

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.

DIALECT, LLC,
Patent Owner

Case IPR2024-00751
U.S. Patent No. 8,447,607 B2
Issue Date: May 21, 2013

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. 8,447,607 B2

TABLE OF CONTENTS

	Page
I. INTRODUCTION AND QUALIFICATIONS	6
A. Qualifications and Experience	6
B. Materials Considered	10
II. PERSON OF ORDINARY SKILL IN THE ART	11
III. STATEMENT OF LEGAL PRINCIPLES	13
A. Claim Construction	13
B. Obviousness (§ 103)	15
IV. THE '607 PATENT	21
A. Overview of the Specification	21
B. The Challenged Claims	23
V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIM	24
A. Brief Summary of Prior Art	25
1. Prior Art Common to All Grounds	25
(a) Coffman (EX1003)	25
(b) Kennewick (EX1004)	30
(c) Ross (EX1005)	33
2. Additional Prior Art for Certain Dependent Claims	39
(a) Lee (EX1006)	39
B. Ground 1: Claims 1-15 Are Obvious Over Coffman In View Of Kennewick and Ross	42
1. Claim 1: “A device for processing natural language inputs, comprising one or more processors configured to:” (1[pre])	42

TABLE OF CONTENTS

(continued)

	Page
(a) “receive a multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;” (1[a])	46
(b) “generate a non-speech transcription from the non-speech input;” (1[b])	51
(c) “identify the user who provided the multi-modal natural language input;” (1[c])	54
(d) “generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;” (1[d]).....	56
(e) “generate a merged transcription from the speech-based transcription and the non-speech transcription;” (1[e])	65
(f) “identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the merged transcription;” (1[f])	67
(g) “identify a domain agent associated with the entry in the context stack;” (1[g]).....	79
(h) “determine a request based on the merged transcription; and” (1[h])	83
(i) “communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.” (1[i])	84
2. Claim 2: “The device of claim 1, wherein the response includes an aggregation of content gathered when the identified domain agent processes the request.”	86

TABLE OF CONTENTS
(continued)

	Page
3. Claim 3: “The device of claim 1, wherein the multi-modal natural language input is a first multi-modal natural language input and is received during a first session, the one or more processors further configured to receive a second multi-modal natural language input from another user during the first session.”	89
4. Claim 4: “The device of claim 1, wherein the one or more processors are configured to support interactions with a plurality of users during an interleaved session.”	91
5. Claim 5: “The device of claim 1, wherein the cognitive model is a first cognitive model, wherein the one or more processors are configured to generate the speech-based transcription based on a second cognitive model that includes information on one or more interactions between the device and a plurality of users of the device.”	94
6. Claim 6: “The device of claim 5, wherein the one or more processors are configured to generate the speech-based transcription further based on an environmental model that has information indicative of an environment of the user.”	96
7. Claim 7: “The device of claim 6, wherein the environmental model has information indicative of how noisy the environment of the user is.”	100
8. Claim 8: “The device of claim 1, wherein the one or more processors are configured to receive one or more subsequent multi-modal inputs, the subsequent multi-modal inputs including at least one of a subsequent natural language utterance or a subsequent non-speech input.”	100
9. Claim 9: “The device of claim 1, wherein the identified domain agent is configured to update the context stack in response to processing the request.”	103

TABLE OF CONTENTS
(continued)

	Page
10. Claim 10: “The device of claim 1, wherein the one or more processors are configured to synchronize the context stack with a context stack of another device.”.....	103
11. Claim 11: “The device of claim 10, wherein the one or more processors are configured to synchronize the cognitive model with a cognitive model of another device.”.....	107
12. Independent Claim 12.....	109
13. Claim 13: “The method of claim 12, wherein the cognitive model is a first cognitive model, and wherein the generating the speech-based transcription is based on a second cognitive model that includes information on one or more interactions between the device and a plurality of users of the device.”	111
14. Independent Claim 14:.....	111
15. Claim 15: “The device of claim 14, wherein the one or more processors are configured to generate the speech-based transcription further based on an environmental model, wherein the environmental model includes information indicative of how noisy the environment of the user is.”	114
C. Ground 2: Claims 5-7, 13-15 Are Obvious Over Ground 1 Prior Art In Further View Of Lee.....	114
VI. NO SECONDARY CONSIDERATIONS OF NONOBVIOUSNESS	118
VII. CONCLUSION	119

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 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. 8,447,607 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. Put That There is also highly cited as a pioneering use of multi-modal input. A user could specify locations and select objects on a map by pointing, while simultaneously interacting with the objects and locations by voice input. 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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

was speaking over a teleconference channel. Along with my students we built the first system for real time spoken driving directions, including 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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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, 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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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 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. 8,447,607 B2 (“’607 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. 8,447,607 B2

Exhibit No.	Description of Document
1001	U.S. Patent No. 8,447,607 B2 to Chris Weider et al. (filed June 4, 2012, issued May 21, 2013) (“ ’607 ” or “ ’607 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	U.S. Patent App. Pub. No. 2002/0133354 A1 to Steven I. Ross (filed August 16, 2001, published September 19, 2002) (“ Ross ”)
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	Redline comparison between Kennewick and the ’607 patent
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 <i>Merriam-Webster’s Collegiate Dictionary</i> (10th ed. 1994)

