting in 1980. Put That There parsed spoken input into a semantic structure representing a command or intent, and could ask questions of the user to evoke responses which would be further parsed and melded into the context of the current command. Some information could be gleaned from non-speech channels, specifically where a user was pointing. I continued to innovate speech user interfaces using recognition, text-to-speech synthesis, and recorded audio in a wide variety of projects. I built one of the first graphical user interfaces for audio editing, employing keyword recognition on voice memos in 1982 (“Intelligent Ear”). I built the first research-grade unified messaging system, which combined text and voice messages into a single inbox, with speech recognition over the phone for remote access, and a graphical user interface for desktop access in 1983 (“Phone Slave”). Again, Phone Slave included a parser which could accept fragmentary input and build up a complete representation of the user’s utterance with additional input. Especially in its follow up multi-user system, Conversational Desktop (1985), a number of contextual cues could also aid the dialog understanding, including date, time, calendar entries, and with whom a user was speaking over a teleconference channel. Along with my students we built the first system for real time spoken driving directions, including

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

speech-accessible maps of Cambridge, Massachusetts in 1987 (“Back Seat Driver”). Back Seat Driver had to translate a route from a map form in which each block was a separate street segment with intersection information contained in nodes, into spoken directions, using words like we use with each other to tell the driver where to go. We built some of the earliest speech-based personal assistants for managing messages, calendar, contacts, etc. (Conversational Desktop 1985, Chatter 1993, MailCall 1996). We built quite a few systems employing speech recognition in handheld mobile devices (ComMotion 1999, Nomadic Radio 2000, Impromptu 2001, and Symphony 2004, for example). We applied speech recognition to large bodies of everyday conversations captured with a wearable device and utilized as a memory aid (Memory Prosthesis 2004). We used speech recognition on radio newscasts to build a personalized version of audio newscasts (Synthetic News Radio, 1999) and also investigated adding speech recognition to a mouse-based window system a few years earlier.

4. In addition to this work running my academic research lab, I helped found and then design speech user interfaces for InTouch Systems, which built natural language based personal digital assistants accessible over the telephone. In this context I helped solve a number of problems in obtaining adequate speech recognition performance in the face of personal information, such as contact lists,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

containing words which were difficult for a system to understand. The system also maintained dialog context, such that while reading email, for example, commands related to email were anticipated, such as finding important messages or messages from some particular person. When a user was accessing a calendar, a different grammar was invoked, emphasizing creation and retrieval of calendar entries by date or time.

5. I was later awarded the prestigious Association for Computing Machinery (ACM) Computer Human Interface (CHI) Academy membership specifically for those years of work pioneering speech user interfaces.

6. In the course of my research, I built a number of speech recognition client/server distributed systems, with the first being in 1985. Much of the initial motivation for a server architecture was that speech recognition required expensive digital signal processing hardware that we could not afford to put on each computer, so a central server with the required hardware was used. Later versions of the speech recognition server architecture allowed certain computers to perform specialized tasks serving a number of client computers providing voice user interfaces, either on screens or over telephone connections. Because of my early work with distributed speech systems, I served for several years in the mid-1990s with a working group on the impact of multimedia systems on the Internet reporting to the Internet

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Engineering Task Force (IETF) and later the Internet Activities Board (IAB). This work impacted emerging standards such as Session Initiation Protocol (SIP).

7. In my faculty position I taught graduate level courses in speech technology and user interaction design, and directly supervised student research and theses at the Bachelors, Masters, and PhD level. I oversaw the Masters and PhD thesis programs for the entire Media Arts and Sciences academic program during my more senior years. I also served on the Media Laboratory intellectual property committee for many years.

8. My Curriculum Vitae is attached as **Exhibit A**.

9. I have been retained by counsel for Petitioner to provide my expert opinion in connection with the above-captioned proceeding as set forth herein.

B. Materials Considered

10. The analysis that I provide in this Declaration is based on my education, research, and experience, as well as the documents I have considered. In forming my opinions, I have read and considered U.S. Patent No. 8,849,652 B2 (“’652 patent”) [EX1001] and its prosecution history. I have cited to the following documents in my analysis below:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Exhibit No.	Description of Document
1001	U.S. Patent No. 8,849,652 B2 to Chris Weider et al. (filed May 20, 2013, issued September 30, 2014) (“ 652 ” or “ 652 patent ”)
1003	International Patent Application Publication No. WO 00/20962 to Daniel Coffman, Stephane H. Maes et al. (filed October 1, 1999, published April 13, 2000) (“ Coffman ”) ¹
1004	U.S. Patent App. Pub. No. 2004/0044516 A1 to Robert A. Kennewick et al. (filed June 3, 2003, published March 4, 2004) (“ Kennewick ”)
1005	Excerpts from Random House Webster’s Unabridged Dictionary (2001)
1006	U.S. Patent App. Pub. No. 2002/0087315 A1 to Victor Wai Leung Lee et al. (filed May 23, 2001, published July 4, 2002) (“ Lee ”)
1007	International Patent Application Publication No. WO 00/21075 to Stephane H. Maes et al. (filed October 1, 1999, published April 13, 2000) (“ Maes ”) ²
1008	Excerpts from Dafydd Gibbon et al., <i>Handbook of Multimodal and Spoken Dialogue Systems : Resources, Terminology, and Product Evaluation</i> (2000) (“ Gibbon ”)
1009	Excerpts from <i>Microsoft Computer Dictionary</i> (5th ed. 2002)
1010	Excerpts from Cezar Borkowski and Marion Manzo, <i>The Complete Idiot’s Guide to Martial Arts</i> (1999)

II. PERSON OF ORDINARY SKILL IN THE ART

11. I understand that, under the patent laws in effect before the America

¹ I understand that Coffman also issued as U.S. Patent No. 7,137,126.

² I understand that Maes also issued as U.S. Patent No. 7,003,463.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Invents Act (“AIA”) of 2011, an assessment of claims of a patent filed before the AIA took effect should be undertaken from the perspective of a person of ordinary skill in the art as of the earliest claimed priority date (i.e., the “time the invention was made”). I have preliminarily for purposes of my Declaration only assumed that date to be August 29, 2005, the filing date of the earliest patent application to which the ’652 patent claims priority (appl. ser. no. 11/212,693).

12. I have also been advised that to determine the appropriate level of a person having ordinary skill in the art, the following factors may be considered: (1) the types of problems encountered by those working in the field and prior art solutions thereto; (2) the sophistication of the technology in question, and the rapidity with which innovations occur in the field; (3) the educational level of active workers in the field; and (4) the educational level of the inventor.

13. The ’652 patent states that it “provides mobile devices having a speech interface and/or a combination of a speech interface and a non-speech interface to enable natural language human-machine interactions” and “enables mobile users to submit natural language speech and/or non-speech questions or commands in a wide range of domains..” (’652, 1:22-28.) In my opinion, a person of ordinary skill in the art would have possessed a bachelor’s degree in electrical engineering or computer science, and two years of work experience in the field of speech- and

natural language-interfaces for interacting with and directing computer systems and software, including speech recognition and processing. A person could also have qualified as a person of ordinary skill in the art with some combination of (1) more formal education (such as a master's of science degree) and less technical experience, or (2) less formal education and more technical or professional experience.

14. My opinions regarding the level of ordinary skill in the art are based on, among other things, my over 40 years of experience in the field of computer science, my understanding of the basic qualifications that would be relevant to an engineer or scientist tasked with investigating methods and systems in the relevant area, and my familiarity with the backgrounds of colleagues, co-workers, and employees, both past and present.

15. Although my qualifications and experience exceed those of the hypothetical person having ordinary skill in the art defined above, my analysis and opinions regarding the '652 patent have been based on the perspective of a person of ordinary skill in the art as of August 2005.

III. STATEMENT OF LEGAL PRINCIPLES

A. Claim Construction

16. I understand that a purpose of claim construction is to determine what

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

a person of ordinary skill in the art would have understood the claim terms to mean. Claim terms are generally given their ordinary and customary meaning, which is the meaning that the term would have to a person of ordinary skill in the art in question as of the effective filing date.

17. I understand that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification. I understand that the patent specification, under the legal principles, has been described as the single best guide to the meaning of a claim term, and is thus highly relevant to the interpretation of claim terms. And I understand for terms that do not have a customary meaning within the art, the specification usually supplies the best context of understanding the meaning of those terms.

18. I further understand that other claims of the patent in question, both asserted and unasserted, can be valuable sources of information as to the meaning of a claim term. Because the claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

19. I understand that the prosecution history can further inform the meaning

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

of the claim language by demonstrating how the inventors understood the invention and whether the inventors limited the invention in the course of prosecution, making the claim scope narrower than it otherwise would be. Extrinsic evidence, such as dictionaries, may also be consulted in construing the claim terms.

20. I understand that, in *Inter Partes* Review (IPR) proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. district court (which I understand is called the “*Phillips*” claim construction standard), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

21. I have been instructed by counsel to apply the “*Phillips*” claim construction standard for purposes of interpreting the claims in this proceeding, to the extent they require an explicit construction. The description of the legal principles set forth above thus provides my understanding of the “*Phillips*” standard as provided to me by counsel.

B. Obviousness (§ 103)

22. I understand that a patent claim is obvious if, as of the effective filing date, it would have been obvious to a person having ordinary skill in the field of the

technology (the “art”) to which the claimed subject matter belongs.

23. I understand that the following factors should be considered in analyzing obviousness: (1) the scope and content of the prior art; (2) the differences between the prior art and the claims; and (3) the level of ordinary skill in the pertinent art. I also understand that certain other facts known as “secondary considerations” such as commercial success, unexplained results, long felt but unsolved need, industry acclaim, simultaneous invention, copying by others, skepticism by experts in the field, and failure of others may be utilized as indicia of nonobviousness. I understand, however, that secondary considerations should be connected, or have a “nexus,” with the invention claimed in the patent at issue.

24. I understand that a reference qualifies as prior art for obviousness purposes when it is analogous to the claimed invention. The test for determining what art is analogous is: (1) whether the art is from the same field of endeavor, regardless of the problem addressed, and (2) if the reference is not within the field of the inventor’s endeavor, whether the reference still is reasonably pertinent to the particular problem with which the inventor is involved.

25. I understand that a person of ordinary skill in the art is assumed to have knowledge of all prior art. I understand that one skilled in the art can combine various prior art references based on the teachings of those prior art references, the

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

general knowledge present in the art, or common sense. I understand that a motivation to combine references may be implicit in the prior art, and there is no requirement that there be an actual or explicit teaching to combine two references. Thus, one may take into account the inferences and creative steps that a person of ordinary skill in the art would employ to combine the known elements in the prior art in the manner claimed by the patent at issue. I understand that one should avoid “hindsight bias” and ex post reasoning in performing an obviousness analysis. But this does not mean that a person of ordinary skill in the art for purposes of the obviousness inquiry does not have recourse to common sense.

26. I understand that when determining whether a patent claim is obvious in light of the prior art, neither the particular motivation for the patent nor the stated purpose of the patentee is controlling. The primary inquiry has to do with the objective reach of the claims, and that if those claims extend to something that is obvious, then the entire patent claim is invalid.

27. I understand one way that a patent can be found obvious is if there existed at the time of the invention a known problem for which there was an obvious solution encompassed by the patent’s claims. I understand that a motivation to combine various prior art references to solve a particular problem may come from a variety of sources, including market demand or scientific literature. I understand

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

that a need or problem known in the field at the time of the invention can also provide a reason to combine prior art references and render a patent claim invalid for obviousness. I understand that familiar items may have obvious uses beyond their primary purpose, and that a person of ordinary skill in the art will be able to fit the teachings of multiple prior art references together like the pieces of a puzzle. I understand that a person of ordinary skill is also a person of at least ordinary creativity. I understand when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this finite number of predictable solutions leads to the anticipated success, I understand that the invention is likely the product of ordinary skill and common sense, and not of any sort of innovation. I understand that the fact that a combination was obvious to try might also show that it was obvious, and hence invalid, under the patent laws. I understand that if a patent claims a combination of familiar elements according to known methods, the combination is likely to be obvious when it does not more than yield predictable results. Thus, if a person of ordinary skill in the art can implement a predictable variation, an invention is likely obvious. I understand that combining embodiments disclosed near each other in a prior art reference would not ordinarily require a leap of inventiveness.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

28. I understand that obviousness may be shown by demonstrating that it would have been obvious to modify what is taught in a single piece of prior art to create the patented invention. Obviousness may also be shown by demonstrating that it would have been obvious to combine the teachings of more than one item of prior art. I understand that a claimed invention may be obvious if some teaching, suggestion, or motivation exists that would have led a person of ordinary skill in the art to combine the invalidating references. I also understand that this suggestion or motivation may come from the knowledge of a person having ordinary skill in the art, or from sources such as explicit statements in the prior art. I understand that when there is a design need or market pressure, and there are a finite number of predictable solutions, a person of ordinary skill may be motivated to apply common sense and his skill to combine the known options in order to solve the problem.

29. I understand the following are examples of approaches and rationales that may be considered in determining whether a piece of prior art could have been combined with other prior art or with other information within the knowledge of a person having ordinary skill in the art:

- (1) Some teaching, motivation, or suggestion in the prior art that would have led a person of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention;

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

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

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

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

(5) Applying a technique or approach that would have been “obvious to try” (choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success);

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

(7) Use of a known technique to improve similar products, devices, or methods in the same way.

30. I understand that, when determining whether a claimed combination is obvious, the correct analysis is not whether one of ordinary skill in the art, writing on a blank slate, would have chosen the particular combination of elements described in the claim. Instead, I understand the correct analysis considers whether

one of ordinary skill, facing the wide range of needs created by developments in the field of endeavor, would have seen a benefit to selecting the combination claimed.

31. I understand that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference. The test for obviousness, in other words, is not whether the references could be physically combined but whether the claimed inventions are rendered obvious by the teachings of the prior art as a whole.

IV. THE '652 PATENT

A. Overview of the Specification

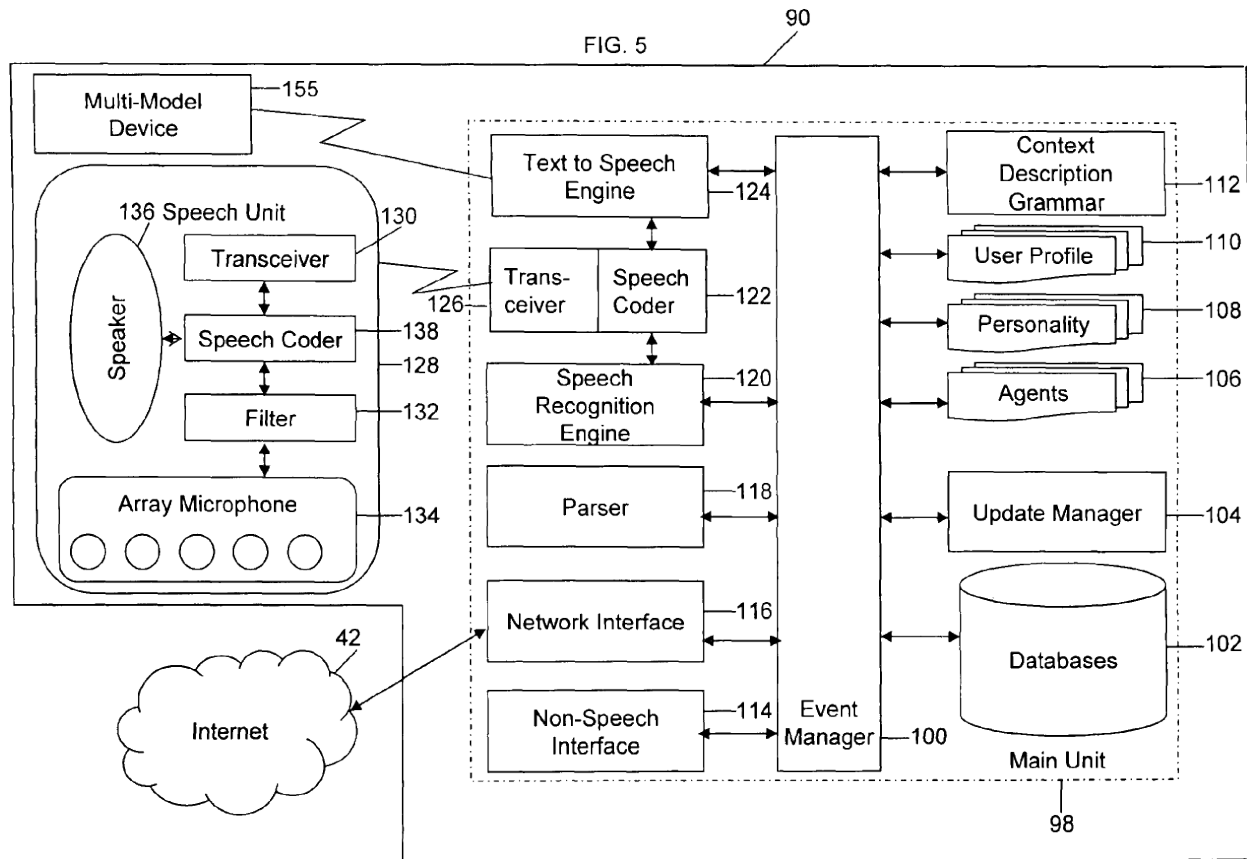
32. The '652 patent states that it “provides mobile devices having a speech interface and/or a combination of a speech interface and a non-speech interface to enable natural language human-machine interactions” and “enables mobile users to submit natural language speech and/or non-speech questions or commands in a wide range of domains.” ('652, 1:22-28.)

33. The '652 patent acknowledges that “[s]peech recognition has steadily improved in accuracy and today is successfully used in a wide range of applications” and that “[n]atural language processing has previously been applied to the parsing of speech queries.” ('652, 1:67-2:4.) With respect to “telematic systems”—i.e., “systems that bring human-computer interfaces to mobile environments”—the '652

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

patent acknowledges that “speech interfaces are being adopted in many telematic applications.” (’652, 1:33-34, 1:39-41.) The ’652 patent purports to overcome “drawbacks of prior telematic systems” (’652, 2:58-59), although the patent’s claims, by their plain language, are not limited to telematic systems or mobile environments (*see also* ’652, 15:61-63 (“The system, or portions thereof, may be used...at a fixed location such as an office or home, or at other locations.”)).

34. “FIG. 5 is an overall diagrammatic view of the interactive natural language processing system according to an embodiment of the invention” (’652, 15:38-40):



(’652, Fig. 5.) “The system 90 may include a main unit 98, speech units 128, and multi-modal devices 155” (such as “mobile devices, stand alone or networked computers, personal digital assistances (PDAs), portable computer devices, or other multi-modal devices”). (’652, 23:38-39, 25:60-65.) “Agents 106 may include a collection of grammars, criteria handlers, and algorithms that are accessible to respond to a set of requests and/or commands.” (’652, 25:32-34.)

35. Speech unit 128 includes a microphone to receive a spoken utterance from a user. (’652, 25:50-52.) After initial processing, the speech is passed to

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

speech recognition engine **120** for processing. ('652, 26:18-20.) After speech recognition, parser **118** “may determine a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile **110**, dialog history, and context stack contents.” ('652, 34:2-5.) “For example, a question with the keywords ‘temperature’ implies a context value of weather for the question.” ('652, 34:7-8.) “Once the context for the question or command has been determined, the parser **118** can invoke the correct agent **156, 150**.” ('652, 34:34-36.) A context stack “includes an ordered list of command contexts, to enable follow-up requests.” ('652, 20:10-12; *see also id.*, 36:13-14 (“Part of the dialog history is maintained in a context stack.”).) “The other registered mobile devices may be informed of context changes through a context tracking module to enable synchronizing of context across the registered modules.” ('652, 4:29-32.)

36. I discuss additional aspects of the '652 patent as needed in the analysis below.

B. The Challenged Claims

37. This Declaration addresses claims 1-27 of the '652 patent. Independent claim 1 recites:

1. A system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests, the system

comprising:

- [a] one or more physical processors programmed with one or more computer program instructions which, when executed, cause the one or more physical processors to:
 - [a][i] generate a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;
 - [a][ii] synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;
 - [a][iii] receive a natural language utterance associated with a command or request;
 - [a][iv] determine one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and
 - [a][v] determine the command or request based on the one or more words and the updated context information.

(’652, 38:39-61 (Claim 1) (bracketed notation (e.g., “[a]”) added).)

38. I address the claims further in my detailed analysis in **Part V** below.

V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIMS

39. I have reviewed and analyzed the prior art references and materials listed in **Part I.B** above. In my opinion the claims of the '652 patent would have been obvious to a person of ordinary skill in the art based on the following combinations of the prior art.