II. PERSON OF ORDINARY SKILL IN THE ART

11. I understand that, under the patent laws in effect before the America Invents Act (“AIA”) of 2011, an assessment of claims of a patent filed before the

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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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 ’607 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 ’607 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 that “[m]ore specifically, the invention enables mobile users to submit natural language speech and/or non-speech questions or commands in a wide range of domains.” (’607, 1:20-25.) 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 '607 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 a person of ordinary skill in the art would have understood the claim terms to mean.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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 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.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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 general knowledge present in the art, or common sense. I understand that a

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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 that a need or problem known in the field at the time of the invention can also provide

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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.

28. I understand that obviousness may be shown by demonstrating that it

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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. 8,447,607 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 '607 PATENT

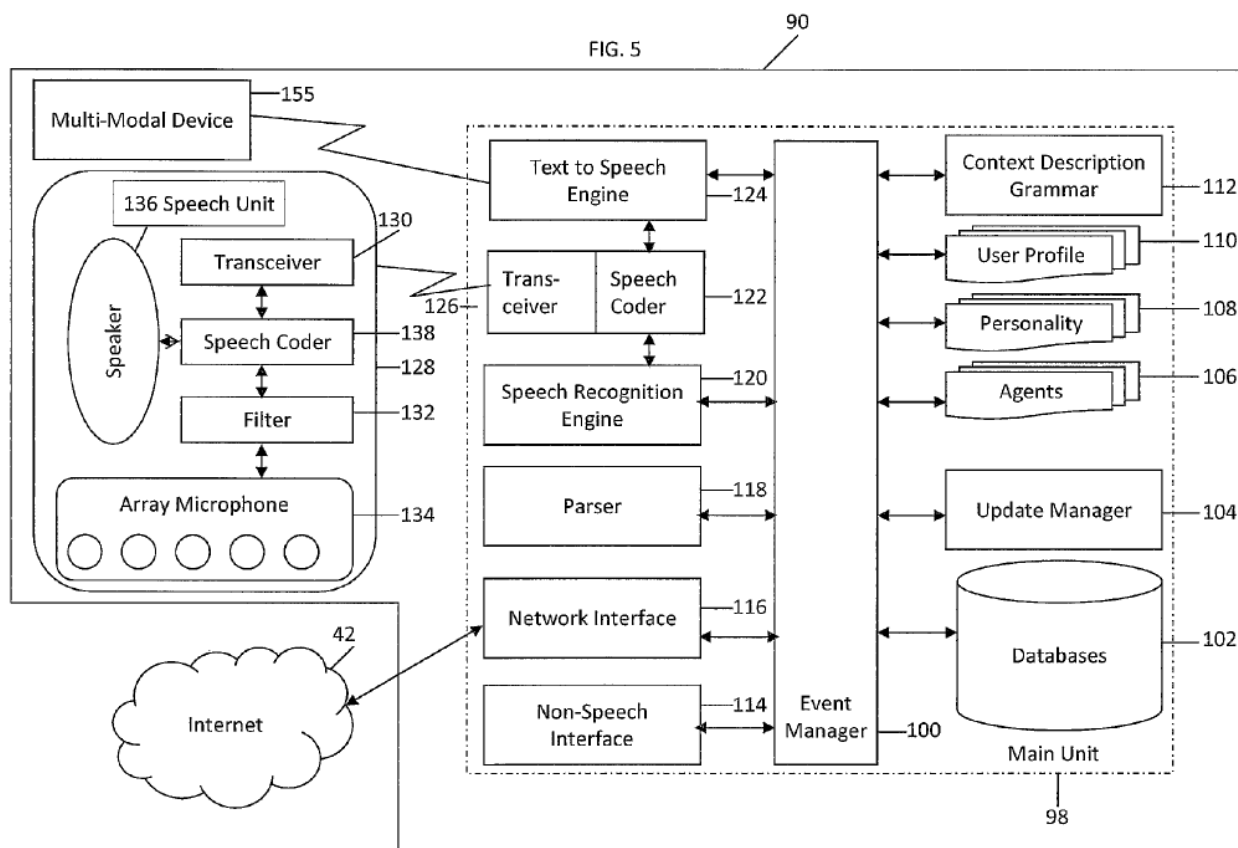
A. Overview of the Specification

32. The '607 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 that “[m]ore specifically, the invention enables mobile users to submit natural language speech and/or non-speech questions or commands in a wide range of domains.” ('607, 1:20-25.)

33. The '607 patent admits 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 been applied to the parsing of speech queries.” ('607, 1:65-2:2.) According to the '607 patent, however, “[t]here remain a number of significant barrier to creation of a complete natural language verbal

and/or textual-based query and response environment.” (’607, 2:6-8.) The ’607 patent purports to “overcome...drawbacks” of prior systems. (’607, 2:57-58.)

34. Figure 5 of the ’607 patent, reproduced below, “is an overall diagrammatic view of the interactive natural language processing system according to an embodiment of the invention”:



(’607, Fig. 5, 15:38-40.) As shown in the figure above, the system includes, among other things, speech unit 128, speech recognition engine 120, context description grammar module 112, parser 118, agents 106, multi-modal device 155 and non-speech interface 114. Speech unit 128 includes a microphone to receive a spoken

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

utterance from a user. ('607, 25:50-52.) After initial processing, the speech is passed to speech recognition engine **120** for processing using the context description grammar module **112**. ('607, 26:18-20.) Then, “[a]ny recognized information may be processed by the parser **118**, which transforms information into complete algorithms and questions using data supplied by knowledge agents.” ('607, 26:20-23.) “The knowledge agents may then process the commands or questions.” ('607, 26:27-31.) The agents may return results to the user. ('607, 26:36-37.) Non-speech interface **114** and multi-modal devices **155** may be used as a substitute or complement to the speech interface. ('607, 26:51-53.)

35. I discuss additional aspects of the '607 patent as needed in the analysis below.

B. The Challenged Claims

36. This Declaration addresses claims 1-15 of the '607 patent. Claim 1 recites:

1. A device for processing natural language inputs, comprising one or more processors configured to:
 - [a]** receive a multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;
 - [b]** generate a non-speech transcription from the non-speech input;

- [c] identify the user who provided the multi-modal natural language input;
- [d] generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;
- [e] generate a merged transcription from the speech-based transcription and the non-speech transcription;
- [f] identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the merged transcription;
- [g] identify a domain agent associated with the entry in the context stack;
- [h] determine a request based on the merged transcription; and
- [i] communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.

(’607, 38:38-62 (Claim 1) (bracketed notation (e.g., “[a]”) added).)

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

V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIM

38. I have reviewed and analyzed the prior art references and materials listed in **Part I.B** above. In my opinion, claims 1-15 of the ’607 patent would have been obvious to a person of ordinary skill in the art based on the following

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

combination of the prior art.

Ground	Claims	Basis for Challenge under § 103
1	1-15	Coffman (EX1003) in view of Kennewick (EX1004) and Ross (EX1005)
2	5-7, 13-15	Ground 1 Prior Art + Lee (EX1006)

39. I am informed by counsel that each of the references cited in the ground above qualifies as prior art to the challenged claims because each reference was filed and/or published before the '607 patent priority date (August 29, 2005).

A. Brief Summary of Prior Art

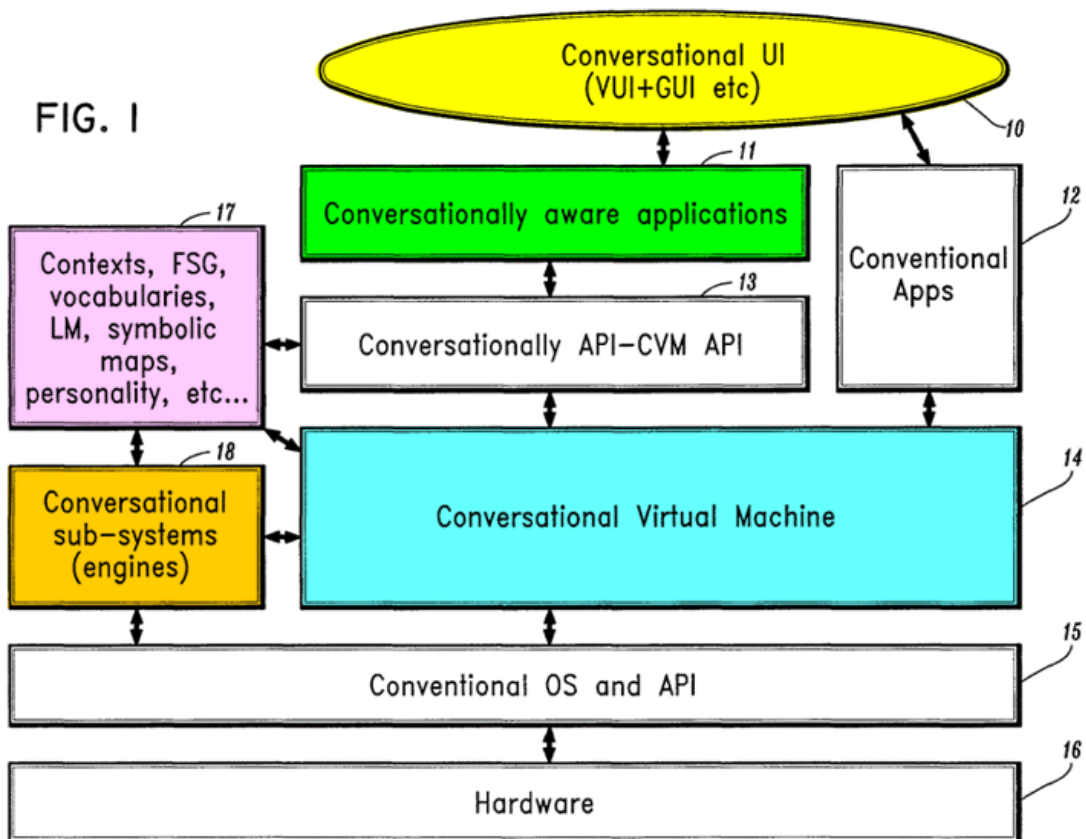
1. Prior Art Common to All Grounds

(a) Coffman (EX1003)

40. **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 that Coffman qualifies as prior art because it published on April 13, 2000, before the earliest priority date of the '607 patent.