Ground	Claims	Basis for Challenge under § 103
1	1-9, 14-18, 20-26	Coffman (EX1003)
2	10-13, 19, 27	Coffman (EX1003) + Kennewick (EX1004)
3	13, 19	Coffman (EX1003), Kennewick (EX1004) and Lee (EX1006)

40. I am informed by counsel that each of the references cited in the grounds above qualifies as prior art to the challenged claims because each reference was filed and/or published before the '652 patent priority date (August 29, 2005).

A. Brief Summary of Prior Art

1. Prior Art Common to All Grounds

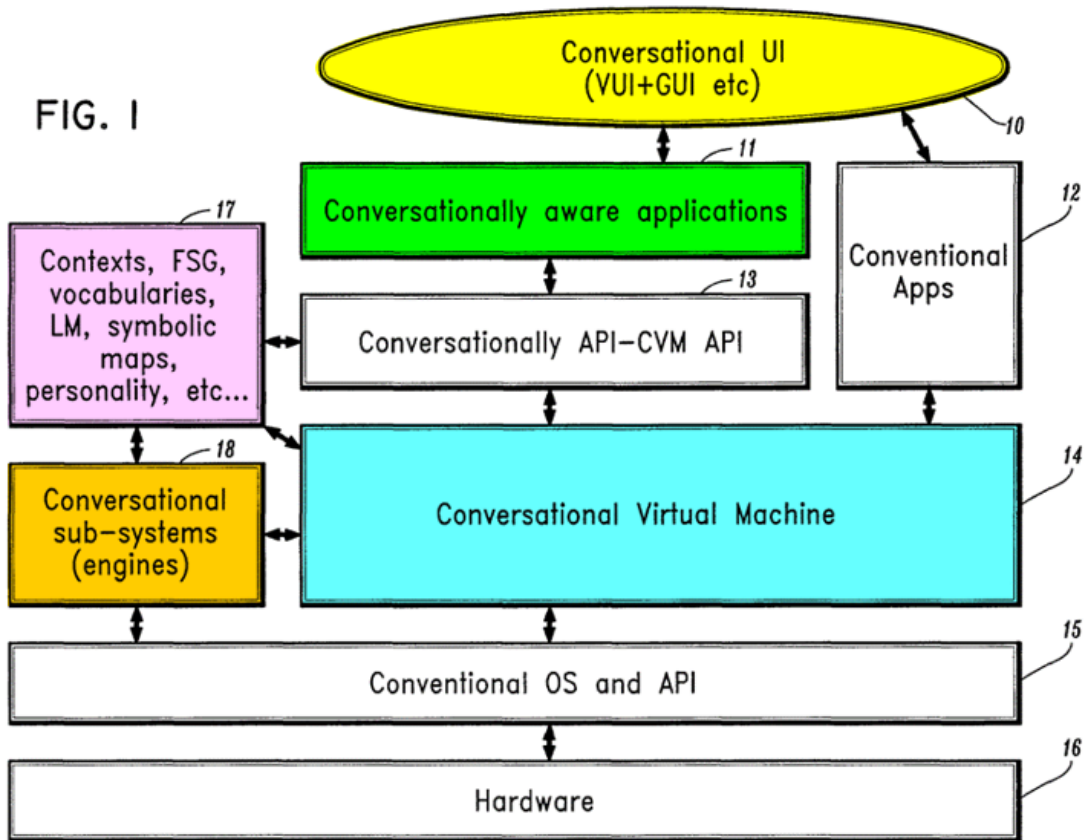
(a) Coffman (**EX1003**)

41. **Coffman**, International Patent Application Publication No. WO 00/20962 to Daniel Coffman et al. is entitled “Conversational Computing via Conversational Virtual Machine.” The face page of Coffman identifies International Business Machines as the applicant. (Coffman, face page (field 71).) I am informed

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

that Coffman qualifies as prior art because it published on April 13, 2000, before the earliest priority date of the '209 patent.

42. Coffman is a strong reference that discloses or suggests the majority of the limitations of the challenged claims. Coffman describes a “conversational computing system.” Generally speaking, a conversational system is a computing system with either purely voice input/output or a multi-modal system with voice input/output (as well as other input/output means). (Maes [EX1007], 1:12-13.) Coffman’s system “provides the capability to have a natural dialog with NLU [natural language understanding], NLG [natural language generation] and mixed initiative across multiple applications, with multiple domains.” (Coffman, 36:29-31.) Figure 1 of Coffman provides a high-level depiction of the system’s components:



(Coffman, Fig. 1 (highlighting added).) As shown above, the system includes, among other things, “a universal coordinated multi-modal conversational user interface (CUI) (10),” “a plurality of conversationally-aware applications (11),” “conversational kernel (14),” “conversational arguments (17),” and “conversational engines (18).” (Coffman, face page (Abstract).)

43. The conversational virtual machine (or CVM) is “[a] key component” of the conversational system that “allows a computer or any other interactive device to converse with a user.” (Coffman, 10:23-25, 11:10-13.) To do this, the CVM uses conversational subsystems “including speech recognition, speaker recognition, text-

to-speech, natural language understanding and natural dialog generation engines to understand and generate dialog between user and machine.” (Coffman, 5:1-4.) The **conversational subsystems** “are responsible for converting voice requests into queries and converting outputs and results into spoken messages using the appropriate **data files 17** (e.g., contexts, finite state grammars, vocabularies, language models, symbolic query maps etc.)” (Coffman, 13:8-11.)

44. “The **conversationally aware applications 11** are applications that are specifically programmed for operating with a **CVM core layer (or kernel) 14** via conversational application APIs 13.” (Coffman, 12:28-30.) A **conversationally aware application** operates with a **CVM** by providing its **data files** to the CVM for use by the **conversational subsystems**—an application hooks into the CVM API by “registering all its conversational capabilities and requirements” including “[a]ctive vocabulary, finite state grammar and language models to control the application” and “[s]ymbolic mapping if NLU is supported or at list [sic; least] a context state list.” (Coffman, 57:23-27; *see also id.*, 36:31-32 (“each application will provide the CVM 401 its own parsing and translation arguments”).)

45. The CVM also “allows the user to run multiple tasks on a machine.” (Coffman, 11:10-13.) The CVM is able to do this by using a “context stack,” which enables the system to determine the appropriate conversationally-aware application

for handling user input. (Coffman, 42:16-17 (“This stack structure allows to maintain non-ambiguous conversational interactions with multiple tasks, threads or processes.”).) The context stack “comprises all the information associated with an application.” (Coffman, 21:23-24.) “Such information includes all the variable, states, input, output and queries to the backend that are performed in the context of the dialog and any extraneous event that occurs during the dialog.” (Coffman, 21:24-26.) Coffman explains:

In general, the context stack 405 accumulates the context (i.e., full query arguments list of attribute value n-uples, and state/mode) of each active process with an activated discourse (i.e., conversational interaction associated with a given task/process/thread) along with any data files 413 (or at least identifiers of such conversational arguments) for the different engines that may be needed for input understanding (e.g., files or arguments that the engines use for performing their respective tasks such as active FSG, topic, vocabulary, HMM (hidden markov models), voiceprints, language models or possible queries for a speech input). In other words, the term “context” refers to the state of each discourse (whether active or nonnative), which keeps track of the past history of the discourse, its current state, and the specific characteristics and full query arguments of the corresponding task (e.g, vocabulary file, language model, parsing, tags, voiceprint, TTS rules, grammar, NLU etc. of each active task/process) along with any data needed for input understanding.

(Coffman, 29:20-32.) To identify the appropriate conversationally-aware application to handle a user's input (such as a user's spoken utterance), the system searches the context stack for a context that is an acceptable match:

Completion/search for the active context is performed from context to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained and optionally confirmation is obtained from the user. As soon a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active.

(Coffman, 42:1-5.) "The first task to have its dialog completed by producing a non-ambiguous query is executed and the corresponding query as interpreted by the other application is stored to activate if the recognized query is rejected by the user."

(Coffman, 37:2-4; *see also id.*, 42:18-19 ("If the request is complete, it will be executed, pending possible request for confirmation by the user, e.g. when it is irreversible.").)

46. I discuss additional aspects of Coffman as needed in my analysis of the challenged claims below.

2. Additional Prior Art for Certain Dependent Claims

(a) Kennewick (EX1004)

47. **Kennewick**, U.S. Patent Application Publication No. 2004/0055516

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

A1 to Robert A. Kennewick et al. is entitled “Systems and Methods for Responding to Natural Language Speech Utterance.” I am informed that Kennewick qualifies as prior art because it published on March 4, 2004, which is more than one year before the earliest priority date of the ’652 patent. Kennewick ultimately issued as U.S. Patent No. 7,398,209 B2.

48. The ’652 patent shares a number of named inventors with Kennewick, and appears to incorporate a substantial amount of Kennewick’s disclosures, including describing an interactive natural language processing system that is substantially identical to the ’652 system discussed in section IV.A above. Figure 1 of Kennewick, reproduced below, is nearly identical to Figure 5 of the ’652 patent. It depicts “an overall diagrammatic view according to one embodiment of the invention” (Kennewick, ¶0073):

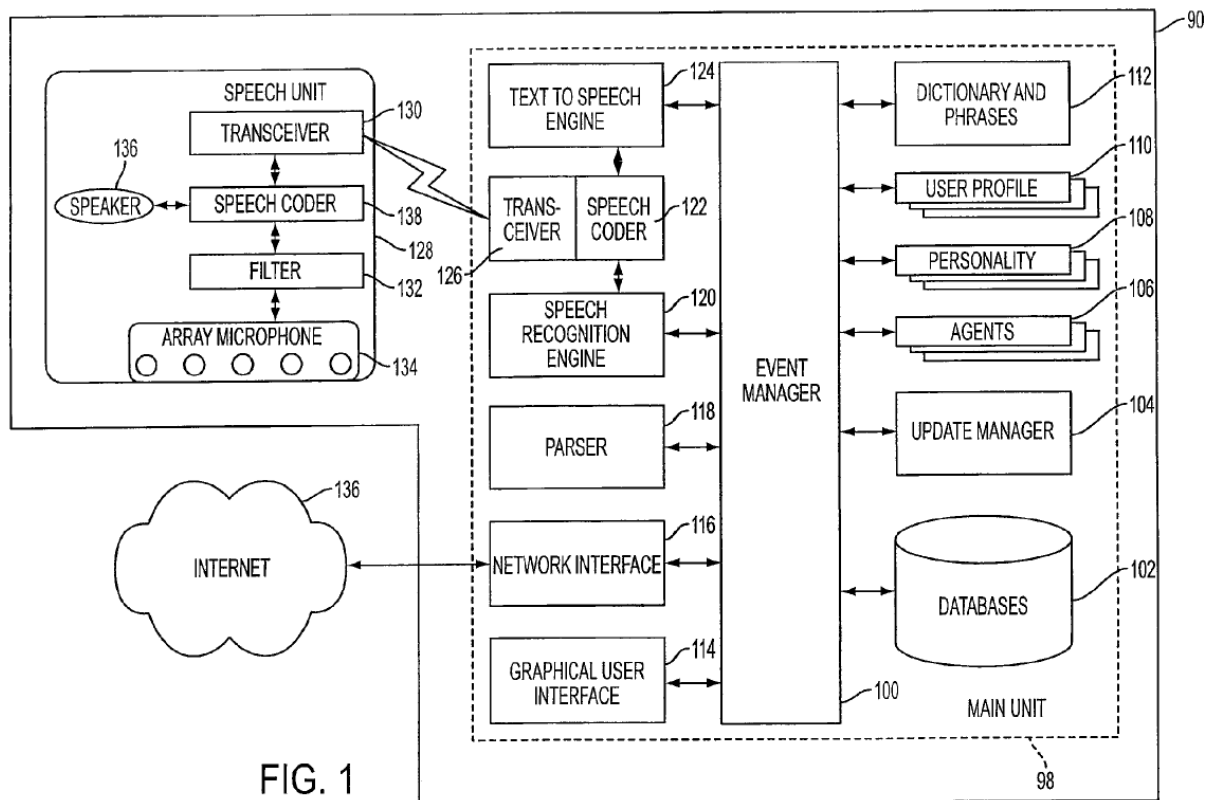


FIG. 1

(Kennewick, Fig. 1.) As shown in the figure above, system 90 includes, among other things, speech unit 128, speech recognition engine 120, dictionary and phrases 112, parser 118, and agents 106. Speech unit 128 includes a microphone to receive a spoken utterance from a user. (Kennewick, ¶0086.) After initial processing, the speech is passed to speech recognition engine 120 for processing “using data in the dictionary and phrases module 112 and received from the agents 106.” (Kennewick, ¶0088.) Then, “[t]he recognized words and phrases are processed by the parser 118, which transforms them into complete commands and questions using data supplied by an agent 106.” (Kennewick, ¶0088.) “The agent 106 then process the commands

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

or questions.” (Kennewick, ¶0088.) In order to select the correct agent to process the command or question, “parser **118** determines a context for an utterance by applying prior probabilities or fuzzy possibilities to keyword matching, user profile **110**, and dialog history.” (Kennewick, ¶0152.) “For example, a question with the keywords ‘temperature’ implies a context value of weather for the question.” (Kennewick, ¶0152.) “Once the context for the question or command has been determined, the parser **118** can invoke the correct agent **156, 150**.” (Kennewick, ¶0154.) The agent returns results to the user. (Kennewick, ¶0089.)

49. My analysis points to Kennewick in connection with limitations related to a license agreement recited in claims 11 and 27. The disclosures in the ’652 patent related to these limitations also are included in Kennewick. In particular, Kennewick describes an “agent manager” that verifies that software use is within allowed license terms and conditions, and if so, allows the software to be used. (Kennewick, ¶0104.) As shown below, Kennewick’s disclosures are nearly identical to the ’652 patent (underlining added to indicate differences):

’652 patent, 28:63-29:9	Kennewick, ¶0104
The agent manager 154 may provide license management clients <u>that are</u> capable of executing most any license terms and conditions. When a particular	The agent manager 154 may provide a license management client capable of executing most any license terms and conditions. When a particular

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

'652 patent, 28:63-29:9	Kennewick, ¶0104
agent 106 and/or database element 102 is <u>selected based on a submitted</u> command, the agent manager 154 verifies that the use of the agent or data element is within the allowed terms and conditions, and if so, invokes the agent or allows access to the data element. License management schemes that can be implemented through the agent manager 154 include outright purchase, subscription for updates, one time or limited time use. Use of shared agents and data elements (such as those downloaded from web sites maintained by groups of domain experts) may also be managed by the agent manager 154.	agent 106 and/or database element 102 is <u>required by a</u> command, the agent manger [sic] 154 verifies that the use of the agent or data element is within the allowed terms and conditions, and if so, invokes the agent or allows access to the data element. License management schemes that can be implemented through the agent manager 154 include outright purchase, subscription for updates, one time or limited time use. Use of shared agents and data elements (such as those downloaded from web sites maintained by groups of domain experts) may also be managed by the agent manager 154.

50. My analysis also cites Kennewick in connection with “**cognitive model(s)**” recited in claims 12-13 and 19, including a “**cognitive model [that] comprises information relating to one or more interactions between the user and the system**” and that is used to determine one or more words of a natural language utterance. As explained in detail in the analysis below, Kennewick similarly describes collecting user profile information—such as a user’s dialog

history—and using that information to improve the system’s speech recognition accuracy. (Kennewick, ¶¶0029, 0114, 0149, 0151.)

51. I discuss additional aspects of Kennewick as needed in my analysis of the challenged claims below.

(b) Lee (EX1006)

52. Lee, U.S. Patent Application Publication No. 2002/0087315 A1 to Victor Wai Leung Lee et al. is entitled “Computer-Implemented Multi-Scanning Language Method and System.” I am informed that Lee qualifies as prior art because it published on July 4, 2002, before the earliest priority date of the ’652 patent.

53. My analysis points to Lee in connection with claims 13 and 19, which recite using a “**second cognitive model [that] comprises information relating to one or more interactions between the system and a plurality of users of the system**” and that is used to determine one or more words of a natural language utterance. As I explain below, Kennewick discloses the claimed “**second cognitive model**” in the form of profile information about other users, which includes users’ prior interactions with the system, such as “history of questions asked” and “session histories.” (Kennewick, ¶¶0114, 0029, 0149, 0151.) Kennewick explains that using other users’ profile information when performing speech recognition could help the system recognize likely input from the user. (Kennewick, ¶0149.)

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

54. **Lee** provides further confirmation that it would have been obvious to use other users' profile information involving users' interactions with the device, such as prior questions and commands, in connection with performing speech recognition. Lee describes techniques for assisting speech recognition accuracy using information collected about multiple users. (Lee, Abstract, ¶¶0002, 0016.) Lee describes a "multi-scan control unit **32**" that may use "recognition assisting databases **48**" to supplement recognition of speech input, including a database that "maintains a personalized profile that have already been recognized for a particular user or for users who have previously submitted requests which are similar to the request at hand." (Lee, ¶0016.) Lee also describes a "popularity engine **144**" that "contains data compiled from multiple users' histories that has been calculated for the prediction of likely user requests." (Lee, ¶0024.) Lee explains that "[t]his increases the accuracy of word recognition"; for example, with this information, the system can "scan[] for the most probable match for words in the user utterance based on past requests." (Lee, ¶¶0024, 0026.)

55. As I explain for Ground 3 below, it would have been obvious to incorporate Lee's teachings into the Coffman's conversational system as taught by the combination of Coffman and Kennewick, and it would have predictably resulted in determining one or more words of a natural language utterance using a "**second**

cognitive model” as claimed.

B. Ground 1: Claims 1-9, 14-18, 20-26 Are Obvious Over Coffman.

1. Claim 1: “A system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests, the system comprising:” (1[pre])

56. For reference, claim 1 recites:

1. A system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests, the system comprising:

[a] one or more physical processors programmed with one or more computer program instructions which, when executed, cause the one or more physical processors to:

[a][i] generate a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;

[a][ii] synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;

[a][iii] receive a natural language utterance associated with a command or request;

[a][iv] determine one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and

[a][v] determine the command or request based on the one or more words and the updated context information.

(’652, 38:39-61 (Claim 1) (bracketed notation (e.g., “[a]”) added).)

57. The preamble recites “[a] **system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests, the system comprising.**” Assuming the preamble of claim 1 imposes a claim limitation, it is disclosed and rendered obvious by the prior art.

58. As explained in more detail for the claim limitations below, Coffman discloses a conversational computing “**system.**” (Coffman, 36:29-31, 5:1-4, 13:8-11.) The system is “**for processing...utterances**”—Coffman explains that the system processes spoken input from a person. (Coffman, e.g., 53:8-10 (“Voice input from a microphone (e.g. the standard voice input) can be arguments of function calls with the sequence of words, phones, or user identity or queries (symbolic language representation provided by NLU).”), 37:14-22 (“Spoken input from a speech client (e.g., telephone 600) is subject to a speech recognition process 604 and other input (e.g., keyboard, mouse clicks etc) are subject to NLU processing 605.”), 41:2-5

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

(“Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with speech inputs being transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active query and complete it.”), 53:22-26 (“The input process described above in accordance with one aspect of the present invention considers that speech or any other input stream is included as a classical input/output stream that is susceptible to all forms of processing typically reserved for character or binary input.”).)³ Figure 6 of Coffman illustrates:

³ All underlining added to quoted material unless noted.