41. 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 non-speech

input/output means). (Coffman, 4:6-13, 11:22-24.) 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).)

42. 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.)

43. “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”).)

44. 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.”.)

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

(b) Kennewick (EX1004)

46. **Kennewick**, U.S. Patent App. Pub. No. 2004/0044516 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 was filed on June 3, 2003 and published on March 4, 2004, both dates being before the earliest priority date of the ’607 patent; in this case, both more than one year before the ’607 patent priority date.

47. The ’607 patent shares a number of named inventors with Kennewick (e.g., Robert A. Kennewick and Richard Kennewick), and appears to incorporate a substantial amount of Kennewick’s disclosures. (*See* EX1007.) Figure 1 of Kennewick, reproduced below, is nearly identical to Figure 5 of the ’607 patent. It depicts “an overall diagrammatic view according to one embodiment of the invention” (Kennewick, ¶0073):

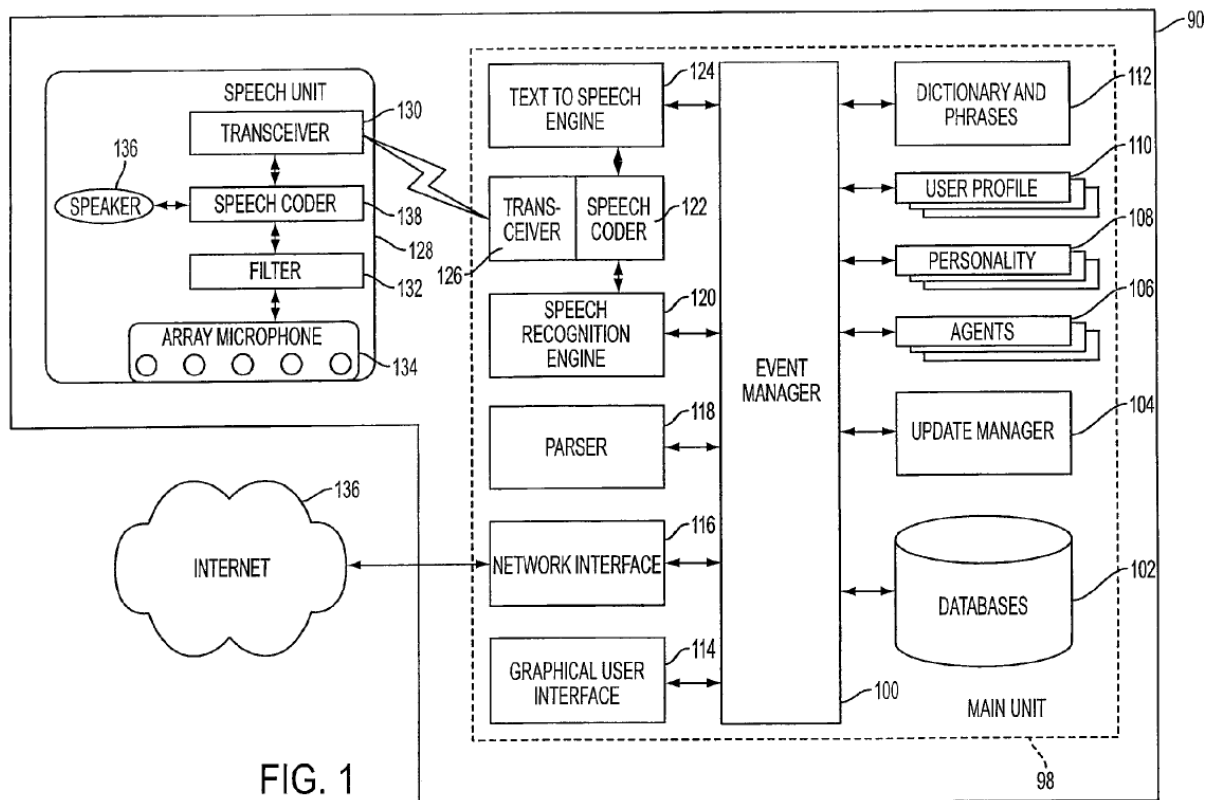


FIG. 1

(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

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.) “Once the context for the question or command has been determined, the parser **118** can invoke the correct agent **156**, **150**.” (Kennewick, ¶0152.) The agent returns results to the user. (Kennewick, ¶0089.)

48. My analysis points to Kennewick for, among other things, its teachings in connection with user “**cognitive models**” recited in the challenged claims, such as in 1[d], which recites “**generat[ing] a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device.**” 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.)