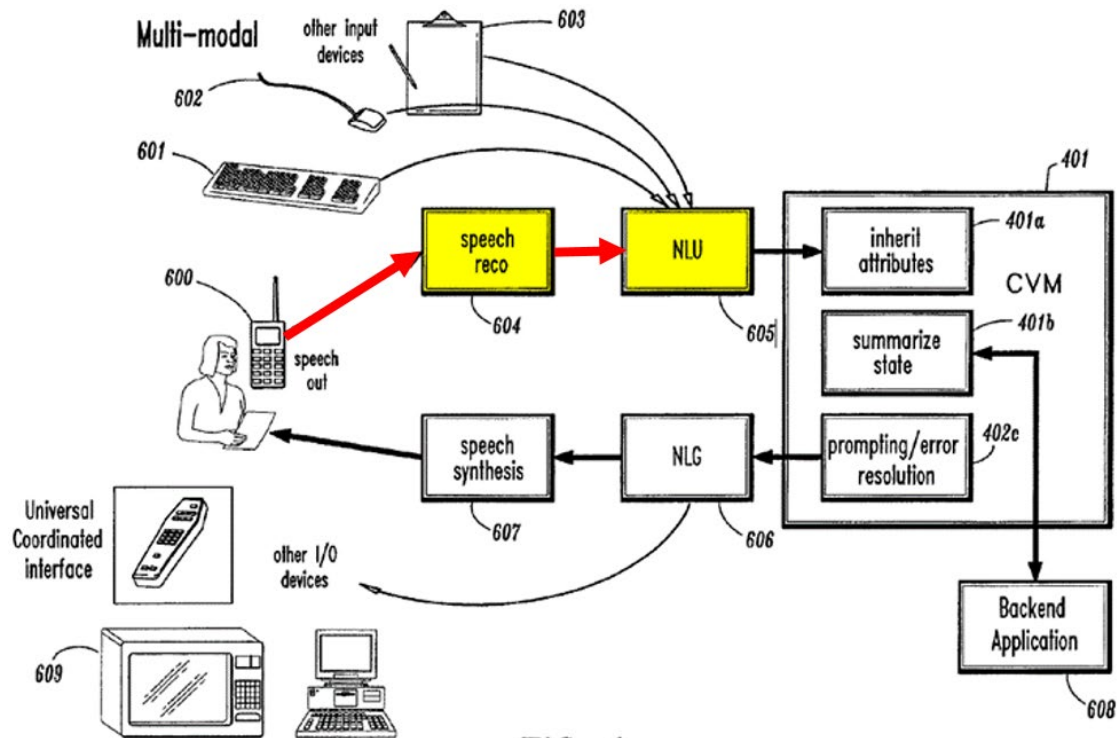


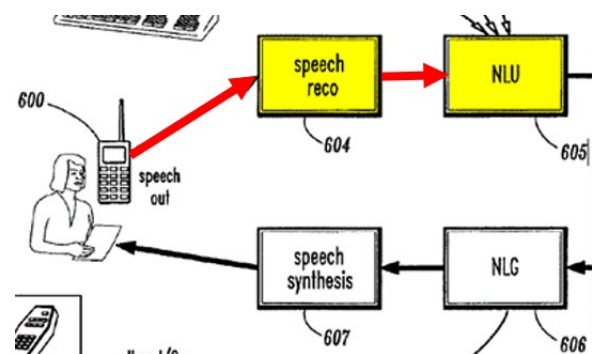
FIG. 6

(Coffman, Fig. 6 (annotations added); *see also id.*, 37:14-22 (“Spoken input from a speech client (e.g., telephone 600)...”).)

59. Coffman further discloses that its conversational computing system processes “**natural language utterances.**” In the context of the ’652 patent, and speech recognition and processing generally, “**natural language**” generally speaking refers to the forms of language used by a person for communication, as opposed to artificial or formal languages such as logics, algebras, and programming languages. (See, e.g., Dafydd Gibbon et al., *Handbook of Multimodal and Spoken Dialogue Systems : Resources, Terminology, and Product Evaluation* (2000) (“Gibbon”) [EX1008], p.444 (definition of “natural language”).) For example,

English is a natural language, while C++ and Java are programming languages. Consistent with this understanding, Coffman discloses several examples of natural language speech utterances, such as “I would like to open this,” “it should be synchronized with my PDA,” and “e-mail from my mother.” (Coffman, 6:32-7:3 (“For example, a mouse click or pointer/stylus pointing action followed by the command ‘I would like to open this’ is disambiguated into a set of attribute value pairs: Command: open, Object: Windows or task selected by the last mouse click.”), 48:1-3 (“Since user identity can be associated in the context of each discourse, command ambiguity can be immediately and transparently resolved (e-mail from my mother is correctly understood, independently of the user).”), 56:25-57:2 (“grocery list,” “it should be synchronized with my PDA,” and “Do it!”).)

60. Coffman’s processing of spoken “**natural language**” input is also confirmed by Coffman explaining that the spoken input, once recognized, is subjected to natural language parsing and understanding (“NLU”) processes, as illustrated by Figure 6 (excerpted and annotated at right). (Coffman, e.g., Fig. 6, 53:22-26 (“User inputs can be represented by their transcriptions or their mappings into a symbolic language after parsing and



NLU.”), 53:8-10 (“Voice input from a microphone (e.g. the standard voice input) can be arguments of function calls with the sequence of words, phones, or user identity or queries (symbolic language representation provided by NLU).”), 37:14-22 (“Spoken input from a speech client (e.g., telephone 600) is subject to a speech recognition process 604 and other input (e.g., keyboard, mouse clicks etc) are subject to NLU processing 605.”), 6:6-8 (“In another aspect, the CVM provides the capability to have natural dialog with NLU, NLG, contexts and mixed-initiatives sorted across multiple tasks, processes and discourses (with multiple domains).”), 5:1-4 (“In one aspect of the present invention, the CVM utilizes conversational subsystems (which may be local or distributed) including speech recognition, speaker recognition, text-to-speech, natural language understanding and natural dialog generation engines to understand and generate dialog between and user and machine.”), 37:29-32 (“The key conversational aspect is in the input procedure wherein commands can be entered in NLU (to provide natural language understanding of input queries) or in FSG mode (for constrained input according to rules: grammar and vocabulary, as opposed to free natural input).”).)

61. Coffman further discloses a system where “**recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests.**” This portion of the preamble describes nothing

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

more than a natural language-based system, such as Coffman's, that enables a user to input application commands or requests without having to recite the exact syntax of the command or request. For instance, as noted above, Coffman describes the system receiving the natural language speech utterance "I would like to open this," along with "a mouse click or pointer/stylus pointing action," and disambiguating that input "into a set of attribute value pairs: Command: open, Object: Windows or task selected by the last mouse click." (Coffman, 6:32-7:3.) Coffman explains that determining the command or request requires more than simply recognizing the words in the user's utterance, "I would like to open this." It requires natural language processing and use of a "context stack" to determine the particular application and command that should be invoked, as I will explain in more detail in connection with claim 1[a][v]. (Coffman, e.g., 29:20-32 (describing context stack), 42:1-5.) It also requires what Coffman refers to as "multi-modal processing" involving speech being used in parallel with mouse, keyboard or other input devices:

As illustrated, a conversational input interface according to an embodiment of the present invention can process multi-modal input, that is, files/streams/resources, speech via a phone 600, keyboard 601, pointing devices 602, handwriting devices 603, including natural interfaces. This means that all the input and output events across all the modalities are caught and transferred to the dialog manager (that also stores it appropriately in the context stack). Spoken input from a speech

client (e.g., telephone 600) is subject to a speech recognition process 604 and other input (e.g., keyboard, mouse clicks etc) are subject to NLU processing 605. Each input is subject to attribute acquisition (401a) whereby the attribute value n-uples are acquired from the input. A summarization process 401b is performed whereby the attribute value n-uples are added to the context and then verifies with the syntax of the back-end application 608 whether the query is complete, incomplete, or ambiguous.

(Coffman, 37:15-26; *see also id.*, 11:23-24 (“This is the notion of ‘multi-modality’ whereby speech is used in parallel with mouse, keyboard, and other input devices such as a pen.”).) Using Coffman’s example, “I would like to open this,” the system determines the command in part by processing additional input—i.e., the user’s “mouse click or pointer/stylus pointer action”—to identify the object of the user’s command—i.e., “Windows or task selected by the last mouse click.”

62. Coffman therefore discloses “[a] **system for processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests.**” And as shown below, the claimed system is obvious.

(a) **“one or more physical processors programmed with one or more computer program instructions which, when executed, cause the one or more physical processors to:” (1[a])**

63. 1[a] describes nothing more than routine and conventional components and operation of a generic computer system. Such a system operates by using physical processor(s) (e.g., a microprocessor) to execute computer program instructions. (See Ex. 1009, pp.338-339 (entry for “microprocessor” explaining that “[w]hen memory and power are added to a microprocessor, all the pieces, excluding peripherals, required for a computer are present.”), p.489 (defining “software” as “[c]omputer programs; instructions that make hardware work.”), p.424 (defining “program” as “[a] sequence of instructions that can be executed by a computer.”), p.200 (defining “execute” as “[t]o perform an instruction” and “impl[ying] loading the machine code of the program into memory and then performing the instructions” and “executable program” as “appl[ying] to a compiled program translated into machine code in a format that can be loaded into memory and run by a computer’s processor”).) Coffman explains that its “CVM system may be built on top of a conventional OS and APIs 15 and conventional device hardware 16 and located on a server or any client device (PC, PDA, Pvc).” (Coffman, 13:1-3.) A person of ordinary skill in the art, therefore, would have understood and found it obvious that Coffman’s conversational computing system would include **“one or more physical**

processors programmed with one or more computer program instructions.” As explained for the following limitations, a person of ordinary skill in the art would have further understood and found it obvious that executing the computer program instructions would cause the processor(s) to perform the claimed steps.

- (i) **“generate a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;” (1[a][i])**

64. Coffman’s conversational system includes a “**context stack**” where the state and history of each active task/process, such as an application process, is tracked in its own “context”:

The core of the CVM 401 is the context stack 405 which operates and is managed under the control of the dialog controller 404 (it is to be understood that the context stack 405 id [sic; is] directly related to the global history and meta information repository discussed above). In general, the context stack 405 accumulates the context (i.e., full query arguments list of attribute value n-uples, and state/mode) of each active process with an activated discourse (i.e., conversational interaction associated with a given task/process/thread) along with any data files 413 (or at least identifiers of such conversational arguments) for the different engines that may be needed for input understanding (e.g., files or arguments that the engines use for performing their respective tasks such as active FSG, topic, vocabulary, HMM (hidden markov models), voiceprints, language models or possible queries for a speech input). In

other words, the term “context” refers to the state of each discourse (whether active or normative), which keeps track of the past history of the discourse, its current state, and the specific characteristics and full query arguments of the corresponding task (e.g. vocabulary file, language model, parsing, tags, voiceprint, TTS rules, grammar, NLU etc. of each active task/process) along with any data needed for input understanding. It is to be appreciated that, in a distributed CVM (as explained below), the context stack (as well as other CVM components) may be directly associated with networked services (i.e., distributed over the network) (as described above with respect to the context and global history).

(Coffman, 29:18-30:3; *see also id.*, 30:4-10 (“More specifically, each new task, process, or thread creates a new stack entry and is associated with a discourse. Each application may be associated with multiple discourses (e.g. the application management discourse and the application content navigation discourses). Each context associated with a given discourse comprises the latest requests made to the corresponding process/task/thread as well as the latest output. The context of a given discourse is also associated with, e.g., any active grammars, vocabularies and symbolic language which maps the actual query.”).)

65. Coffman’s context stack comprises **“context information that corresponds to a plurality of prior utterances.”** Coffman explains, as shown in the excerpt above, that a context stack “accumulates the context (i.e., full query

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

arguments list of attribute value n-uples, and state/mode) or each active process.”⁴

(Coffman, 29:20-27.) Coffman further explains that “‘context’ refers to the state of each discourse (whether active or normative), which keeps track of the past history of the discourse, its current state, and the specific characteristics and full query arguments of the corresponding task.” (Coffman, 29:27-32.) Coffman therefore makes clear that a context stack comprises context information corresponding to a plurality of prior utterances, such as full query arguments list of attribute value n-uples for multiple applications that were derived from a user’s spoken input. (Coffman, e.g., 41:2-5 (“Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with speech inputs being transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active

⁴ An “n-uple” refers to a collection of “n” elements, where “n” is a number. Therefore, for example, a “4-uple” (or “quadruple”) refers to a collection of four elements. Coffman’s “attribute value n-uple” refers to an n-uple having “n” attribute-value pairs. Coffman’s example, “Command: open, Object: Windows or task selected by the last mouse click” based on the user’s utterance “I would like to open this” (Coffman, 6:32-7:3), is a 2-uple (or “duple”) because it includes two attribute-value pairs, (1) “Command: open” and (2) “Object: Windows or task selected by last mouse click.” These two attribute-value n-uples together are an example of a query argument list of attribute-value n-uples. (Coffman, 6:8-10 (input streams are “transcribed in an ASCII command or query (i.e., lists of attribute-value pairs or n-uples)”)), 6:26-28.)

query and complete it.”), 37:20-26 (“Each input is subject to attribute acquisition (401a) whereby the attribute value n-uples are acquired from the input. A summarization process 401b is performed whereby the attribute value n-uples are added to the context and then verifies with the syntax of the back-end application 608 whether the query is complete, incomplete, or ambiguous.”).)

66. Coffman further discloses a “**first context stack associated with a first device.**” Coffman explains that its conversational computing system can take the form of a distributed system involving multiple devices. For example, referring to Figure 3 (reproduced below), Coffman explains that “distributed conversational protocols 300 allow network conversational applications 200, 200a and network-connected devices (local client and other networked devices such as a server) to exchange information register their current conversational state, arguments (data files 209) and context with each other.” (Coffman, 22:32-23:3.)

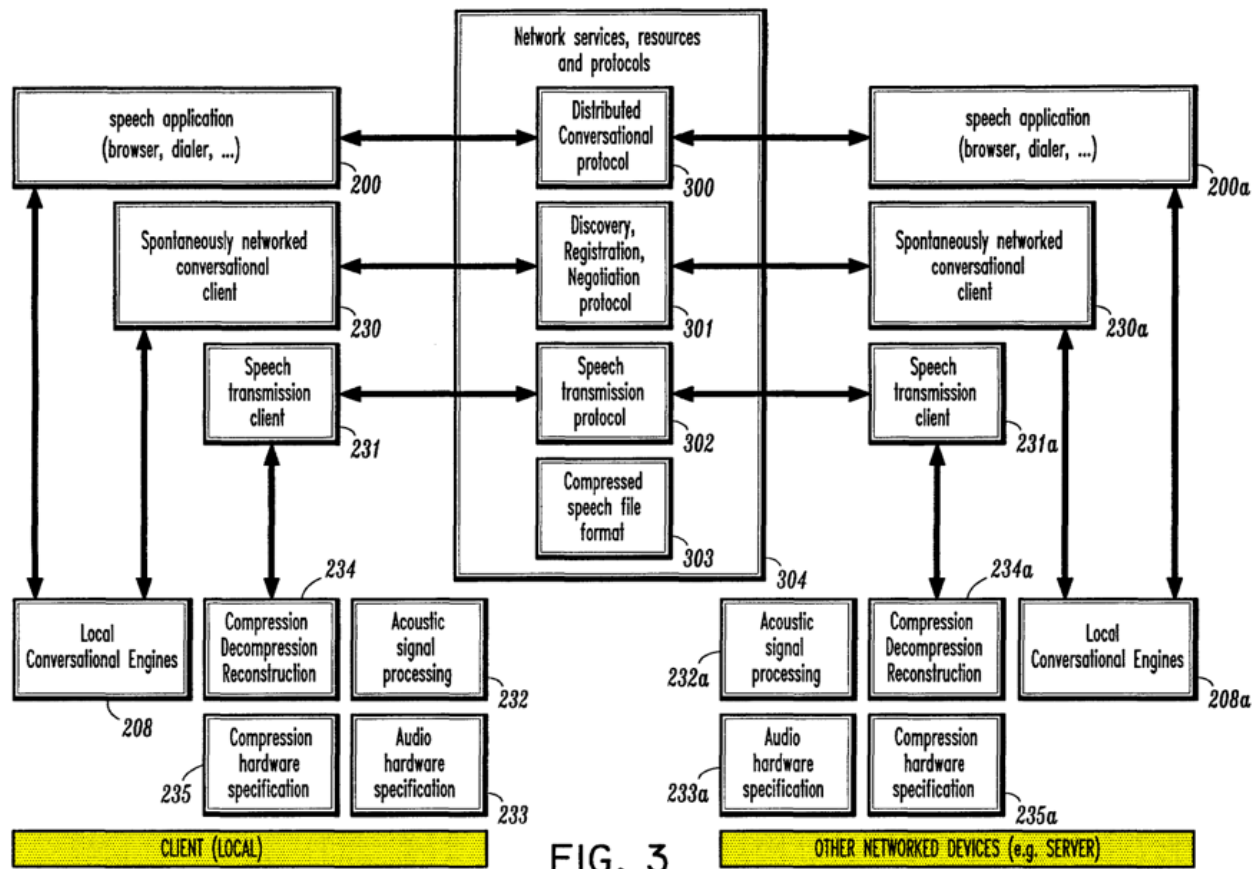


FIG. 3

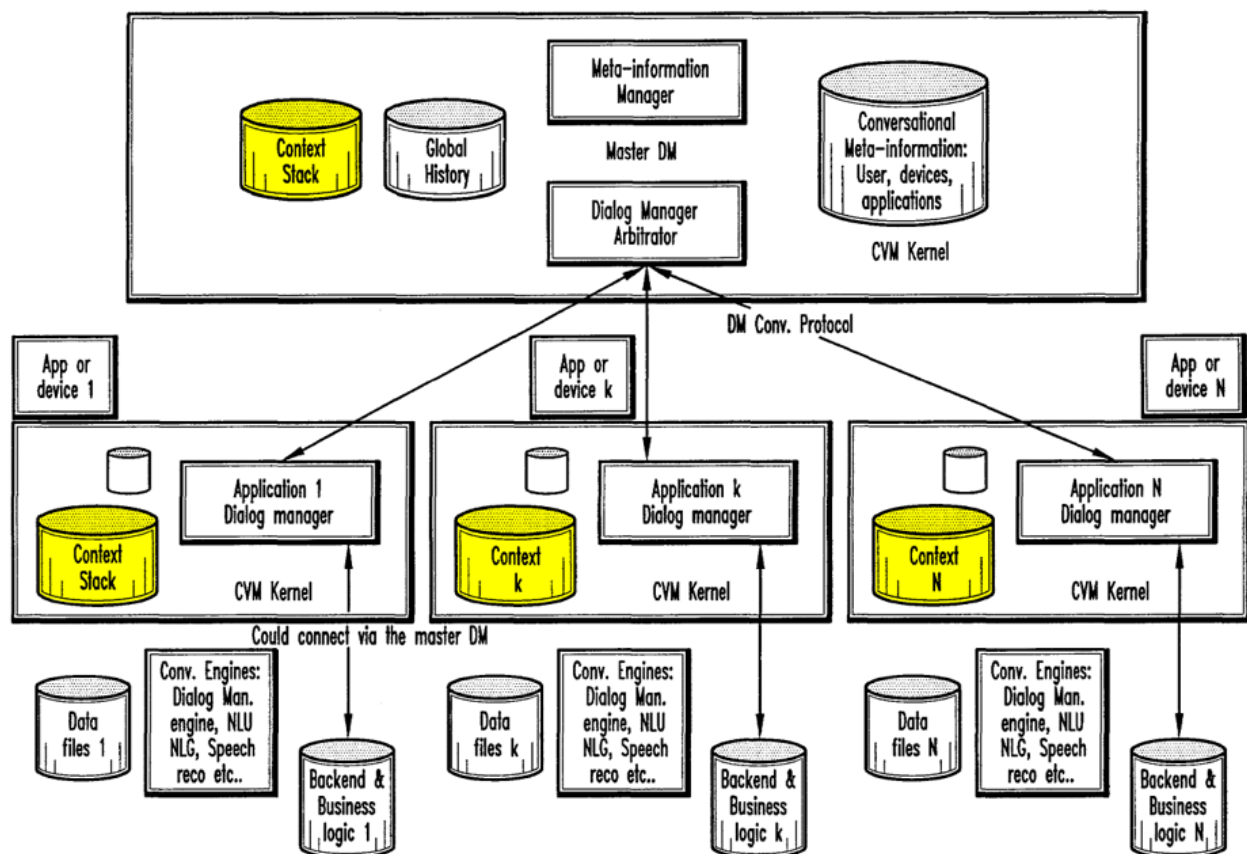
(Coffman, Fig. 3 (highlighting added to indicate multiple network-connected devices).) Coffman further explains that the devices in its conversational system “typically...will have their own...context stack 217, 217a”—i.e., each device has a “**context stack associated with [that] device**”:

Typically, different devices, CVMs or different applications will have their own dialog manager, context stack 217, 217a and global history 218, 218a. Through the dialog manager DM protocols (which are part of the distributed protocols 300 (Fig. 3), the different dialog managers will negotiate a topology with a master dialog manager and slave or peer dialog managers. The active master dialog manager (illustrated as

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

dialog manger 219 in Fig. 9) will be responsible for managing the flow of I/O to the different managers to decide the active dialog and appropriately execute a query and update the context and history.

(Coffman, 27:1-7.) Figure 10 of Coffman, for example, depicts a conversational system with multiple devices, each device having its own context stack (indicated in yellow highlighting):



(Coffman, Fig. 10 (highlighting added); *see also id.*, 27:24-26 (“Fig. 10 illustrates a hierarchical dialog between multiple dialog managers (i.e, the master arbitrator, and the slave dialog managers 1, k, and N) of various devices/applications (1, k and

N).”).)

67. With respect to the claimed “**first device**,” Coffman explains that Figure 10 “illustrates a typical master slave topology,” which “is formed by exchanging relevant information via the DM [dialog manager] protocols.” (Coffman, 27:26-28.) In Coffman, the “**first device**” could correspond to the device in a master slave topology having the master/central dialog manager, which is “responsible for managing the flow of I/O to the different managers to decide the active dialog and appropriately execute a query and update the context and history.” (Coffman, 27:5-7; *see also id.*, 29:18-20 (context stack operates and is managed under control of dialog manager/controller), 28:24-27 (dialog manager also referred to as dialog controller).)

68. Lastly, referring back to 1[a], with respect to “**physical processor(s)**” executing “**computer program instructions**” to “**generate**” the claimed context stack, a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to generate the context stack, consistent with the routine and conventional operation of generic computer systems, such as PCs and PDAs, as discussed for 1[a]. Moreover, as noted, for example, Coffman explains that the context stack is operated and managed under the control of dialog manager software,

which receives all input and output events across all modalities and “stores it appropriately in the context stack.” (Coffman, 29:18-20, 37:18-20; *see also id.*, 19:26-29 (describing dialog manager as “programming layer”), 30:4-10 (“More specifically, each new task, process, or thread creates a new stack entry and is associated with a discourse.”).)

- (ii) **“synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;”**
(1[a][ii])

69. As I discussed above for 1[a][i], Coffman discloses a distributed conversational computing system having a “master slave topology” where a master/central device coordinates with one or more slave devices, each device having its own context stack. (*E.g.*, Coffman, Figs. 3 & 10, 27:1-7, 27:24-26.) The claimed **“second context stack associated with a second device”** therefore corresponds in Coffman to a context stack associated with a slave device in a master slave topology.

70. Coffman further makes clear that a master device’s context stack (**“the first context stack”**) is **“synchronized”** with a slave device’s context stack (**“a second context stack associated with a second device”**), explaining:

With a master/slave network configuration, the master dialog manager

can compile all the lists and match the functions and needs with conversational capabilities. In addition, context information may be transmitted by indicating passing or pointing to the context stack/history of the device or application that the controller can access and add to its context stack.

(Coffman, 25:10-14.) Coffman further explains that, using dialog manager protocols, devices can exchange various kinds of information, including notification of “addition to contexts,” “context update” and “notification of context changes.” (Coffman, 27:7-20.)

71. A person of ordinary skill in the art would have further understood and found it obvious that, as a result of synchronizing device context stacks in Coffman’s system, **“the context information of the first context stack is updated based on related context information of the second context stack.”** The context information in the two context stacks would be **“related”** in at least two ways in Coffman. First, the context information is **“related”** because the associated devices themselves are related as a result of the master slave topology relationship. Second, the context information is **“related”** because, as Coffman explains, it can comprise **“context updates,”** which would update previous, related context information in the device’s context stack. The context information of the first stack would be **“updated”** based on the context information of the second context stack both by

adding information to the first context stack and by updating existing information in the first context stack.

72. Lastly, for substantially the same reasons as those explained for 1[a][i], a person of ordinary skill in the art would have understood and found it obvious that the device's processor(s) would execute software (computer program instructions) to synchronize the first context stack as claimed.

(iii) “receive a natural language utterance associated with a command or request;” (1[a][iii])

73. As discussed for the preamble, Coffman discloses “receiv[ing] a **natural language utterance**,” such as “I would like to open this.” (Coffman, 6:32-7:3; *see also id.*, 48:1-3 (“e-mail from my mother”), 56:25-57:2 (“grocery list,” “it should be synchronized with my PDA,” and “Do it!”).)

74. Coffman further discloses that the utterance is “**associated with a command or request**.” For example, Coffman explains that the user's utterance, “I would like to open this,” is associated with a command or request for the system to perform an “open” action. (Coffman, 6:32-7:3 (“For example, a mouse click or pointer/stylus pointing action followed by the command ‘I would like to open this’ is disambiguated into a set of attribute value pairs: Command: open, Object: Windows or task selected by the last mouse click.”).) Coffman more generally explains that the system “convert[s] voice requests into queries.” (Coffman, 13:8-

13 (“In general, conversational sub-systems 18 are responsible for converting voice requests into queries and converting outputs and results into spoken messages using the appropriate data files 17 (e.g., contexts, finite state grammars, vocabularies, language models, symbolic query maps etc.)”); *see also id.*, 38:26-27 (“[T]he input procedure provides a set of multi-mode input streams, each transcribed into an ASCII command, query, or list of attribute value n-uples.”).)

75. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to receive a natural language utterance as claimed in order to carry out further processing such as speech recognition and natural language understanding in Coffman’s computer-based system.

(iv) **“determine one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and” (1[a][iv])**

76. Coffman’s system includes a “speech recognition engine.” (Coffman, 10:28-30 (“In general, the CVM provides conversational APIs and protocols

between conversational subsystems (e.g. speech recognition engine, text-to speech etc.) and conversational and/or conventional applications.”), 29:1-4 (“The conversational system 400 further comprises a plurality of conversational resource subsystems (engines) 407 including, for example, a speech recognition system 408”),

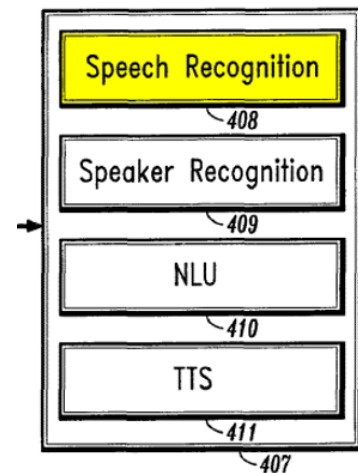


Fig. 4 (excerpted and highlighted right).) Coffman further makes clear that its speech recognition engine is used to **“determine one or more words of the natural language utterance by performing speech recognition on the natural language utterance,”** explaining that “[s]poken input from a speech client (e.g., telephone 600) is subject to a speech recognition process 604 and other input (e.g., keyboard, mouse clicks etc) are subject to NLU processing 605.” (Coffman, 37:20-22; *see also id.*, Fig. 6 (depicting input to and output from “speech reco” 604), 38:9-11 (“Each input stream is converted into its ASCII transcription and aligned with input time-marks via speech recognition processing 604.”).) As noted above, Coffman describes various examples of utterances of words and phrases, including “I would like to open this,” “e-mail from my mother,” “grocery list,” and “it should be synchronized with my PDA.” (Coffman, 6:32-7:3, 48:1-3, 56:25-57:2.)

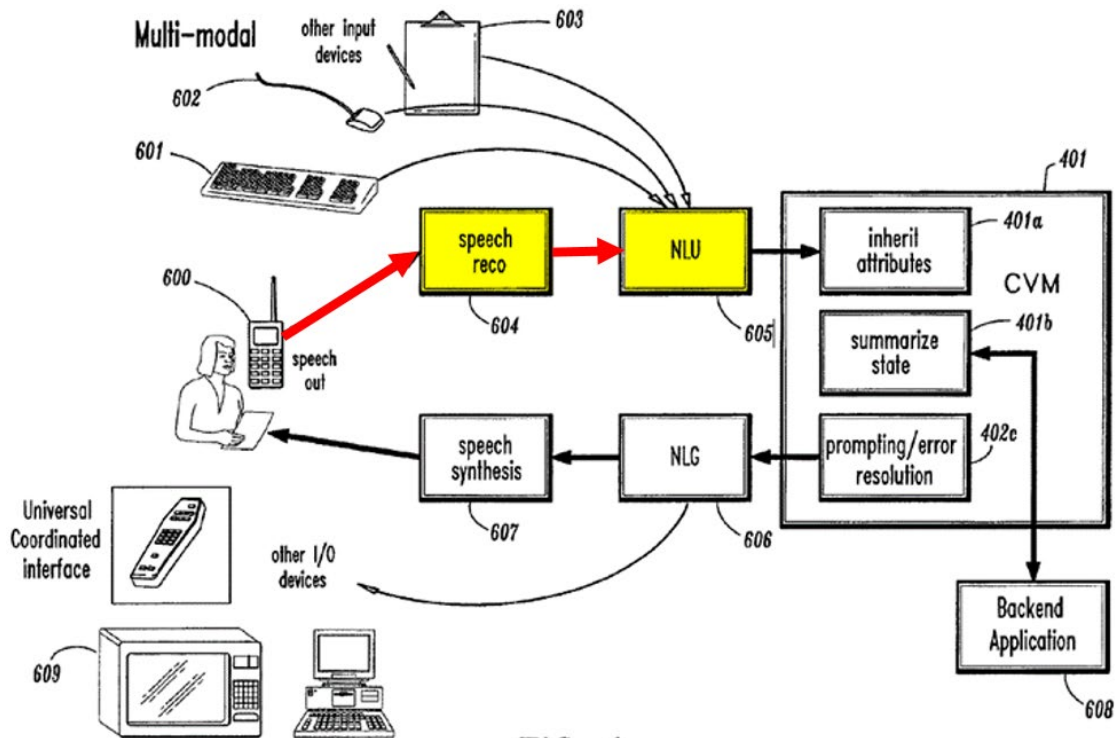


FIG. 6

(Coffman, Fig. 6 (annotations added); *see also id.*, 37:14-22 (“Spoken input from a speech client (e.g., telephone 600)....”).)

77. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to perform speech recognition on received natural language utterances.

(v) **“determine the command or request based on the one or more words and the updated context information.” (1[a][v])**

78. Coffman discloses that, in order to determine the command or request for the system to execute, recognized words in the utterance are matched against

context information in the context stack:

Completion/search for the active context is performed from context to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained and optionally confirmation is obtained from the user. As soon [sic] a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active.

(Coffman, 42:1-5; *see also id.*, 41:2-5 (“Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with speech inputs being transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active query and complete it.”), 30:8-10 (“The context of a given discourse is also associated with, e.g., any active grammars, vocabularies and symbolic language which maps the actual query.”), 51:12-14 (“Queries can be decoded, parsed and then translated into a logic combination of queries (symbolic query) using NLU technology.”), 40:12-15 (“By way of example, the task dispatcher/controller 402 will decide on the basis of the context stack 405 the different statistical parsers that must operate on a query for the dialog controller 404 to identify the active context and complete the query.”), 37:20-26 (“Each input is subject to attribute acquisition (401a) whereby the attribute value n-uples are

acquired from the input. A summarization process 401b is performed whereby the attribute value n-uples are added to the context and then verifies with the syntax of the back-end application 608 whether the query is complete, incomplete, or ambiguous.”.)

79. A person of ordinary skill in the art would have further understood and found it obvious that determining the command or request could take place after the master/central device had updated its context information, predictably resulting in determining the command or request **“based on the one or more words and the updated context information.”** (Coffman, e.g., 55:27-30 (“When a user interacts with the collection of devices, the interaction may always be via a central unit such as a PVA (personal vehicle assistant) 710 in a car, or a speech browser 713. The task dispatcher and context stack accumulates the contexts associated to all the devices and will parse and dispatch commands to each device accordingly.”).)

80. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to determine the command or request as claimed.

2. Claim 2: “The system of claim 1, wherein the one or more physical processors are further caused to:”

81. As explained below, it would have been obvious in Coffman’s system

for the device to perform the following steps. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device's processor(s) would execute software (computer program instructions) to perform those steps.

- (a) **“prompt a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and” (2[a])**

82. Coffman discloses prompting a user both (1) for **“additional information regarding the command or request,”** and (2) for **“confirmation regarding the command or request.”**

83. With respect to prompting a user for **“additional information regarding the command or request,”** Coffman explains that a user's command/query that is deemed incomplete or ambiguous “will generate a request (a prompt) for the missing or ambiguous information and update the context (requested missing fields).” (Coffman, 41:16-21; *see also id.*, 37:2-4 (“The first task to have its dialog completed by producing a non-ambiguous query is executed...”), 40:9-19 (“provide feedback to the user for incomplete or ambiguous queries/command”).)

84. With respect to prompting a user for **“confirmation regarding the command or request,”** Coffman explains that “new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is

obtained and optionally confirmation is obtained from the user.” (Coffman, 42:1-4; *see also id.*, 41:30-32 (“Command may also require user confirmation before execution based on the preferences/settings coming from the CVM platform, application, or user preferences.”), 42:18-20 (“If the request is complete, it will be executed, pending possible request for confirmation by the user, e.g. when it is irreversible.”), 56:30 (“You may need to confirm the selection if your PC CVM requests it.”).)

85. The prompted user in Coffman is “**associated with the natural language utterance**” for at least three reasons. First, the prompted user is associated with the utterance because it is the same user who provided the utterance in the first place. (Coffman, e.g., 40:9-19, 41:16-21, 42:1-4, 42:18-20.) Second, Coffman also discloses identifying the user who spoke the utterance, thereby associating the utterance with a user. Coffman explains that its conversational system includes functionality for “speaker recognition” and “speaker identification,” such as “text-independent speaker identification” using a user’s command or request. For example, Coffman explains that “the CVM utilizes conversational subsystems (which may be local or distributed) including speech recognition, speaker recognition, text-to-speech, natural language understanding and natural dialog generation engines to understand and generate dialog between and user and machine.

(Coffman, 5:1-4.) Figure 4 of Coffman depicts conversational system 400 with speaker recognition system 409:

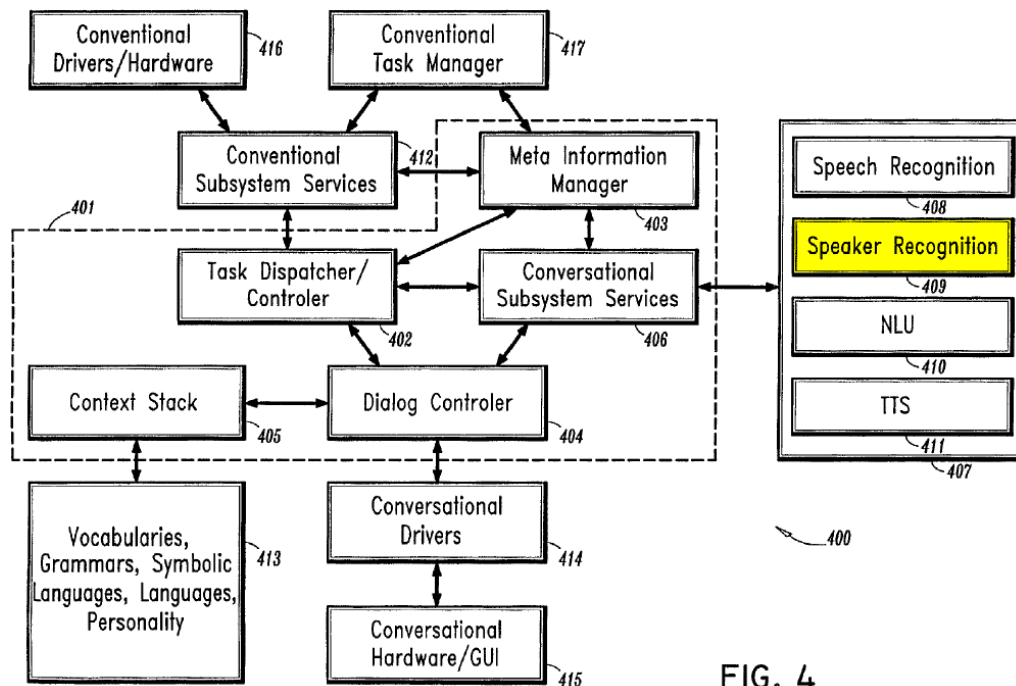


FIG. 4

(Coffman, Fig. 4 (highlighting added); *see also id.*, 29:1-4, 15:13-16:24 (listing various speaker-identification-related “conversational foundation classes” including “Acoustic Identify speaker,” “Acoustic Verify speaker” and “Speech biometrics identify.”).) Coffman further discloses that “[i]n a speech-based system, each command can be used to perform text-independent speaker identification.” (Coffman, 47:25-27; *see also id.*, 8:16-18 (“A user is automatically identified whenever a query to a resource is made.”).) “Text-independent speaker identification,” as its name suggests, is a form of speaker identification using some

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

features of a speech signal to determine the identity of the speaker of a given utterance that does not require the utterance to comprise specific linguistic content. (See Gibbon [EX1008], pp.474 (defining “speaker recognition” as “[a]ny decision-making process that uses some features of the speech signal to determine some information on the identity of the speaker of a given utterance”), 486 (defining “text-independent speaker recognition system” as “[a] speaker recognition system for which the linguistic content of test speech utterance varies across trials”).) Coffman’s speaker identification is therefore another way that Coffman discloses **“prompt[ing] a user associated with the natural language utterance.”**

86. Coffman further explains that “user identity can be associated in the context of each discourse,” so that “command ambiguity can be immediately and transparently resolved (e-mail from my mother is correctly understood, independently of the user),” further confirming that a user is **“associated with the natural language utterance.”** (Coffman, 48:1-3; *see also id.*, 46:26-31 (“the stack of contexts should contain the identity of the user as the most recently authenticated identity”), 48:11-14 (“[W]ith respect to conversational multi-users and conversational security, it is to be appreciated that dialogs, categories, meta-information, and access to resources can be a function of the identity of the user and its associated meta-information history. And conversely, the conversational

information collected on a query can be used to recognize the user.”.)

- (b) **“receive a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.” (2[b])**

87. Coffman does not appear to detail the type of input Coffman’s conversational computing system receives in response to a prompt for additional information or confirmation. But it would have been obvious to a person of ordinary skill in the art for the system to receive **“a non-speech input.”** This is because Coffman describes a “multi-modal” system that can receive both speech input and non-speech input, such as a mouse click, to use in carrying out system operations. For example:

As illustrated, a conversational input interface according to an embodiment of the present invention can process multi-modal input, that is, files/streams/resources, speech via a phone 600, keyboard 601, pointing devices 602, handwriting devices 603, including natural interfaces. This means that all the input and output events across all the modalities are caught and transferred to the dialog manager (that also stores it appropriately in the context stack). Spoken input from a speech client (e.g., telephone 600) is subject to a speech recognition process 604 and other input (e.g., keyboard, mouse clicks etc) are subject to NLU processing 605. Each input is subject to attribute acquisition (401a) whereby the attribute value n-uples are acquired from the input.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

(Coffman, 37:15-23; *see also id.*, 11:23-24 (“This is the notion of ‘multi-modality’ whereby speech is used in parallel with mouse, keyboard, and other input devices such as a pen.”), 12:8-10 (“The ‘multi-modality’ aspect of the CUI implies that various I/O resources such as voice, keyboard, pen, and pointing device (mouse), keypads, touch screens, etc can be used in conjunction with the CVM platform.”), 6:30-7:3 (“Indeed, any input in any modality is captured by the dialog manager of the CVM kernel layer as an event that is added to the associated context or context stack. For example, a mouse click or pointer/stylus pointing action followed by the command ‘I would like to open this’ is disambiguated into a set of attribute value pairs: Command: open, Object: Windows or task selected by the last mouse click.”).) In fact, Coffman expressly acknowledges that non-speech input can be used as an alternative to speech input, explaining that “[f]or example, with one conversational request (or by pressing a button on a keyboard or clicking a mouse or providing a key), the user can listen to the output of a task or re-claim the input....” (Coffman, 35:2-4.)

88. A person of ordinary skill in the art would have also understood and found it obvious that **“the command or request is determined further based on”** the user’s input regarding the additional information or confirmation (**“the non-speech input”**). As discussed for 2[a], Coffman explains that the additional

information enables the system to complete or resolve any ambiguity in the user's input so that the system can determine correct application and function to perform. (Coffman, 41:16-21, 37:2-4, 40:9-19.) Likewise, user confirmation allows the system to determine that it correctly understood user input and that it should actually carry out the operations that it prompted the user about. (Coffman, e.g., 42:18-20, 56:30.)

3. Claim 3: “The system of claim 1, wherein the first context stack includes a plurality of context entries, and wherein the one or more physical processors are further caused to:”

89. As discussed above for claim 1, Coffman explains that a context stack includes “**a plurality of context entries.**” (Coffman, 30:4-10 (“More specifically, each new task, process, or thread creates a new stack entry and is associated with a discourse. Each application may be associated with multiple discourses (e.g. the application management discourse and the application content navigation discourses).”), 30:13-16 (“The dialog controller 404 manages the context stack 405 by creating a new stack entry in the context stack 405 for each new task/process/thread that is spawned either local or remotely from a networked device (with task management being controlled by the task dispatcher/controller 402 as discussed below).”), 41:8-9 (“Each new (active) task/process/thread creates a new stack entry in the context stack 405, with or without activated discourse.”), 40:25-

27 (“Queries that are rejected or too improbable cause the dialog controller 404 to peruse down the context stack 405 to look for the next available context.”), 42:1-2 (“Completion/search for the active context is performed from context to context down the stack.”).) A person of ordinary skill in the art would have therefore understood and found it obvious that **“the first context stack”** would **“include a plurality of context entries.”**

- (a) **“identify, from among the plurality of context entries, one or more context entries that correspond to the one or more words, wherein the updated context information includes the one or more context entries.”**
(3[a])

90. Coffman discloses **“identify[ing], from among the plurality of context entries, one or more context entries that correspond to the one or more words.”** Coffman describes searching through a context stack to identify a context entry that matches the user’s utterance:

Completion/search for the active context is performed from context to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained and optionally confirmation is obtained from the user. As soon [sic] a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active.