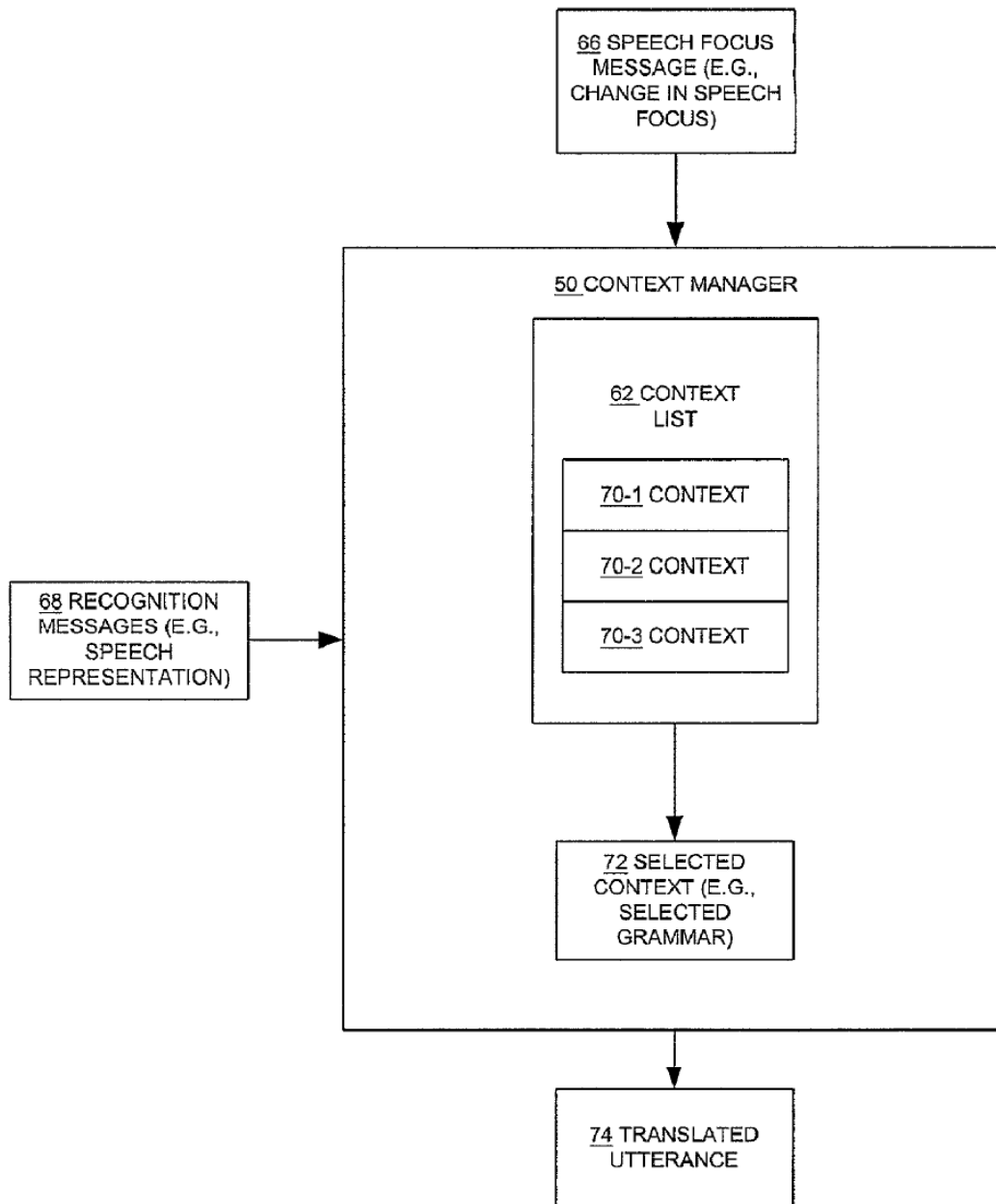
49. I discuss additional aspects of Kennewick as needed in my analysis of the challenged claims below.

(c) **Ross (EX1005)**

50. **Ross**, U.S. Patent Application Publication No. 2002/0133354 A1 to Steven I. Ross et al., is entitled “System and Method for Determining Utterance Context in a Multi-Context Speech Application.” I am informed that Ross qualifies as prior art because it was filed on August 16, 2001 and published on September 19, 2002, both dates being before the earliest priority date of the ’607 patent. Ross, as its title suggests, describes techniques for determining a context associated with a user’s spoken command or question in order to determine an application to invoke to process the command or question, just like Coffman, Kennewick and the ’607 patent. (Ross, e.g., ¶¶0010, 0013.)

51. My analysis points to Ross for its teachings regarding the use of context information to match a transcription of user input to select an application to invoke. For example, 1[f] recites a step of identifying an entry in a context stack that **“matches information in the merged transcription.”** As I explain in my analysis below, to the extent that Coffman alone does not disclose and render obvious the claimed identifying step, it would have been obvious in view of Ross.