(Coffman, 42:1-5.)

91. A person of ordinary skill in the art would have further understood and found it obvious that identifying the one or more context entries could take place after the master/central device had updated its context information in its context stack, predictably resulting in identifying a context entry “**wherein the updated context information includes the one or more context entries.**” (Coffman, e.g., 55:27-30 (“When a user interacts with the collection of devices, the interaction may always be via a central unit such as a PVA (personal vehicle assistant) 710 in a car, or a speech browser 713. The task dispatcher and context stack accumulates the contexts associated to all the devices and will parse and dispatch commands to each device accordingly.”).)

92. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to perform this step.

4. Claim 4: “The system of claim 3, wherein identifying the one or more context entries comprises:”

93. As explained below, a person of ordinary skill in the art would have understood and found it obvious for Coffman’s process of identifying one or more context entries to comprise the following steps.

94. For substantially the same reasons as those explained for 1[a], a person

of ordinary skill in the art would have understood and found it obvious that the device's processor(s) would execute software (computer program instructions) to perform these steps.

(a) **“comparing the plurality of context entries with the one or more words;” (4[a])**

95. Coffman makes clear that its process of identifying a context entry that matches the user's utterance involves **“comparing the plurality of context entries with the one or more words.”** As just discussed for claim 3, Coffman describes comparing context entries in a context stack to the user's recognized utterance. (Coffman, 42:1-5; *see also id.*, 41:2-5 (“Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with speech inputs being transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active query and complete it.”), 37:2-4 (“The first task to have its dialog completed by producing a non-ambiguous query is executed and the corresponding query as interpreted by the other application is stored to activate if the recognized query is rejected by the user.”).) For instance, taking Coffman's example, “I would like to open this,” Coffman teaches that the system compares the recognized word “open” in the user's utterance to a context entry for an application's “open” command. (Coffman, 6:32-7:3 (“For example, a mouse click or

pointer/stylus pointing action followed by the command ‘I would like to open this’ is disambiguated into a set of attribute value pairs: Command: open, Object: Windows or task selected by the last mouse click.”).)

(b) “generating one or more rank scores for individual context entries of the plurality of context entries based on the comparison; and” (4[b])

96. Coffman describes how identifying a context entry involves searching the context stack for a context entry match that is sufficiently “acceptable” and not “too improbable.” A context stack is searched to find a context that is “an acceptable match” by “fit[ting] the utterance at the level of the NLU symbolic language.” (Coffman, 42:1-5 (“That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained and optionally confirmation is obtained from the user. As soon [sic] a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active.”).) If the match with a context in the context stack is “too improbable,” then the next context in the context stack is considered for matching. (Coffman, 40:24-27 (“Rejection or likelihood of a new query is managed by each activated task based on the conversational arguments. Queries that are rejected or too improbable cause the dialog controller 404 to peruse down the context stack 405 to look for the next available context.”).)

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

97. A person of ordinary skill in the art would have understood and found it obvious that Coffman's technique of determining whether a context entry was an "acceptable match" or "too improbable" would involve "**scoring**" the context. The concept of "scoring" has been well known in the field of speech recognition and processing since at least the 1970s. For example, Coffman incorporates by reference International Patent Application Publication No. WO 00/21075 (**Maes** [EX1007]) (Coffman, 22:25-29 ("...patent application IBM Docket No. YO999- 113P, filed concurrently herewith, entitled 'System and Method for Providing Network Coordinated Conversational Services,' which is commonly assigned and incorporated herein by reference...").) I am informed by counsel for Petitioner that, because Coffman incorporates Maes by reference, Maes is effectively part of Coffman as if it were explicitly contained in it. Maes details a number of techniques for generating scores in connection with speech recognition. (Maes, 17:23-19:8.) Maes further describes comparing speech recognition scores against a threshold in order to determine whether the scores are acceptable. (Maes, 16:20-17:7 ("[T]he remote server is contacted when these scores are determined to fall below a given threshold....[A]ny suitable measure or approach to estimate the confidence or the recognized utterance or query (during or after the dialog) can be used to determine if the results obtained by a conversational system are acceptable or not").) Similarly,

in Coffman, with respect to matching context entries, a person of ordinary skill would have understood that determining whether a context was “acceptable” or “too improbable” would involve calculating some sort of probability-related score for the context entry.

98. Coffman’s context entry scoring teaches “**generating one or more rank scores**” in at least three ways. First, the scoring would be a “**rank score**” because it would serve to rank the context against some level of acceptability—similar to a pass/fail ranking system. By scoring “pass,” a context entry is selected by the system. (Coffman, 42:1-5.) Second, the scoring would be a “**rank score**” because it would serve to determine the order of activating matching context entries. Coffman makes clear that, even after identifying a first matching entry, its context-matching process continues to evaluate other context entries to identify and queue-up additional matching entries to potentially activate. (Coffman, 37:2-4 (“The first task to have its dialog completed by producing a non-ambiguous query is executed and the corresponding query as interpreted by the other application is stored to activate if the recognized query is rejected by the user.”), Coffman, 40:24-27 (“Queries that are rejected or too improbable cause the dialog controller 404 to peruse down the context stack 405 to look for the next available context.”).) Third, the scoring would be a “rank score,” because determining that the context entry is

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

an acceptable match results in reordering the context stack to move the matched context entry to the top of the stack. (Coffman, 42:4-7 (“As soon [sic] a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active. Until the active command is completed, or until a new command is provided, the selected context is marked active, and pushed to the top of the context stack 405.”).) Therefore, Coffman's context stack is rank ordered, where rank indicates recency in the conversational interaction, and the score, based on this ordering, is also ranked by recency combined with linguistic fit with the input speech. These ways in which Coffman discloses “**generating one or more rank scores**” mirror the '652 written description, which states that “the possible contexts are scored and the top one or few are used for further processing” and that “[t]he parser **118** uses a scoring system to determine the most likely context or domain for a user’s question or command.... Based on this scoring, the system **90** invokes the correct agent. If 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.”

('652, 34:13-24.)⁵

99. I note that the concept of “rank” or “ranking” broadly encompasses the simple act of assigning something to a particular position or category, and does not require, for example, comparing two things to each other to determine which is first and which is second. (See EX1005, p.003 (“**rank** ... v.t... 19. to assign to a particular position, station, class, etc.”).) To use a simple real-world analogy, enlisted military personnel are assigned a rank such as Private, Corporal, Sergeant, and so forth. The assignment of a rank for a particular soldier is “objective” rather than “comparative” because it is based on that soldier’s individual experience, qualifications, performance, and other characteristics. For example, if the military chooses to assign a rank of “Sergeant” to that soldier (for example, as a promotion from “Corporal”), that assignment can occur independently of the assigned ranks of other personnel. As another example, Karate and other forms of martial arts maintain a belt ranking system in which different ranks are based on different colored belts, starting with White Belt and continuing with darker colors to Black Belt. (E.g.,

⁵ Notably, Kennewick includes this same disclosure verbatim (Kennewick, ¶¶0152-0153), further confirming the knowledge of a person of ordinary skill in the art and the obviousness of scoring and rank scores in Coffman’s system.

EX1010, pp.006-0015.) Whether a Karate participant is assigned the rank of Black Belt, for example, is an objective rather than comparative assignment based on that participant's mastery of the qualifications for that ranking, and as with the military example, the ranking assignment can occur independently of the belt ranking that has been assigned to others who are also practicing Karate. Both of these examples are properly referred to as assigning a "ranking" to a person even though the ranking assigned was objective and independent of rankings assigned to other people.

100. Coffman therefore discloses and renders obvious **"generating one or more rank scores for individual context entries of the plurality of context entries based on the comparison."**

(c) **"identifying, from among the plurality of context entries, the one or more context entries based on the one or more rank scores." (4[c])**

101. My analysis for 4[b] addressed 4[c]. As discussed, Coffman describes identifying a context entry in a context stack to activate based on whether the context match is acceptable and not too improbable—i.e., **"based on the one or more rank scores."** (Coffman, 40:24-27, 42:1-7.)

5. Claim 5: “The system of claim 4, wherein the plurality of context entries are ordered in the first context stack, and wherein the one or more physical processors are further caused to update the ordering of the plurality of context entries in the first context stack based on the one or more rank scores.”

102. As discussed for 4[b], Coffman discloses that, based on matching a context entry, the matched entry is moved to the top of the context stack, thereby **“updat[ing] the ordering of the plurality of context entries”** in the context stack. (Coffman, 42:1-7 (“As soon [sic] a context is found that fits the utterance at the level of the NLU symbolic language, the context becomes active and the corresponding process becomes active. Until the active command is completed, or until a new command is provided, the selected context is marked active, and pushed to the top of the context stack 405.”).)

103. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to perform this step.

6. Claim 6: “The system of claim 5, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:”

104. For the reasons discussed for 2[a], Coffman discloses **“the natural language utterance is associated with a user”** in three ways—first, simply because

by providing the natural language utterance, the user is associated with that utterance; second, by performing speaker recognition/identification on a user's utterance; and third, by associating user identity with context to determine commands to perform. (Coffman, e.g., Fig. 4 (speaker recognition), 5:1-4, 29:1-4, 15:13-16:24, 47:25-27, 8:16-18, 48:1-3, 46:26-31, 48:11-14.)

105. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device's processor(s) would execute software (computer program instructions) to perform the following steps.

(a) “receive a subsequent natural language utterance associated with the user; and” (6[a])

106. 6[a] simply corresponds to the master/central device in Coffman's conversational system, as described for claim 1, receiving another utterance from the user, such as a new command or request. For example, 6[a] would be satisfied by the system, after receiving the user's utterance “I would like to open this,” subsequently receiving another utterance from that user, such as “email from my mother” or “it should be synchronized with my PDA.” (Coffman, 6:32-7:3 (“I would like to open this”), 48:1-3 (“e-mail from my mother”), 56:25-57:2 (“grocery list,” “it should be synchronized with my PDA,” and “Do it!”); *see also id.*, 42:5-7 (“Until the active command is completed, or until a new command is provided, the selected

context is marked active, and pushed to the top of the context stack 405.”).) Coffman expressly describes a scenario where a user’s PC in a distributed conversational system receives a series of natural language utterances from the user to perform various actions. (Coffman, 56:26-57:2.)

(b) “determine one or more subsequent words of the subsequent natural language utterance by performing speech recognition on the subsequent natural language utterance;” (6[b])

107. 6[b] simply corresponds to the master/central device in Coffman’s conversational system, as described for claim 1, performing speech recognition on the user’s subsequent natural language utterance. It is disclosed and obvious for the reasons explained for 1[a][iv].

(c) “identify, from among the plurality of context entries, one or more other context entries that correspond to the one or more subsequent words based on the updated ordering; and” (6[c])

108. 6[c] simply corresponds to the master/central device in Coffman’s conversational system, as described for claim 1, identifying one or more context entries in the context stack corresponding to the recognized words, as discussed for 3[a]. (Coffman, e.g., 42:1-5.) And as discussed for claim 5, the context stack would have had an updated ordering as a result of processing the previous user utterance. (Coffman, 42:1-7 (“As soon [sic] a context is found that fits the utterance at the level

of the NLU symbolic language, the context becomes active and the corresponding process becomes active. Until the active command is completed, or until a new command is provided, the selected context is marked active, and pushed to the top of the context stack 405.") As noted, Coffman explains that the process of identifying a matching context involves searching down the context stack. (Coffman, 40:24-27 ("Queries that are rejected or too improbable cause the dialog controller 404 to peruse down the context stack 405 to look for the next available context."), 42:1-5 ("That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained and optionally confirmation is obtained from the user.")) Therefore, the claimed "**identify[ing]**" would have been "**based on the updated ordering.**"

(d) "determine a subsequent command or request based on the one or more subsequent words and the one or more other context entries." (6[d])

109. 6[d] simply corresponds to the master/central device in Coffman's conversational system, as described for claim 1, determining a command or request based on the recognized words in the subsequent utterance and the one or more identified context entries, as described for 6[b] and 6[c], respectively. It is disclosed and obvious for the reasons explained for 1[a][iv] and 3[a]. (Coffman, e.g., 42:1-5, 41:2-5; *see also id.*, 30:8-10, 37:20-26, 40:12-15, 40:20-24.)

7. **Claim 7: “The system of claim 3, wherein identifying the one or more context entries comprises identifying, from among the plurality of context entries, the one or more context entries that most closely correspond to the one or more words.”**

110. It would have been obvious for the identified one or more context entries in Coffman to be **“the one or more context entries that most closely correspond to the one or more words.”** This is because, as discussed, Coffman explains that the system can prompt the user to confirm that the identified context entry correctly corresponds to the user’s input. (Coffman, 42:1-5, 41:30-32, 42:18-20.) By providing confirmation, the user indicates that the identified context most closely corresponds to the user’s utterance.

8. **Claim 8: “The system of claim 3, wherein the one or more physical processors are further caused to:”**

111. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the device’s processor(s) would execute software (computer program instructions) to perform the following steps.

- (a) **“identify one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and” (8[a])**

112. 8[a] recites a **“domain agent.”** Generally speaking, in the software

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

context, an “agent” broadly refers to a program that performs actions on behalf of another entity, such as a user or another software program or computing system. For example, a web browser is sometimes referred to more technically as an “HTTP user agent” (HTTP being short for Hypertext Transfer Protocol, a communication protocol typically associated with the World Wide Web). Similarly, desktop email software (such as Microsoft Outlook) is sometimes referred to as a “mail user agent” or MUA. Consistent with this general understanding, the ’652 patent describes a “**domain agent**” as an agent that organizes “domain specific behavior and information” and “may be data-driven, scripted or created with compiled code.” (’652, 27:14-17, 28:11-12.)

113. A “**domain agent**” in Coffman corresponds to the executable software of a conversationally-aware application (and associated data, such as vocabulary, grammar, language models and NLU symbolic mapping). (Coffman, e.g., 13:26-29 (“The code implementing the applications, API calls and protocol calls includes interpreted and compiled scripts and programs, with library links, conversational logic (as described below) engine call and conversational foundation classes.”).) Conversationally-aware applications “are applications that are specifically programmed for operating with a CVM core layer (or kernel) 14 via conversational application APIs 13.” (Coffman, 12:28-30.) A conversationally aware application

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

operates with a CVM by providing its data files to the CVM for use by the conversational subsystems—an application hooks into the CVM API by “registering all its conversational capabilities and requirements” including “[a]ctive vocabulary, finite state grammar and language models to control the application” and “[s]ymbolic mapping if NLU is supported or at list [sic; least] a context state list.” (Coffman, 57:23-27; *see also id.*, 36:31-32 (“each application will provide the CVM 401 its own parsing and translation arguments”).)

114. A conversationally-aware application, consistent with the understanding of “agent,” receives, processes, and responds to user questions, queries and commands. (Coffman, e.g., 13:8-13 (queries transformed into “application calls” and application output returned to user), 37:2-4 (task executes query), 40:9-19 (“execute completed queries” and “output of executed or running processes”).) A conversationally-aware application, consistent with the ’652 patent’s description of “domain agent,” also usually provides functionality related to a specific domain, such as a particular subject matter (such as email or spreadsheet) or device (such as a PDA or smart phone). (Coffman, 16:35-17:7 (listing “application specific” foundation classes, including “Manage_e-mail,” “Manage_calendar,” and “Manage_addressbook/director”), 56:11-14 (e-mail), 58:30-32 (spreadsheet), 56:22-25 (“conversational PDA,” “conversational car PC

(PVA, Personal Vehicle Assistant),” and “conversational smart phone”), 36:29-31 (“More specifically, the present invention provides the capability to have a natural dialog with NLU, NLG and mixed initiative across multiple applications, with multiple domains.”).)

115. A conversationally-aware application in Coffman is “**associated with the one or more context entries.**” As discussed for claim 1 [a][i], Coffman explains that context stack entries are created for and associated with applications. (Coffman, 30:4-10 (“More specifically, each new task, process, or thread creates a new stack entry and is associated with a discourse. Each application may be associated with multiple discourses (e.g. the application management discourse and the application content navigation discourses.”); *see also id.*, 29:18-30:3.)

116. A conversationally-aware application in Coffman is “**configured to process the command or request.**” As discussed for claim 1 [a][v], Coffman describes completing and activating a context entry’s command/query for the associated application to process. (Coffman, 37:2-4 (“The first task to have its dialog completed by producing a non-ambiguous query is executed and the corresponding query as interpreted by the other application is stored to activate if the recognized query is rejected by the user.”); *see also id.*, 42:1-5, 41:2-5, 40:12-15 (“By way of example, the task dispatcher/controller 402 will decide on the basis of

the context stack 405 the different statistical parsers that must operate on a query for the dialog controller 404 to identify the active context and complete the query.”.)

(b) “generate a response to the command or request using the one or more domain agents.” (8[b])

117. As discussed above, Coffman discloses processing a command or request using a conversationally-aware application (“**domain agent**”). (Coffman, e.g., 37:2-4.) Coffman further discloses that a conversationally-aware application returns output as a result of processing the command or request, which is used to generate a response to provide to the user. For example, Coffman describes executing completed queries and “return[ing] output of executed or running processes” and presenting them to user using, for example, text-to-speech, as Coffman illustrates in Figure 6:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

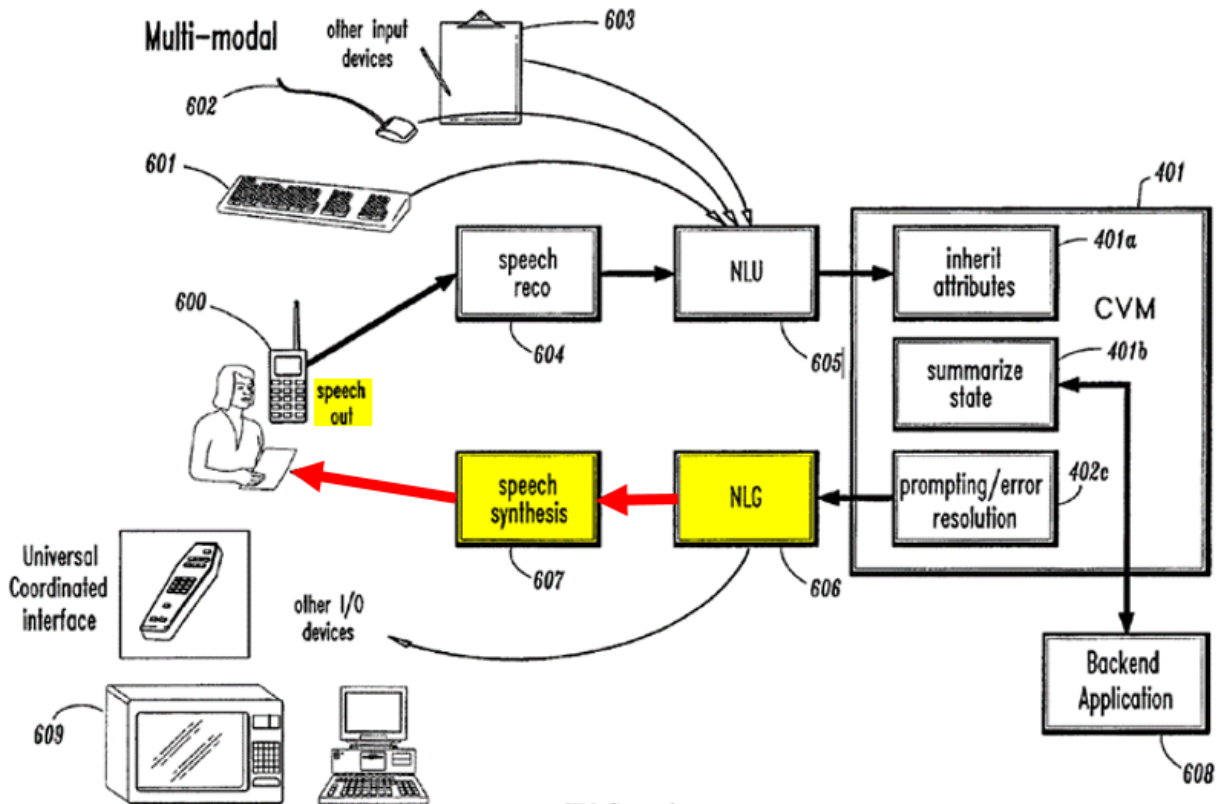


FIG. 6

(Coffman, Fig. 6 (annotations added), 40:9-19; *see also id.*, 13:8-13 & 58:9-11 (CVM converts application “output into speech...provided to the user”), 39:3-5 (“With regard to conversational output dispatches and interface, the CVM 401 produces output to files/streams/resources, display (single or multi-windows, GUI, color, images, movies), audio. Individually, each output stream behaves conventionally.”).)