52. Ross describes a technique where identifying a context involves evaluating grammars associated with potential contexts against a user’s recognized speech input—Figure 4 of Ross illustrates a general overview:



(Ross, Fig. 4.) As shown above, Ross includes a context manager **50** that includes a context list **62**. (Ross, ¶0033.) Context list **62**, in turn, “includes contexts **70** (e.g., **70-1**, **70-2**, **70-3**, etc.) for speech-enabled applications **26**, which represent grammars

for the applications 26.² (Ross, ¶0033.) A user's transcribed utterance (which Ross calls a "recognition message") is tested against the grammars in the context list in order to determine an application to invoke:

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

(Ross, ¶0034.)

53. Ross further makes clear that "test[ing]...against the active grammars" and finding a successful "match" involves matching information in a user's transcribed utterance to grammar information of the context entries in the context list. Ross explains that "a grammar is defined for each application **26**" and describes two example grammars, one for an electronic mail application:

² All underlining added to quoted material unless noted.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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

(Ross, ¶¶0035, 0040) and one for a calendar application:

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

(Ross, ¶0046). A grammar expression can generally comprise word tokens, rule references, and grammar operators. For example, in the grammar for the electronic mail application shown above, the line “<message> = the? <nth> message | it | this,” defines a rule named “message.” The rule specifies a grammar expression that includes word tokens (e.g., “the,” “message”), a reference to another rule (“<nth>”) and a grammar operator (“?”). As shown above, each grammar includes a number of grammar expressions. Therefore, as Ross explains, the grammar for the electronic mail application allows spoken phrases such as “open the first message,” “create a message,” “send this,” and “print it” to be matched, while the grammar for the calendar application matches phrases such as “open the first appointment,” “create

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

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

If both of these grammars were loaded into the context manager **50**, (ignoring foreground/background issues in this example, assuming that the grammars are the same for both foreground and background), the speech center system **20** is listening for any of the phrases accepted by either grammar. If the speech center **20** hears a phrase such as “print

the first message” or “print the first appointment,” the context manager 50 can readily figure out the intended target application 26 for the uttered sentence. Only one grammar will accept the phrase, which thus indicates the selected context 72 for that phrase and that associated application 26 is the one that should be targeted to receive the corresponding command.

If the sentence is “print it” however, both grammars are capable of accepting the utterance. The context manager 50 therefore has to make a choice by referring to the context list 62 of applications in order of recency of access. The context manager 50 tests the utterance against these grammars (indicated by the contexts 70 in the context list 62) in priority order, and passes the commands on to the first application 26 that has a grammar that will accept the phrase.

(Ross, ¶¶0052-0053.)

55. In addition, with respect to context list 62, Ross teaches that the contexts in the list are sorted in priority order, such that the most-recently-accessed context will be the first context evaluated when processing the next user utterance.

(Ross, ¶¶0035 (“[W]henver an application 26 which is not the top priority application 26 is chosen as the target for a speech command, the application 26 indicated by the selected context 72 will move to the head of the list 62. In this way, the last application 26 that the user touched or talked to will get the first opportunity at interpreting the next user utterance, and the other applications 26 will be ordered

in a most-recently-accessed way.”), 0037 (“[T]he context manager **50** receives the recognition message **68** and selects a context **70** for a speech enabled application **26** by checking the speech representation against contexts **70**, starting at top of context list **62** to determine the highest priority context **70** that can process the speech representation”).) Ross’s priority ordering description of a context list is consistent with the well-known concept of a “**stack**,” in which the most recent item added to the stack will be the first item considered from the stack in a later operation. (EX1009, p.305 (definition of “last in, first out”).) This “last in, first out” (“LIFO”) behavior of a stack contrasts with, for example, the “first in, first out” (“FIFO”) behavior of a queue. (EX1009, pp.215 & 305 (definitions of “first in, first out” and “last in, first out”).)

56. I discuss additional aspects of Ross as needed in my analysis of the challenged claims below.

2. Additional Prior Art for Certain Dependent Claims

(a) Lee (EX1006)

57. 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 ’607 patent.

58. My analysis points to Lee in connection with claims 5, 13 and 14, which recite using a **“second cognitive model”** or **“general cognitive model”** to generate a speech-based transcription. The **“second cognitive model”** as recited in claims 5 and 13 **“includes information on one or more interactions between the device and a plurality of users of the device,”** and the **“general cognitive model”** similarly **“includes information on one or more prior interactions between the device and a plurality of users.”** As I explain below, Kennewick discloses the claimed **“second cognitive model”** and **“general 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.)

59. 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 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.)

60. As I explain for Ground 2 below, it would have been obvious to incorporate Lee's teachings into the Coffman's conversational system as taught by the combination of Coffman, Kennewick and Ross, and it would have predictably resulted in generating a speech-based transcription using a “**second cognitive model**” or “**general cognitive model**” as claimed.