9. **Claim 9: “The system of claim 8, wherein each of the one or more domain agents comprises domain knowledge associated with a particular domain, the domain knowledge comprising: one or more of (i) a keyword; (ii) a link to an information source; (iii) a list of responses associated with a plurality commands or requests; (iv) a substitution list used to format the plurality of commands or requests; or (v) content including dictionaries, encyclopedias, or almanacs.”**

118. As explained for claim 8[a], a “**domain agent**” in Coffman corresponds to the executable software of a conversationally-aware application (and associated data, such as vocabulary, grammar, language models and NLU symbolic mapping). (Coffman, e.g., 57:23-27, 36:31-32.)

119. With respect to claim 9, a conversationally-aware application’s “**domain knowledge**” includes its data, such as its vocabulary, grammar, language model(s) and NLU symbolic mapping. This data is “**associated with a particular domain**” and comprises at least “**a keyword**” because it is supplied by the application in connection with registration and includes words that Coffman’s system uses to recognize a user’s utterance and map it to a query associated with an application. For example, using Coffman’s example, the word “open” (“**keyword**”) in the request “I would like to open this” matches an “open” command in an application’s vocabulary and associated symbolic language. (Coffman, 6:30-7:3.) More generally, Coffman describes “match[ing]” recognized words and phrases against an NLU symbolic language and “mapping” user input into a symbolic

language. (Coffman, 42:1-7; *see also id.*, 53:22-26 (“User inputs can be represented by their transcriptions or their mappings into a symbolic language after parsing and NLU.”).) Coffman’s example is an instance of the well-known concept of “keyword detection,” also known as “word spotting.” (Gibbon [EX1008], p.428 (defining “keyword detection” as “[t]he detection of a given word or word sequence in a stream of speech or text” and identifying “word spotting” as a synonym).)

10. Independent Claim 14

120. The table below provides a side-by-side comparison of claim 1 and claim 14, with underlining added to show identical language. As the table shows, claim 14 is not meaningfully different from claim 1 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 1. It therefore would have been obvious for the reasons explained for claim 1.

Claim 1	Claim 14
1. A system for <u>processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests</u> , the system comprising:	A computer-implemented method of <u>processing natural language utterances where recognized words of the natural language utterances alone are insufficient to completely determine one or more commands or requests</u> ,
<u>one or more physical processors programmed with one or more computer program instructions which, when executed</u> , cause the one or more physical processors to:	the method being implemented by a computer system that includes <u>one or more physical processors executing one or more computer program instructions which, when executed</u> , perform the method, the method comprising:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 1	Claim 14
<u>generate</u>	<u>generating</u> , by the one or more physical processors,
<u>a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;</u>	<u>a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;</u>
<u>synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;</u>	<u>synchronizing the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;</u>
<u>receive</u>	<u>receiving</u> , at the one or more physical processors,
<u>a natural language utterance associated with a command or request;</u>	<u>a natural language utterance associated with a command or request;</u>
<u>determine</u>	<u>determining</u> , by the one or more physical processors,
<u>one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and</u>	<u>one or more words of the natural language utterance by performing speech recognition on the natural language utterance; and</u>
<u>determine</u>	<u>determining</u> , by the one or more physical processors,
<u>the command or request based on the one or more words and the updated context information.</u>	<u>the command or request based on the one or more words and the updated context information.</u>

11. Claim 15

121. The table below provides a side-by-side comparison of claim 2 and claim 15, with underlining added to show identical language. As the table shows, claim 15 is not meaningfully different from claim 2 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 2. It therefore would have been obvious for the reasons explained for claim 2.

Claim 2	Claim 15
2. The system of claim 1, wherein the one or more physical processors are further caused to:	15. The method of claim 14, further comprising:
<u>prompt</u>	<u>prompting</u> , by the one or more physical processors,
<u>a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and</u>	<u>a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and</u>
<u>receive</u>	<u>receiving</u> , at the one or more physical processors,
<u>a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.</u>	<u>a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.</u>

12. Claim 16

122. The table below provides a side-by-side comparison of claim 3 and

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

claim 16, with underlining added to show identical language. As the table shows, claim 16 is not meaningfully different from claim 3 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 3. It therefore would have been obvious for the reasons explained for claim 3.

Claim 3	Claim 16
3. The system of claim 1, <u>wherein the first context stack includes a plurality of context entries,</u> and wherein the one or more physical processors are further caused to:	16. The method of claim 14, <u>wherein the first context stack includes a plurality of context entries,</u> the method further comprising:
<u>identify,</u> <u>from among the plurality of context entries, one or more context entries that correspond to the one or more words,</u> <u>wherein the updated context information includes the one or more context entries.</u>	<u>identifying, by the one or more physical processors,</u> <u>from among the plurality of context entries, one or more context entries that correspond to the one or more words,</u> <u>wherein the updated context information includes the one or more context entries.</u>

13. Claim 17

123. The table below provides a side-by-side comparison of claim 4 and claim 17, with underlining added to show identical language. As the table shows, claim 17 is not meaningfully different from claim 4 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 4. It therefore would have been obvious for the reasons explained for claim 4.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 4	Claim 17
4. The system of claim 3, <u>wherein identifying the one or more context entries comprises:</u>	17. The method of claim 16, <u>wherein identifying the one or more context entries comprises:</u>
<u>comparing the plurality of context entries with the one or more words;</u>	<u>comparing the plurality of context entries with the one or more words;</u>
<u>generating one or more rank scores for individual context entries of the plurality of context entries based on the comparison; and</u>	<u>generating one or more rank scores for individual context entries of the plurality of context entries based on the comparison; and</u>
<u>identifying, from among the plurality of context entries, the one or more context entries based on the one or more rank scores.</u>	<u>identifying, from among the plurality of context entries, the one or more context entries based on the one or more rank scores.</u>

14. Claim 18

124. The table below provides a side-by-side comparison of claim 8 and claim 18, with underlining added to show identical language. As the table shows, claim 20 is not meaningfully different from claim 8 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 8. It therefore would have been obvious for the reasons explained for claim 8.

Claim 8	Claim 18
8. The system of claim 3, wherein the one or more physical processors are further caused to:	18. The method of claim 16, further comprising:
<u>identify</u>	<u>identifying</u> , by the one or more physical processors,

Claim 8	Claim 18
<u>one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and</u>	<u>one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and</u>
<u>generate</u>	<u>generating, by the one or more physical processors,</u>
<u>a response to the command or request using the one or more domain agents.</u>	<u>a response to the command or request using the one or more domain agents.</u>

15. Independent Claim 20

125. The table below provides a side-by-side comparison of claim 1 and claim 20, with underlining added to show identical language. As the table shows, claim 20 is not meaningfully different from claim 1 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 1. While claim 20 recites a “**context set**” instead of a “**context stack**,” this is not a meaningful difference, as a person of ordinary skill in the art would have understood that a “set” is generally more broadly understood as a collection of information and therefore that a “context stack” is a type of “context set.” Claim 20 therefore would have been obvious for the reasons explained for claim 1.

Claim 1	Claim 20
1. A system for <u>processing natural language utterances where recognized words of the natural language utterances</u>	20. A computer-implemented method of <u>processing natural language utterances where recognized words of the natural</u>

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 1	Claim 20
<u>alone are insufficient to completely determine one or more commands or requests</u> , the system comprising:	<u>language utterances alone are insufficient to completely determine one or more commands or requests</u> ,
<u>one or more physical processors</u> programmed with <u>one or more computer program instructions which, when executed</u> , cause the one or more physical processors to:	the method being implemented by a computer system that includes <u>one or more physical processors</u> executing <u>one or more computer program instructions which, when executed</u> , perform the method, the method comprising:
<u>generate</u>	<u>generating</u> , by the one or more physical processors,
<u>a first context stack associated with a first device, the first context stack comprising context information that corresponds to a plurality of prior utterances;</u>	<u>a first context set associated with a first device, the first context set comprising context information that corresponds to a plurality of prior utterances;</u>
<u>synchronize the first context stack with a second context stack associated with a second device such that the context information of the first context stack is updated based on related context information of the second context stack;</u>	<u>synchronizing the first context set with a second context set associated with a second device such that the context information of the first context set is updated based on related context information of the second context set;</u>
<u>receive</u>	<u>receiving</u> , at the one or more physical processors,
<u>a natural language utterance associated with a command or request;</u>	<u>a natural language utterance associated with a command or request;</u>
<u>determine</u>	<u>determining</u> , by the one or more physical processors,
<u>one or more words of the natural language utterance by performing</u>	<u>one or more words of the natural language utterance by performing</u>

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 1	Claim 20
<u>speech recognition on the natural language utterance; and</u>	<u>speech recognition on the natural language utterance; and</u>
<u>determine</u>	<u>determining</u> , by the one or more physical processors,
<u>the command or request based on the one or more words and the updated context information.</u>	<u>the command or request based on the one or more words and the updated context information.</u>

16. Claim 21

126. The table below provides a side-by-side comparison of claim 2 and claim 21, with underlining added to show identical language. As the table shows, claim 21 is not meaningfully different from claim 2 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 2. It therefore would have been obvious for the reasons explained for claim 2.

Claim 2	Claim 21
2. The system of claim 1, wherein the one or more physical processors are further caused to:	21. The method of claim 20, further comprising:
<u>prompt</u>	<u>prompting</u> , by the one or more physical processors,
<u>a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and</u>	<u>a user associated with the natural language utterance for one or more of (i) additional information regarding the command or request or (ii) confirmation regarding the command or request; and</u>

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 2	Claim 21
<u>receive</u>	<u>receiving</u> , at the one or more physical processors,
<u>a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.</u>	<u>a non-speech input regarding one or more of the additional information or the confirmation in response to the prompt, wherein the command or request is determined further based on the non-speech input.</u>

17. Claim 22

127. The table below provides a side-by-side comparison of claim 3 and claim 22, with underlining added to show identical language. As the table shows, claim 22 is not meaningfully different from claim 3 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 3. It therefore would have been obvious for the reasons explained for claim 3.

Claim 3	Claim 22
3. The system of claim 1, <u>wherein the first context stack includes a plurality of context entries</u> , and wherein the one or more physical processors are further caused to:	22. The method of claim 20, <u>wherein the first context set includes a plurality of context entries</u> , the method further comprising:
<u>identify</u> ,	<u>identifying</u> , by the one or more physical processors,
<u>from among the plurality of context entries, one or more context entries that correspond to the one or more words,</u>	<u>from among the plurality of context entries, one or more context entries that correspond to the one or more words,</u>

Claim 3	Claim 22
<u>wherein the updated context information includes the one or more context entries.</u>	<u>wherein the updated context information includes the one or more context entries.</u>

18. Claim 23

128. The table below provides a side-by-side comparison of claim 7 and claim 23, with underlining added to show identical language. As the table shows, claim 23 is not meaningfully different from claim 7 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 7. It therefore would have been obvious for the reasons explained for claim 7.

Claim 7	Claim 23
7. The system of claim 3, <u>wherein identifying the one or more context entries comprises identifying, from among the plurality of context entries, the one or more context entries that most closely correspond to the one or more words.</u>	23. The method of claim 22, <u>wherein identifying the one or more context entries comprises identifying, from among the plurality of context entries, the one or more context entries that most closely correspond to the one or more words.</u>

19. Claim 24

129. The table below provides a side-by-side comparison of claim 8 and claim 24, with underlining added to show identical language. As the table shows, claim 24 is not meaningfully different from claim 8 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 24. It therefore would have been obvious for the reasons explained for claim 8.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 8	Claim 24
8. The system of claim 3, wherein the one or more physical processors are further caused to:	24. The method of claim 22, further comprising:
<u>identify</u>	<u>identifying</u> , by the one or more physical processors,
<u>one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and</u>	<u>one or more domain agents associated with the one or more context entries, wherein the one or more domain agents are configured to process the command or request; and</u>
<u>generate</u>	<u>generating</u> , by the one or more physical processors,
<u>a response to the command or request using the one or more domain agents.</u>	<u>a response to the command or request using the one or more domain agents.</u>

20. Claim 25

130. The table below provides a side-by-side comparison of claim 9 and claim 25, with underlining added to show identical language. As the table shows, claim 25 is not meaningfully different from claim 9 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 25. It therefore would have been obvious for the reasons explained for claim 9.

Claim 9	Claim 25
9. The system of claim 8, <u>wherein each of the one or more domain agents comprises domain knowledge associated with a particular domain, the</u>	The method of claim 24, <u>wherein each of the one or more domain agents comprises domain knowledge associated with a particular domain, the</u>

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 9	Claim 25
<u>domain knowledge comprising: one or more of (i) a keyword; (ii) a link to an information source; (iii) a list of responses associated with a plurality commands or requests; (iv) a substitution list used to format the plurality of commands or requests; or (v) content including dictionaries, encyclopedias, or almanacs.</u>	<u>domain knowledge comprising: one or more of (i) a keyword; (ii) a link to an information source; (iii) a list of responses associated with a plurality commands or requests; (iv) a substitution list used to format the plurality of commands or requests; or (v) content including dictionaries, encyclopedias, or almanacs.</u>

21. Claim 26

131. The table below provides a side-by-side comparison of claim 10 and claim 26, with underlining added to show identical language. As the table shows, claim 26 is not meaningfully different from claim 10 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 10. It therefore would have been obvious for the reasons explained for claim 10.

Claim 10	Claim 26
10. The system of claim 8, wherein <u>the response comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.</u>	26. The method of claim 24, wherein <u>the response comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.</u>

C. Ground 2: Claims 10-13, 19 and 27 Are Obvious Over Coffman and Kennewick.

1. Claim 10: “The system of claim 8, wherein the response comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.”

132. The additional limitation of claim 10—“**wherein the response comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request**”—would have been obvious over Coffman alone or in view of **Kennewick**.

133. As discussed above for claim 8, Coffman describes how a conversationally-aware application receives and processes a request and provides a response. Coffman also describes, in connection with processing a request, collecting different types of information or gathering information from multiple sources. For example, Coffman describes a name dialer application on a smartphone that can use both information stored locally on the smartphone as well as information stored on a network machine. (Coffman, 45:2-6 (“For example, a name dialer application can be added to a conversational smart phone. The names that are often used are stored locally and recognized. On the other hand, unknown names or names that were never used before are sent to a more powerful networked machine for recognition and then download the updated information (phone number to dial

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

etc.).”).) As another example, Coffman describes an in-vehicle navigation functionality where location information can be retrieved from local or remote sources and combined with traffic information in order to provide turn by turn directions to the user. (Coffman, 57:5-7 (“You now instruct the PVA to guide you turn by turn to the store. Your position is computed, the location of the store is fetched, locally or from a server, and an itinerary is computed to take into account the latest traffic information.”).) Coffman’s in-vehicle navigation functionality, where different types of information (e.g., location and traffic information) are gathered in order to provide a response in the form of turn by turn directions, is an example of a **“response [that] comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.”**

134. Claim 10 would have also been obvious in view of U.S. Patent Application Publication No. 2004/0044516 A1 (**Kennewick** [EX1004]). Kennewick shares a number of named inventors with the ’652 patent, and the ’652 patent appears to incorporate a substantial amount of Kennewick’s disclosures. Like Coffman and the ’652 patent, Kennewick discloses a speech-based interface for interacting with and directing computer systems and software. (Kennewick, ¶0003.) With respect to claim 10, Kennewick similarly discloses domain agents and that

“queries are...sent to the designated agent[s].” (Kennewick, ¶0186.) Kennewick further explains that “[t]he agents may then, in turn, generate their own queries and/or commands to be sent to, for example, local or remote information sources to retrieve needed data.” (Kennewick, ¶0186.) The queries are executed, and the domain agents return the results, which may have been based on information obtained from several sources. (Kennewick, ¶0186 (“As a result of the execution of the queries, results may be returned by the domain agents and/or sources. The system 90 may then extract or scrape the desired information from the one or more results, which may be returned in any one of a number of different formats at 414. That is, the results sought by a user may be the summary or the results of further processing of information obtained from several sources for example.”).) Kennewick further describes how the results can be “gather[ed] and combin[ed] into a single best result” and “the compound results [] presented through the text to speech engine 124, to the user in a useful and expected manner at 420.” (Kennewick, ¶0186.) Kennewick therefore discloses generating a response that “**comprises an aggregation of one or more responses generated by the one or more domain agents based on the command or request.**”

135. **Rationale and Motivation to Combine (Coffman and Kennewick).**

It would have been obvious to a person of ordinary skill in the art to combine

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Coffman and Kennewick, predictably resulting in using a conversationally aware application (“**domain agent**”) in Coffman’s conversational system to generate a response “**compris[ing] an aggregation of one or more responses generated by**” a conversationally aware application (“**the one or more domain agents**”) “**based on the command or request.**”

136. Both Coffman and Kennewick are analogous references in the same field as the ’652 patent. (’652, 1:22-25 (“The invention provides mobile devices having a speech interface and/or a combination of a speech interface and a non-speech interface to enable natural language human-machine interactions.”); Coffman, e.g., 1:7-8 (“systems and methods for conversational computing”), Figs. 1, 4 & 6; Kennewick, ¶0003 (“The present invention relates to the retrieval of online information or processing of commands through a speech interface. More specifically, the invention is a fully integrated environment allowing users to submit natural language speech questions and commands.”).) Coffman and Kennewick would have also been reasonably pertinent to a number of problems facing the patent inventors, including how to use application(s) to generate responses based on a user’s command or request—as noted, like the ’652 patent, both Coffman and Kennewick describe agent-based systems. (Coffman, e.g., 12:28-30, 57:23-27; Kennewick, e.g., ¶¶0088-0094.)

137. A person of ordinary skill in the art would have been motivated to combine Coffman and Kennewick. As discussed above, Coffman and Kennewick disclose similar systems for controlling a number of applications via a speech-based interface. A person of ordinary skill in the art would have been motivated to incorporate Kennewick's teachings so that the response generated for the user would have included an aggregation of information gathered when Coffman's conversationally-aware application processed the request. A person of ordinary skill in the art would have readily appreciated the benefits of generating such a response, and that those benefits would have been applicable to a wide range of applications. For example, a person of ordinary skill in the art would have recognized that responding to a query often requires gathering different pieces of information from different sources—such as the in-vehicle navigation scenario that Coffman describes. As such, a person of ordinary skill in the art would have recognized that aggregating information would allow the system to provide more complete and useful responses to the user.

138. A person of ordinary skill in the art would have had a reasonable expectation of success. Incorporating Kennewick's teachings into Coffman's conversational system would have involved the straightforward application of Kennewick's technique to improve Coffman's similar agent-based conversational

computing system in the same way. A person of ordinary skill in the art would have found the disclosures of Coffman and Kennewick sufficiently detailed to implement the techniques on which I rely, without undue experimentation.

2. Claim 11: “The system of claim 8, wherein the one or more physical processors are further caused to:”

139. As explained below, the additional steps recited in claim 11 would have been obvious over Coffman in view of Kennewick. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the processor(s) of Coffman’s master/central device would execute software (computer program instructions) to perform the following steps.

(a) “obtain information relating to a license agreement that is associated with at least one of the one or more domain agents;” (11[a])

140. Coffman discloses that its conversational computing system includes help and support services providing functionality to install, upgrade and re-install conversationally-aware applications (“**domain agents**”):

In yet another aspect, conversational help, manuals and support is provided through a ubiquitous coordinated conversational interface, using local and remote resources, usage history of a user and agents to complete request, guide through procedure, search for information and upgrade/install new applications.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

(Coffman, 9:4-7; *see also id.*, 52:7-11 (“Different degrees and modalities of help are provided....It can take the form of conversational technical support involving local or remote agents (e.g. to upgrade or re-install and application in the background).”); 21:15-18 (“It is to be appreciated that services such as conversational help and assistance, as well as some dialog prompts (introduction, questions, feedback etc) provided by the CVM system...”).) Coffman also describes providing the ability for parties to develop conversationally aware applications for the CVM system, explaining that “development tools are provided for developer to build, simulate and debug conversational aware application for CVM.” (Coffman, 7:29-30; *see also id.*, 18:16-18 (“The development environment offered by the CVM is referred to herein as SPOKEN AGE™. Spoken Age allows a developer to build, simulate and debug conversational aware application for CVM.”), 13:21-23 (“[T]he conversational programming languages/scripts provide the conversational APIs that allow an application developer to hook (or develop) conversationally aware applications 200”).)