B. Ground 1: Claims 1-15 Are Obvious Over Coffman In View Of Kennewick and Ross.

1. Claim 1: “A device for processing natural language inputs, comprising one or more processors configured to:” (1[pre])

61. For reference, claim 1 recites:

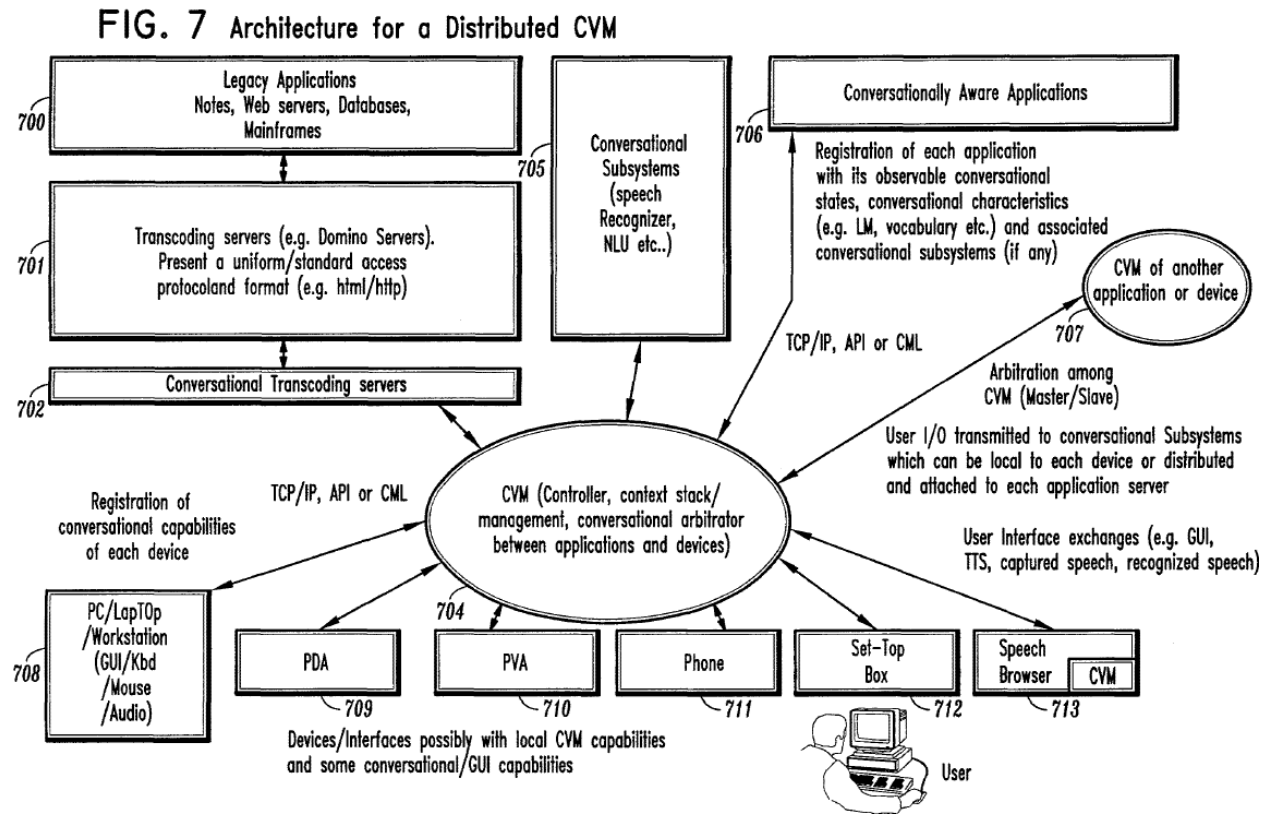
1. A device for processing natural language inputs, comprising one or more processors configured to:
 - [a] receive a multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;
 - [b] generate a non-speech transcription from the non-speech input;
 - [c] identify the user who provided the multi-modal natural language input;
 - [d] generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;
 - [e] generate a merged transcription from the speech-based transcription and the non-speech transcription;
 - [f] identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the merged transcription;
 - [g] identify a domain agent associated with the entry in the context stack;

[h] determine a request based on the merged transcription; and

[i] communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.

(’607, 38:38-62 (Claim 1) (bracketed notation (e.g., “[a]”) added).)

62. The preamble of claim 1 recites “**A device for processing natural language inputs, comprising one or more processors configured to[.]**” To the extent the preamble provides a limitation, Coffman discloses a conversational computing system implemented on a “**device,**” such as a PC, PDA or “conversational car PC (PVA, Personal Vehicle Assistant).” (Coffman, 13:1-3 (“The 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).”), 56:22-25 (“conversational PDA,” “conversational car PC (PVA, Personal Vehicle Assistant),” and “conversational smart phone”), 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.”).) Figure 7 of Coffman illustrates various devices, including PC 708, PDA 709 and PVA 710:



(Coffman, Fig. 7; *see also id.* 55:12-19 (“Referring now to Fig. 7, a diagram illustrates an architecture for a distributed CVM according to one aspect of the present invention. The heart of the distributed system is a CVM 704 (which may be located on a server, a PC, etc) which acts as the conversational arbitrator between a plurality of applications 706, devices 708-713, other CVM applications or devices 707 and conversational resources 705.”).³ As explained in the analysis below, the conversational computing device, such as a PC or PVA, is “**for processing natural**

³ All underlining added to quoted material unless noted.

language inputs,” such as spoken natural language utterances from a user.