141. Software licenses were commonly known at the time and even casual computer users would recall being prompted to click “I agree” to an end-user license agreement presented during software installation. Coffman does not detail licensing-related issues associated with installing and using conversationally-aware

applications, but the claimed step of “**obtain[ing] information relating to a license agreement that is associated with**” a conversationally-aware application (“**at least one of the one or more domain agents**”) would have been obvious in view of Kennewick. With respect to claim 11, Kennewick describes an “agent manager” that verifies that software use is within allowed license terms and conditions, and if so, allows the software to be used:

The agent manager **154** may provide a license management client capable of executing most any license terms and conditions. When a particular agent **106** and/or database element **102** is required by a command, the agent manager [sic] **154** verifies that the use of the agent or data element is within the allowed terms and conditions, and if so, invokes the agent or allows access to the data element. License management schemes that can be implemented through the agent manager **154** include outright purchase, subscription for updates, one time or limited time use. Use of shared agents and data elements (such as those down-loaded from web sites maintained by groups of domain experts) may also be managed by the agent manager **154**.

(Kennewick, ¶0104.) These disclosures are nearly identical to those in the ’652 patent. (*Compare* Kennewick, ¶0104 with ’652, 28:63-29:9.)

142. I discussed the rationale and motivation to combine Coffman and Kennewick in connection with my analysis of claim 10, which applies here as well. With respect to claim 11, combining Coffman and Kennewick would have

predictably resulted in Coffman's conversational system having functionality to verify that use of conversationally-aware applications was within allowed license terms and conditions. Combining the teachings of Coffman and Kennewick in this manner would have therefore resulted in **"obtain[ing] information relating to a license agreement that is associated with"** a conversationally-aware application (**"at least one of the one or more domain agents"**), as claimed. A person of ordinary skill in the art would have been motivated to combine. As discussed above, Coffman and Kennewick disclose similar systems for controlling a number of applications via a speech-based interface. As is commonly known, software application developers often make their software available pursuant to license terms and conditions, and expect that a user of their software will not violate those terms and conditions. Application developers have various motivations for terms and conditions, including revenue generation. If the terms and conditions cannot be adequately enforced, the application developer may opt to not develop the application, or to not develop the application for a particular platform. A person of ordinary skill in the art would have readily appreciated that this would have the effect of both making the application unavailable, as well as weakening the overall value of the platform by reducing the size of its application ecosystem. A person of ordinary skill in the art would have therefore been motivated by these market-forces-

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

related issues to include license management functionality as taught by Kennewick as part of Coffman's conversational system and thereby increase the value and usability of the system to its users.

143. A person of ordinary skill in the art would have had a reasonable expectation of success. The combination mostly relates to non-technical licensing issues and would have involved the straightforward application of conventional techniques and technologies to, for example, retrieve license information for a conversationally-aware application (such as license expiration or limitation to particular machine(s)) and compare it to the circumstances of the application's use. A person of ordinary skill in the art would have found the disclosures of Coffman and Kennewick sufficiently detailed to implement the techniques on which I rely, without undue experimentation.

(b) “determine, based on the information related to the license agreement, that use of the at least one of the one or more domain agents to process the command or request is permitted; and” (11[b])

144. I addressed 11[b] in my analysis of 11[a] above. As discussed, Kennewick discloses “agent manager” functionality that verifies that software use is within allowed license terms and conditions, and if so, allows the software to be used. (Kennewick, ¶0104.) And as discussed, it would have been obvious to combine Kennewick's teachings regarding license management with Coffman. The

combination would have predictably resulted in Coffman's conversational system having functionality to **"determine, based on the information related to the license agreement, that use of"** the conversationally-aware application (**"the at least one of the one or more domain agents"**) **"to process the command or request is permitted."**

(c) **"use, based on the determination that the use is permitted, the at least one of the one or more domain agents to generate the response."** (11[c])

145. I also addressed 11[c] in my analysis of 11[a] above. The combination of Coffman and Kennewick in the manner described would have predictably resulted in Coffman's conversational system having functionality to **"use, based on the determination that the use is permitted,"** the conversationally-aware application (**"the at least one of the one or more domain agents"**) **"to generate the response."**

3. **Claim 12: "The system of claim 1, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:"** (12[pre])

146. As discussed for 2[a], Coffman discloses that the natural language utterance is **"associated with a user"** because the system receives spoken natural language input from a person in connection with using the system; and moreover, Coffman discloses identifying the user who spoke the utterance, thereby associating the utterance with a user. (Coffman, e.g., 5:1-4, 29:1-4, 47:25-27, 8:16-18.)

147. For substantially the same reasons as those explained for 1[a], a person of ordinary skill in the art would have understood and found it obvious that the system's processor(s) would execute software (computer program instructions) to perform the following steps.

- (a) **“obtain one or more of a cognitive model or an environmental model associated with the user, wherein the cognitive model comprises information relating to one or more interactions between the user and the system, and the environmental model comprises information indicative of how noisy an environment surrounding the user is,” (12[a])**

148. With respect to a **“cognitive model,”** the '652 patent describes it as information derived from a user's interactions with the system that may be used in determining a user's future actions by “assisting with speech recognition and/or question or command recognition”:

According to one embodiment of the invention, personalized cognitive model **810** is a model derived from a user's interaction pattern with the system and may be used to predict what actions the user may take next in time, thus assisting with speech recognition and/or question or command recognition. Personalized cognitive model **810** may track actions performed by the user. When the system is attempting to predict user behavior, the personalized cognitive model may be consulted first. The system may have multiple personalized cognitive models, wherein one may correspond to each user.

(’652, 24:1-11.)

149. Turning back to Coffman, it describes collecting information about the user, such as preferences, security habits, biometrics and behavior. (Coffman, 33:1-5 (“The CVM 401 further comprises a meta information manager 403 which manages elements such as...information about the user (preferences, security habits, biometrics, behavior, etc.)”).) Coffman teaches that the collected information can be used “to perform more robust recognition” and to “recognize the user.” (Coffman, 37:5-10.) Coffman further teaches that the collected information can include “**information relating to one or more interactions between the user and the system,**” such as “the usual phrasing of a query, the type of query phrased, command frequency (often used, not often used), preferred applications, time or usage, etc.” (Coffman, 37:8-10.) Coffman therefore discloses a “**cognitive model**” comprising information collected about a user during their interactions with the device that the system may use in connection with recognizing words in the user’s natural language utterance. The cognitive model is “**obtained**” by collecting information about the user, as well as when the collected information is accessed in connection with performing recognition.

150. **Kennewick** further confirms and renders obvious the claimed “**cognitive model.**” Like Coffman, Kennewick similarly discloses that its “system

may create, store and use extensive personal profile information for each user.” (Kennewick, ¶0029.) The personal profile information includes “**information relating to one or more interactions between the user and the system,**” such as “the history of [the user’s] interactions with the system,” “their special use of terminology,” “history of questions asked,” and “session histories.” (Kennewick, ¶¶0114 (“Personal profiles may include information specific to the individual, their interests, their special use of terminology, the history of their interactions with the system, and domains of interest.”), 0029 (“Information in the profile may be added and updated automatically as the user uses the system or may be manually added or updated by the user or by others....Examples of information in a user profile includes, history of questions asked, session histories, formatting and presentation preferences, special word spelling, terms of interest, special data sources of interest, age, sex, education, location or address, place of business, type of business, investments, hobbies, sports interests, news interests and other profile data.”).)

151. Kennewick further discloses that the collected user profile information may be used in connection with speech recognition, stating that “[t]he personal profile data may be used by the agents 106, the speech recognition engine 120, the text to speech engine 124, and the parser 118.” (Kennewick, ¶0114.) Kennewick explains, for example, that “the system 90 may employ a speech recognition engine

124 seeding for improved word recognition accuracy, using data from the dictionary and phrase tables 112, user profiles 110, and the agents 106.” (Kennewick, ¶0149.) To “maximize the probability of correct recognition,” word probabilities “may be dynamically updated based on a number of criteria including the application domain, the questions or commands, contexts, the user profile and preferences, user dialog history, the recognizer dictionary and phrase tables, and word spellings.” (Kennewick, ¶0149.) For example, Kennewick states that “when a user uses a word that is not recognized at all or is not correctly recognized by the speech recognition engine 120 then the user may be asked to spell the word,” and the pronunciation for the word can then be added to the user’s profile. (Kennewick, ¶0151.) “Through this process the speech recognition engines learns with time and improves accuracy.” (Kennewick, ¶0151; *see also id.* (“The word is looked up in the dictionary 112 and the pronunciation for the word is added to either the dictionary, the agent 106, or the user’s profile 110. The word pronunciation can then be associated with the domain, the question, the context and the user. Though this process the speech recognition engine learns with time and improves accuracy.”).)

152. The rationale and motivation to combine Coffman and Kennewick discussed in connection with claim 10 applies here as well. In addition, with respect to claim 12, a person of ordinary skill in the art would have been motivated to

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

combine Coffman's conversational system with Kennewick's teachings providing additional detail about collected profile information and using the profile information to provide improved speech recognition. Because the user profile information includes information associated with a user's prior interactions with the system—such as dialog history and word spellings—the combination would have predictably resulted in **“obtain[ing] one or more of a cognitive model [or an environmental model associated with the user], wherein the cognitive model comprises information relating to one or more interactions between the user and the system** [, and the environmental model comprises information indicative of how noisy an environment surrounding the user is]” as claimed.

153. Coffman and Kennewick describe collecting similar information about the user, such as dialog history. (Kennewick, ¶¶0149-0151; Coffman, 37:8-10 (“the usual phrasing of a query, the type of query phrased, command frequency (often used, not often used), preferred applications, time or usage, etc.”).) And as noted, Coffman already discloses using collected user information to “perform more robust recognition” and to “recognize the user.” Kennewick provides express motivation to combine, explaining that the user profile information can be used to improve speech recognition accuracy. (Kennewick, ¶¶0149 (“[T]he system **90** may employ a speech recognition engine **124** seeding for improved word recognition accuracy,

using data from the dictionary and phrase tables **112**, user profiles 110, and the agents **106.**”), 0151 (“Through this process the speech recognition engines learns with time and improves accuracy.”).) A person of ordinary skill in the art therefore would have been motivated to incorporate Kennewick’s techniques into Coffman’s similar system to obtain the same benefits and advantages.

154. A person of ordinary skill in the art would have had a reasonable expectation of success. The combination would have involved the application of known technologies and techniques. Kennewick, for example, acknowledges that “[s]peech recognition...today is successfully used in a wide range of applications.” (Kennewick, ¶0006.) Collecting and using information about a user’s interactions with a speech-based system in order to improve speech recognition was also well known. (EX1008, p.371 (entry for “adaptive language model” describing adapting language model probabilities to the most recent history).) And because I understand that a patent need not teach, and preferably omits what is well known in the art, the passing mention that the ’652 patent, Coffman and Kennewick all make to using user profile information to improve recognition provides further confirmation that doing so was well within the knowledge of a person of ordinary skill in the art. A person of ordinary skill in the art would have found the disclosures of Coffman and Kennewick sufficiently detailed to implement the techniques on which I rely,

without undue experimentation.

(b) “wherein the one or more words are determined further based on one or more of the cognitive model or the environmental model.” (12[b])

155. As discussed for 12[a], both Coffman and Kennewick disclose that collected user profile information (“**cognitive model**”) is used in performing speech recognition, i.e., determining one or more words of the natural language utterance. (E.g., Coffman, 37:5-10 (“It is to be appreciated that conversational biometrics can be used to collect any context and meta information on the user not only to customize or adapt for purposes of user preferences or to authorize a query, but also to use the information to perform more robust recognition. Accordingly, any information can be accumulated to recognize the user. Namely, the usual phrasing of a query, the type of query phrased, command frequency (often used, not often used), preferred applications, time or usage, etc.”); Kennewick, ¶¶0149 (“[T]he system **90** may employ a speech recognition engine **124** seeding for improved word recognition accuracy, using data from the dictionary and phrase tables **112**, user profiles **110**, and the agents **106**. At the same time, the fuzzy set possibilities or prior probabilities for the words in the dictionary and phrase tables may be dynamically updated to maximize the probability of correct recognition at each stage of the dialog. The probabilities or possibilities may be dynamically updated based on a number of

criteria including the application domain, the questions or commands, contexts, the user profile and preferences, user dialog history, the recognizer dictionary and phrase tables, and word spellings.”), 0151.) 12[b] is therefore obvious over Coffman in view of Kennewick.

4. Claim 13: “The system of claim 1, wherein the natural language utterance is associated with a user, and wherein the one or more physical processors are further caused to:” (13[pre])

156. As discussed for 2[a] (and 12[pre]), Coffman discloses that the natural language utterance is “**associated with a user**” because the system receives spoken natural language input from a person in connection with using the system; and moreover, Coffman discloses identifying the user who spoke the utterance, thereby associating the utterance with a user. (Coffman, e.g., 5:1-4, 29:1-4, 47:25-27, 8:16-18.)

(a) “obtain a first cognitive model that comprises information relating to one or more interactions between the user and the system; and” (13[a])

157. For the same reasons explained for 12[a], Coffman and Kennewick disclose and render obvious the claimed “**first cognitive model**.”

(b) **“obtain a second cognitive model that comprises information relating to one or more interactions between the system and a plurality of users of the system,” (13[b])**

158. As discussed above for 12[a], the written description of the '652 patent describes a “personalized cognitive model **810**.” ('652, 24:1-11.) In addition to that model, the '652 patent also describes a second, “general cognitive model **806**” that “corresponds to interaction patterns with the system for multiple users”:

According to another embodiment of the invention, general cognitive model **806** is a statistical abstract that corresponds to interaction patterns with the system for multiple users. Data stored within general cognitive model **806** may be used to predict a user's next action, thus assisting with speech recognition and/or question or command recognition. The general cognitive model **806** may also track what actions a particular user has performed and may be used when the user interacts with the system in a way that is not handled in the personalized cognitive model.

('652, 24:12-21.)

159. Kennewick discloses the claimed “**second cognitive model**” in the form of profile information about other users. This is because Kennewick describes how its system “may employ a speech recognition engine **124** seeding for improved word recognition accuracy, using data from...user profiles **110**.” (Kennewick, ¶0149.) As discussed above for claim 12[a], the collected user profile information

includes prior interactions with the system, such as “the history of [the user’s] interactions with the system,” “their special use of terminology,” “history of questions asked,” “session histories” and spellings and pronunciations provided by users. (Kennewick, ¶¶0114, 0029, 0151.) A person of ordinary skill in the art would have understood and found it obvious that these interactions would have been **“between the system and a plurality of users of the system”** because Kennewick explains that its system supports multiple users in a given session—e.g., multiple users of the same device. (Kennewick, ¶0045 (“Multiple users may use the system at different times or during interleaved or overlapping sessions.”), 0147 (“multiple users engage in overlapping sessions”), 0139 (describing microphone “steered to point in the direction of the one or more users speaking”).) Coffman describes multi-user sessions as well. (Coffman, 47:25-26, 47:27-48:7, 29:22-23.) Like the first cognitive model, the second cognitive model is **“obtained”** by collecting information about the users, as well as when the collected information is accessed in connection with performing recognition.

160. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis for 12[a] and applies here as well. With respect to claim 13[b], it would have further been obvious to incorporate Kennewick’s teachings regarding using other users’ profile information (**“second cognitive model”**) in

connection with speech recognition. Kennewick provides express motivation to combine, explaining that user profile information can be used to improve word recognition accuracy. (Kennewick, ¶0149.) For example, if a user added a new word and pronunciation to their profile, including that new word for other users could improve the speech recognition accuracy of the other users' spoken input as well. (Kennewick, ¶0151 (speech recognition engine learns with time and improves accuracy based on user-provided spellings).) Similarly, past questions input by other users could help the system recognize likely input from the user. (Kennewick, ¶0149 (describing updating word probabilities to maximize the probability of correct recognition based on, among other things, questions or commands, the user profile and preferences, word spellings, and user dialog history).) A person of ordinary skill in the art would have appreciated that applying Kennewick's teachings to Coffman would improve a similar system in the same way, and there would have found it at least obvious to try.

161. In **Ground 3**, I discuss how speech recognition based on a “**second cognitive model**” would also have been obvious in further view of **Lee** (EX1006).

(c) “**wherein the one or more words are determined further based on the first cognitive model and the second cognitive model.**” (13[c])

162. As discussed for 12[a] (for the “**first cognitive model**”) and 13[b] (for

the “**second cognitive model**”), Kennewick discloses that collected user profile information (“**first cognitive model**” and “**second cognitive model**”) are used in performing speech recognition, i.e., determining one or more words of the natural language utterance. (Kennewick, *e.g.*, ¶¶0149, 0151.) 13[c] is therefore obvious over Coffman in view of Kennewick.

5. Claim 19

163. The table below provides a side-by-side comparison of claim 13 and claim 19, with underlining added to show identical language. As the table shows, claim 19 is not meaningfully different from claim 13 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 13. It therefore would have been obvious for the reasons explained for claim 13.

Claim 13	Claim 19
13. The system of claim 1, <u>wherein the natural language utterance is associated with a user</u> , and wherein the one or more physical processors are further caused to:	19. The method of claim 14, <u>wherein the natural language utterance is associated with a user</u> , the method further comprising:
<u>obtain</u>	<u>obtaining</u> , by the one or more physical processors,
<u>a first cognitive model that comprises information relating to one or more interactions between the user and the system; and</u>	<u>a first cognitive model that comprises information relating to one or more interactions between the user and the system; and</u>
<u>obtain</u>	<u>obtaining</u> , by the one or more physical processors,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 13	Claim 19
<u>a second cognitive model that comprises information relating to one or more interactions between the system and a plurality of users of the system,</u>	<u>a second cognitive model that comprises information relating to one or more interactions between the system and a plurality of users of the system,</u>
<u>wherein the one or more words are determined further based on the first cognitive model and the second cognitive model.</u>	<u>wherein the one or more words are determined further based on the first cognitive model and the second cognitive model.</u>

6. Claim 27

164. The table below provides a side-by-side comparison of claim 11 and claim 27, with underlining added to show identical language. As the table shows, claim 27 is not meaningfully different from claim 11 for purposes of my analysis, reciting the “computer-implemented method” counterpart to the “system” of claim 11. It therefore would have been obvious for the reasons explained for claim 11.

Claim 11	Claim 27
11. The system of claim 8, wherein the one or more physical processors are further caused to:	27. The method of claim 24, further comprising:
<u>obtain</u>	<u>obtaining</u> , by the one or more physical processors,
<u>information relating to a license agreement that is associated with at least one of the one or more domain agents;</u>	<u>information relating to a license agreement that is associated with at least one of the one or more domain agents;</u>
<u>determine</u> ,	<u>determining</u> , by the one or more physical processors,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

Claim 11	Claim 27
<u>based on the information related to the license agreement, that use of the at least one of the one or more domain agents to process the command or request is permitted; and</u>	<u>based on the information related to the license agreement, that use of the at least one of the one or more domain agents to process the command or request is permitted; and</u>
<u>use,</u>	<u>using, by the one or more physical processors,</u>
<u>based on the determination that the use is permitted, the at least one of the one or more domain agents to generate the response.</u>	<u>based on the determination that the use is permitted, the at least one of the one or more domain agents to generate the response.</u>

D. Ground 3: Claims 13 and 19 Are Obvious Over Coffman, Kennewick and Lee.

165. As discussed above in my Ground 2 analysis, claims 13 and 19 recite using a “**second cognitive model**” in performing speech recognition, i.e., determining one or more words of the user’s natural language utterance. I explained that Kennewick discloses the claimed “**second cognitive model**” in the form of profile information about other users, which includes users’ prior interactions with the system, such as “history of questions asked” and “session histories.” (Kennewick, ¶¶0114, 0029, 0149, 0151.) I further explained that using other users’ profile information when performing speech recognition could help the system recognized likely input from the user. (Kennewick, ¶0149.)

166. Lee offers more detail and provides confirmation that it would have

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

been obvious to use other users' profile information involving users' interactions with the device, such as prior questions and commands, in connection with performing speech recognition. Lee, U.S. Patent Application Publication No. 2002/0087315, entitled "Computer-Implemented Multi-Scanning Language Method and System," describes techniques for assisting accurate speech recognition of a user by using not only information collected about that user but also information collected about other users. (Lee, Abstract, ¶¶0002, 0016, 0022 (describing using user's dialogue history), Fig. 3 (showing dialog history 148 and popularity engine 144).) Lee describes a "multi-scan control unit 32" that may use "recognition assisting databases 48" to supplement recognition of speech input, including a database that "maintains a personalized profile that have already been recognized for a particular user or for users who have previously submitted requests which are similar to the request at hand." (Lee, ¶0016.) Lee also describes a "popularity engine 144" that "contains data compiled from multiple users' histories that has been calculated for the prediction of likely user requests." (Lee, ¶0024.) Lee explains that "[t]his increases the accuracy of word recognition"; for example, with this information, the system can "scan[] for the most probable match for words in the user utterance based on past requests." (Lee, ¶¶0024, 0026.)

167. **Rationale and Motivation to Combine (Lee with Coffman and**

Kennewick). It would have been obvious to a person of ordinary skill in the art to combine Lee's teachings regarding using information collected about multiple users to improve speech recognition accuracy with the conversational system as taught by the combination of Coffman and Kennewick described in connection with Ground 2. The combination would have predictably resulted in Coffman's conversational system using information on one or more prior interactions between a device and a plurality of users when performing speech recognition. As discussed above, both Coffman and Kennewick already disclose collecting user profile information, including users' prior interactions. (Kennewick, ¶¶0114, 0029, 0149, 0151; Coffman, 33:1-5, 37:-8-10.) Incorporating Lee's teachings to use prior interaction information for improved speech recognition would have therefore resulted in Coffman's conversational system determining one or more words of the user's natural language utterance using a "**second cognitive model**" as claimed.

168. Lee is an analogous reference in the same field of speech recognition and processing as the '652 patent. ('652, 1:22-27; Lee, Abstract, ¶0002.) Lee would have also been reasonably pertinent to problems facing the '652 patent inventors, including techniques for improving speech recognition accuracy, including in multi-user systems.

169. A person of ordinary skill in the art would have been motivated to

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,849,652 B2

combine. As discussed above, Lee, Coffman and Kennewick all describe collecting similar profile information about users, such as dialog history. (Lee, ¶¶0016, 0024, 0026; Kennewick, ¶¶0029, 0114, 0149-0151; Coffman, 33:1-5, 37:8-10.) Kennewick also describes using user profiles for improved word recognition accuracy. (Kennewick, ¶0149.) Lee simply provides further detail and confirmation that using information collected about other users when performing speech recognition can improve speech recognition accuracy. (Lee, ¶¶0016, 0024, 0026.) A person of ordinary skill in the art therefore would have been motivated to incorporate Lee's teachings into Coffman's conversational system to obtain the same benefits and advantages.

170. A person of ordinary skill in the art would have had a reasonable expectation of success. The combination would have involved the application of known technologies and techniques. As I discussed above, Kennewick, for example, acknowledges that "[s]peech recognition...today is successfully used in a wide range of applications." (Kennewick, ¶0006.) As noted, collecting and using information about a user's interactions with a speech-based system in order to improve speech recognition was also well known. (EX1008, p.371 (entry for "adaptive language model" describing adapting language model probabilities to the most recent history).) And the passing mention that the '652 patent, Coffman, Kennewick and

Lee all make to using user profile information to improve recognition provides further confirmation that doing so was well within the knowledge of a person of ordinary skill in the art. A person of ordinary skill in the art would have found the disclosures of Coffman, Kennewick and Lee sufficiently detailed to implement the techniques on which I rely, without undue experimentation.

VI. NO SECONDARY CONSIDERATIONS OF NONOBVIOUSNESS

171. As explained in **Part III.B**, I understand that “secondary considerations” may be used as indicia of nonobviousness, provided that any such consideration has a nexus to the invention claimed. I understand from Petitioner’s counsel that Patent Owner in the related district court litigation has not yet identified any evidence related to secondary considerations with a nexus to the claimed invention. Nor am I aware of information, such as commercial success, unexplained results, long felt but unsolved need, industry acclaim, simultaneous invention, copying by others, skepticism by experts in the field, and failure of others, suggesting that the claims addressed in this Declaration are not obvious. To the extent Patent Owner later provides information it claims relates to secondary considerations, I reserve the right to supplement my analysis and opinions to comment on it.

VII. CONCLUSION

172. In signing this Declaration, I recognize that the Declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in this proceeding. If required, I will appear for cross-examination at the appropriate time. I reserve the right to offer opinions relevant to the invalidity of the challenged claims and/or offer testimony in support of this Declaration.

173. I hereby declare that all statements made herein of my own knowledge are true and that all statements are made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Dated: April 4, 2024

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'Chris Schmandt', written over a horizontal line.

Christopher M. Schmandt
Winchester, MA

Exhibit A

Christopher Schmandt

4 Longfellow
Winchester, MA 01890
+1-617-230-4257

Education

M.I.T., Master of Science, Visual Studies (Computer Graphics), 1980

M.I.T., Bachelor of Science, Computer Science, 1978

Professional Experience -- MIT

Media Laboratory, Principal Research Scientist, 1985-2018 (retired)
Director, Living Mobile Research Group (formerly Speech + Mobility)

Architecture Machine Group, Research Associate, 1980-1984

Architecture Machine Group, Research Assistant, 1979-1980

Architecture Machine Group, Graphics Programmer, 1977-1979

Departmental Undergraduate Research Opportunities Program Coordinator, 1984-2018

Laboratory Intellectual Property Committee 2001-2016, chair 2002-2009

Departmental Committee on Graduate Studies, 1996-2001, 2007-2018

Sponsored Research Activities

Alerting and Mobile Messaging, Digital Life Consortium, MIT Media Lab, 1997-2018

Acoustical Cues to Discourse Structure, National Science Foundation, Principal Investigator, 1995-1998

Parsing Radio, News in the Future Consortium, MIT Media Lab, 1993-1999

Desktop Audio, SUN Microsystems, Inc., Principal Investigator, 1989-1996

Voice Interaction in Hand Held Computers, Apple Computer, Principal Investigator, 1991-1993

Voice Interfaces for Network Services, AT&T, Principal Investigator, 1989-1991

Back Seat Driver, NEC, Principal Investigator, 1988-1991

Acoustic and Visual Cues for Speech Recognition, DARPA, co-Principal Investigator, 1986-1988

Personal Computers and Telephony, NTT Public Corporation, Principal Investigator, 1984-1989

Home Telecomputing, Atari, Inc., Principal Investigator, 1983

Testifying legal engagements within the last five years, representing party in italics

- *Microsoft v. SpeakWare*, 2019
- *Bumble Trading LLC v. Match Group LLC*, 2019
- *Tile Inc. v. Cellwitch Inc.*, 2019
- *Blackberry Ltd v. Facebook, Inc* 2020
- *EROAD Ltd v. PerDiem Co.*, 2020
- *Snap Inc. v. SRK Technology LLC*, 2020
- *Shopify Inc Ltd v. Express Mobile Inc.*, 2020
- *Express Mobile Inc. v Wix Ltd* 2020
- *Motorola Mobility LLC v. Ironworks patents, LLC.* 2021
- *Express Mobile Inc. v GoDaddy Inc* 2021
- *Bumble Trading LLC v. Kinectus, LLC.* 2021
- *Tile, Inc v. Linquet Technologies, Inc.* 2021
- *Facebook, Inc. v. Palo Alto Research Center, Inc.* 2021
- *Unified Patents, LLC v. Gesture Technology Partners, LLC.* 2021
- *Quantum Metric, Inc v. Content Square Isreal, Ltd.* 2021
- *Meta Platforms Inc. v. AlmondNet, Inc* 2022
- *Apple Inc. v. Zentian Limited* 2022
- *Google LLC v. Mira Advanced Technology Systems Inc* 2022
- *Google LLC v. Robocast* 2023
- *Google LLC v. Buffalo Patents LLC* 2023
- *Amazon v. AlmondNet* 2023
- *Google LLC v. Valtrus* 2023
- *Samsung Electronics Co., Ltd v. Stanton Techiya, LLC* 2024
- *Roku, Inc. v. AlmondNet, Inc.* 2024
- *LinkedIn v. AlmondNet, Inc* 2024

Professional Experience -- Intellectual Property

Co-inventor on U.S. Patents

- 5,177,685 for "Automobile navigation system using real time spoken driving instructions"
- 6,728,348 for "System for storing voice recognizable identifiers using a limited input device such as a telephone key pad"
- 6,937,986 for "Automatic dynamic speech recognition vocabulary based on external sources of information"
- 7,098,776 for "Methods and apparatus for vibrotactile communication"
- 7,392,280 for "Method for summarization of threads in electronic mail"
- 7,738,637 "Interactive voice message retrieval"
- 7,443,283 for "Methods and apparatus for connecting an intimate group by exchanging awareness cues and text, voice instant messages, and two-way voice communications"
- 7,865,560 for "System for summarization of threads in electronic mail"
- 8,121,653 for "Methods and apparatus for autonomously managing communications using an intelligent intermediary"
- 8,135,128 for "Animatronic creatures that act as intermediaries between human users and a telephone system"

Co-inventor on U.S. Patent Applications

- 20020076009 "International dialing using spoken commands"
- 20020087328 "Automatic dynamic speech recognition vocabulary based on external sources of information"

- 20020064257 "System for storing voice recognizable identifiers using a limited input device such as a telephone keypad"
- 20030023688 "Voice-based message sorting and retrieval method"
- 20030081738 "Method and apparatus for improving access to numerical information in voice messages"
- 20030144846 "Method and system for modifying the behavior of an application based upon the application's grammar"
- 20030158903 "Calendar bar interface for electronic mail interaction"

Publications

Field Study of a Tactile Sound Awareness Device for Deaf and Hard of Hearing Users ISWC 2020. (with D. Jain, B. Chiu, S. Goodman, L. Findlater, J. Froehlich)

SkinMorph: Texture-Tunable On-Skin Interface Through Thin, Programmable Gel ISWC 2018. (with Cindy Kao, M. Banforth, D. Kim)

Technical Interventions to Detect, Communicate, and Deter Sexual Assault. ISWC 2017. (with Manisha Mohan)

Exploring Interactions and Perceptions of Kinetic Wearables. DIS 2017. (with Cindy Hsin-Liu Kao, D. Ajilo, O. Anilonyte, A. Dementyev, I. Choi and S. Follmer)

Leveraging User-made Predictions to Help Understand Personal Behavior. MobileHCI 2017. (with Miriam Greis, Tilman Dangler and Albrecht Schmidt)

Rovables: On-Body Robots as Mobile Wearables. UIST 2016. (with Cindy Hsin-Liu Kao, A. Dementyev, I. Choi, D. Ajilo, M. Xu, and S. Follmer)

DuoSkin: Rapidly Prototyping On-Skin User Interfaces Using Skin-Friendly Materials. ISWC 2016. (with Cindy Hsin-Liu Kao, Christian Holz, Asta Roseway, and Andres Calvo)

Immersive Terrestrial Scuba Diving Using Virtual Reality (with Dhruv Jain, Misha Sra, Jingru Go, Rodrigo Margues, Raymond Wu and Justin Chiu) Proceedings, UIST 2016

Expanding social mobile games beyond the device screen (with Misha Sra) Journal of Personal and Ubiquitous Computing, 2015

NailO: Fingernails as an input surface (with Cindy Hsin-Liu Kao, Artem Dementyev, Joseph Paradiso) CHI 2015

Mugshots: A mug display for front and back stage social interaction in the workplace (with Cindy Hsin-Liu Kao) TEI (Tangible and Embedded Interfaces) 2015

Mime: compact, low power 3D gesture sensing for interaction with head mounted displays (with Andrea Colaco, Ahmend Kirmani, Hye Soo Yang, Nan-Wei Gong, and Vivek Goyal) Proceedings of UIST 2013.

Spotz: A location-based approach to self-awareness (with Misha Sra) Proceedings of Persuasive 2013.

Setting the stage for interaction: A tablet application to augment group discussion in a seminar class (with Drew Harry and Eric Gordon) Proceedings of CSCW 2012.

Indoor Location Sensing using Geo-Magnetism (with Jaewoo Chung, Matt Donahoe, Ig-Jae Kim, Pedram Razavai and Micaela Wiseman) Proceedings of International Conference on Mobile Systems, Applications, and Services (Mobisys) 2011.

My second bike: a TV-enabled social and interactive riding experience (with Jaewoo Chung, Kuang Xu, Andrea Colaco, and Victor Li) Proceedings of IEEE Communications and Networking Conference, Jan 2010.

Going my way?: User-aware route planner (with Jaewoo Chung) Proceedings CHI 2009.

Globetoddler: Designing for remote interaction between preschoolers and their traveling parents (with Paulina Modlitba) CHI 2008 Extended Abstracts

Are we there yet? - a temporally aware media player (with Matt Adcock and Jaewoo Chung), Australian User Interface Conference (AUIC) 2008

Physical embodiments for mobile communication agents (with Stefan Marti), UIST 2005

Giving the caller the finger: collaborative responsibility for cellphone interruptions (with Stefan Marti) Extended Abstracts, CHI 2005

Active Messenger: email filtering and delivery in a heterogeneous network (with Stefan Marti) Human-Computer Interaction Journal (HCI) Volume 20 (2005)

WatchMe: communication and awareness between members of a closely-knit group (with Natalia Marmasse) Proceedings of Ubicomp 2004

An audio-based personal memory aid (with S. Vemuri, W. Bender, S. Tellex and B. Lassey) Proceedings of Ubicomp 2004

Improving speech playback using time-compression and speech recognition (with Sunil Vemuri, Philip DeCamp, and Walter Bender) Proceedings of CHI 2004

Impromptu: managing networked audio applications for mobile users (with Kwan Lee, Jang Kim, and Mark Ackerman), Proceedings of MobiSys 2004

TalkBack: a conversational answering machine (with Vidya Lakshminpathy and Natalia Marmasse) Proceedings of UIST 2003

"ListenIn" to domestic environments from remote locations (with Gerardo Vallejo) Proceedings of the 2003 International Conference on Auditory Display (ICAD)

Safe & Sound: a wireless leash (with Natalia Marmasse) Extended Abstracts, Proceedings of CHI 2003

Mediated voice communication via mobile IP (with Jang Kim, Kwan Lee, Gerardo Vallejo, and Mark Ackerman), Proceedings of UIST 2002.

The Audio Notebook: Paper and pen interaction with structured speech (with Lisa Stifelman and Barry Arons), Proceedings of CHI 2001

Synthetic News Radio (with Keith Emnett) IBM Systems Journal, Vol. 39 Nos. 3-4, pp. 646-659, 2000.

Everywhere messaging (with Natalia Marmasse, Stefan Marti, Nitin Sawhney, and Sean Wheeler) IBM Systems Journal, Vol. 39 Nos. 3-4, pp. 660-677, 2000.

Location-aware information delivery with comMotion (with Natalia Marmasse), Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing, pp. 157-171, Springer, 2000.

Nomadic Radio: Scalable and contextual notification for wearable audio messaging (with Nitin Sawhney), Proceedings of CHI 1999.

Speaking and listening on the run: Design for wearable audio computing (with Nitin Sawhney), Proceedings of International Symposium on Wearable Computing, 1998.

Audio Hallway: A virtual acoustic environment for browsing, Proceedings of UIST 1998.

Dynamic Soundscape: Mapping time to space for audio browsing (with Minoru Kobayashi), Proceedings of CHI 1997.

CLUES: Dynamic personalized message filtering (with Matt Marx), Proceedings of CSCW 1996.

Using acoustic structure in a hand-held audio playback device (with Deb Roy), IBM Systems Journal, Vol 35, Nos. 3 and 4, 1996.

Mailcall: Message presentation and navigation in a nonvisual environment (with Matt Marx), Proceedings of CHI 1996

AudioStreamer: Exploiting simultaneity for listening (with Atty Mullins), short paper, CHI 1995.

Multimedia momadic services on today's hardware, IEEE Network, September/October 1994.

Putting people first: Specifying proper names in speech interfaces (with Matt Marx), proceedings of UIST 1994.

Chatter: A conversational learning speech interface (with E. Ly) AAAI Workshop on Intelligent Multi-Media Multi-Modal Systems, 1994.

Voice Communication with Computers: Conversational Systems. New York: Van Nostrand Reinhold. 1994.

Capturing, structuring, and representing ubiquitous audio (with D. Hindus and C. Horner), ACM Transactions on Information Systems, Vol. 11, No. 4, October 1993.

Speech Recognition Architectures for Multimedia Environments, (with E. Ly and B. Arons), Proceedings of the 1993 AVIOS Conference, September 1993.

Phoneshell: the Telephone as Computer Terminal, Proceedings of the ACM Multimedia Conference, August 1993.

Voicenotes: A Speech Interface for a Hand-Held Voice Notetaker (with L. Stifelman, B. Arons, and E. Hulteen), Proceedings of INTERCHI'93, April 1993.

From Desktop Audio to Mobile Access: Opportunities for Voice in Computing, book chapter in Advances in Human-Computer Interaction, Vol. 4, H.R. Hartson and D. Hix editors. 1992.

Ubiquitous Audio: Capturing Spontaneous Collaboration (with D. Hindus), Proceedings of CSCW'92, November 1992.

Integrating Audio and Telephony in a Distributed Workstation Environment (with S. Angebrannt, R. Hyde, D. Luong, and N. Siravara), Proceedings of the Summer 1991 USENIX Conference, June 1991.

Augmenting a Window System with Speech Input (with M. Ackerman and D. Hindus), Computer, IEEE Computer Society, Vol. 23, No. 8, August 1990.

Observations on Using Speech Input for Window Navigation (with D. Hindus, M. Ackerman, and S. Manandhar), Proceedings, Human-Computer Interaction, Interact '90, IFIP, August 1990.

Phonetool: Integrating Telephones and Workstations (with S. Casner), Proceedings, GLOBECOM '89, IEEE Communications Society, November 1989.

Desktop Audio (with B. Arons), UNIX Review, October 1989.

Synthetic Speech for Real Time Direction-Giving (with J. Davis), IEEE Transactions on Consumer Electronics, IEEE, September 1989.

The Back Seat Driver: Real Time Spoken Driving Instructions (with J. Davis), Proceedings, IEEE Vehicle Navigation and Information Systems Conference, IEEE, Toronto, Canada, September 1989.

An Audio and Telephone Server for Multi-media Workstations (with M. McKenna), Proceedings, Second IEEE Conference on Workstations, IEEE, Palo Alto, CA., 1988.

Employing Voice Back Channels to Facilitate Audio Document Retrieval, Proceedings, ACM Conference on Office Information Systems (COIS), Santa Clara, CA, 1988.

Conversational Telecommunications Environments, Proceedings, Second International Conference on Human-Computer Interaction, 1987.

Understanding Speech Without Recognizing Words, Proceedings, American Voice Input/Output Society Conference, AVIOS, 1987.

A Robust Parser and Dialog Generator for a Conversational Office System (with B. Arons and C. Simmons), Proceedings, American Voice Input/Output Society Conference, AVIOS, Palo Alto, CA, 1987.

Integrated Messages and Network Services for a Personal Workstation, IEEE Workshop on Telematics and Message Handling Systems, IEEE, 1986.

Voice Interaction in an Integrated Office and Telecommunications Environment, Proceedings, American Voice Input/Output Society Conference, AVIOS, San Francisco, CA, 1985.

Voice Communication with Computers, book chapter in Advances in Human-Computer Interaction, H. R. Hartson ed., 1985.

Speech Synthesis Gives Voiced Access to an Electronic Mail System, Speech Technology, Vol. 2, No. 3, Aug/Sept 1984.

A Conversational Telephone Messaging System (with B. Arons), IEEE Transactions on Consumer Electronics, IEEE, Vol CE-30, August 1984.

Phone Slave: A Graphical Telecommunications Interface (with B. Arons), Proceedings, Society for Information Display International Symposium, SID, San Francisco, CA, June 1984.

Input/Display Registration in a Stereoscopic Workstation, Displays, April 1984.

Remote Access to Voice and Text Messages, Proceedings, American Voice Input/Output Society Conference, AVIOS, Washington D.C., 1984.

Fuzzy Fonts: Analog Models Improve Digital Text Quality, Proceedings, National Computer Graphics Association Conference, Chicago, IL, 1983.

Greyscale Fonts Designed From Video Signal Analysis, Society of Applied Learning Technology, Houston, TX, 1983.

Spatial Input/Display Correspondence in a Stereoscopic Computer Graphic Work Station, Proceedings, ACM/SIGGRAPH, Detroit, MI, 1983.

A Programmable Virtual Vocabulary Speech Processing Peripheral (with W. Bender), Proceedings, American Voice Input/Output Society Conference on Voice Data Entry Systems Applications, AVIOS, 1983.

The Intelligent Voice Interactive Interface (with E. A. Hulteen), Proceedings, Human Factors in Computer Systems, National Bureau of Standards/ACM, Gaithersburg, MD, 1982.

Interactive Three-Dimensional Computer Space, Proceedings, SPIE Conference on Processing and Display of Three-Dimensional Data, SPIE, San Diego, CA, 1982, Vol. 367.

Speech Communications, a Systems' Approach, Proceedings, American Voice Input/Output Society Conference on Entry Systems Applications, 1982.

Voice Interaction: Putting Intelligence into the Interface, Proceedings, IEEE International Conference on Cybernetics and Society, IEEE, Seattle, WA, 1982.

The Intelligent Ear: A Graphical Interface to Digital Audio, Proceedings, IEEE International Conference on Cybernetics and Society, IEEE, Atlanta, GA, 1981.

Soft Typography, Information Processing 1980, IFIPS, S. Lavington ed., North-Holland Publishing Co., 1980.