63. A person of ordinary skill in the art would have understood and found it obvious for the device to “**compris[e] one or more processors,**” which 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 EX1009, 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 the device for

processing natural language inputs in Coffman's conversational computing system would comprise **"one or more processors."** A person of ordinary skill in the art would have further understood and found it obvious that the one or more processors would be **"configured"** to perform the claimed steps, which describe operations performed by a combination of computer hardware and/or software.

64. As shown below, the claimed device is disclosed and rendered obvious by the prior art.

- (a) **"receive a multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;"**
(1[a])

65. To start, Coffman discloses **"receiv[ing] a multi-modal...input from a user."** Coffman explains that its conversational computing system "provides a universal coordinated multi-modal conversational user interface (CUI) 10." (Coffman, 12:7-8.) With Coffman's multi-modal conversational system, "speech is used in parallel with mouse, keyboard, and other input devices such as a pen," as illustrated in Figure 6 below:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

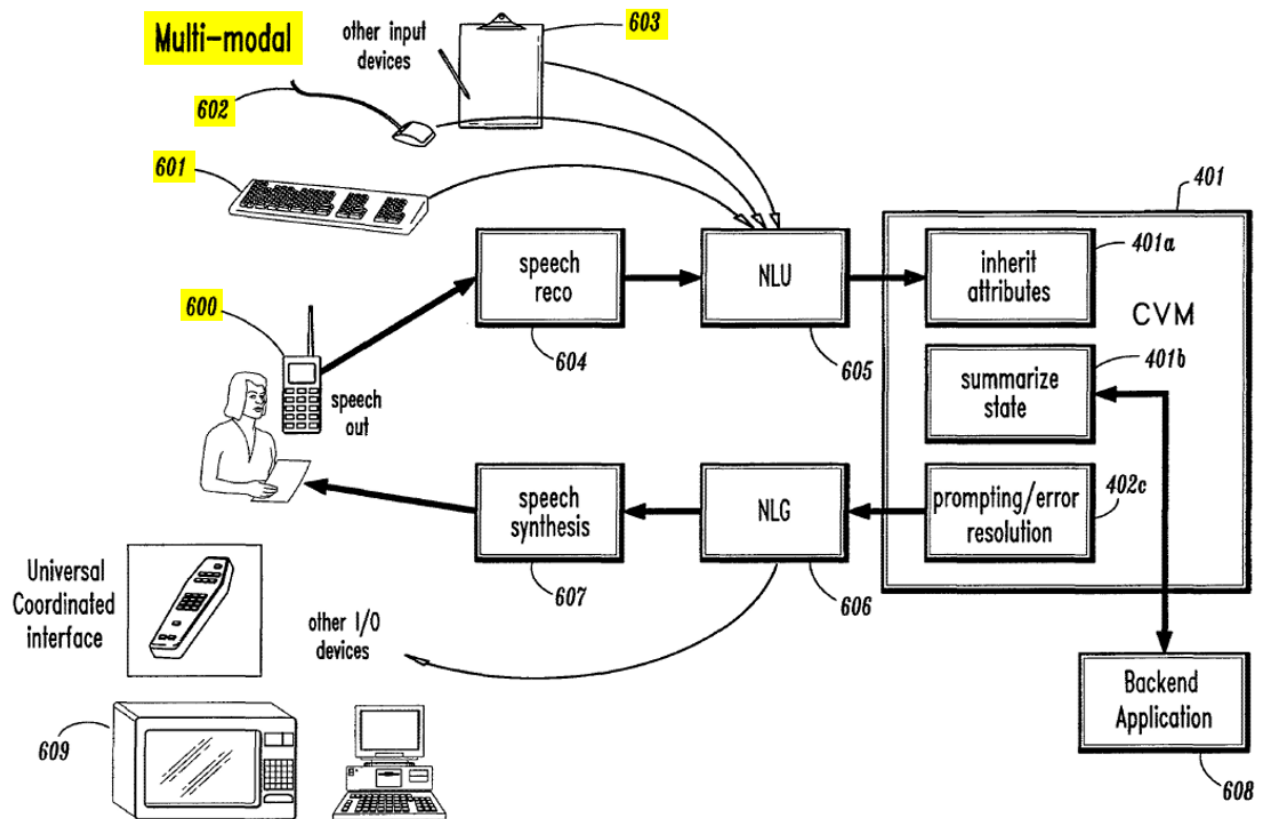


FIG. 6

(Coffman, Fig. 6 (highlighting added, showing multi-modal input); *see also id.*, 11:22-24 (“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’ [sic] 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.”), 19:20-23 (describing I/O API layer 210), 53:8-10 (“Voice input from a microphone (e.g. the standard voice input)”), 53:22-26 (“speech or any other input stream”).) Referring to Figure 6, Coffman describes receiving and processing multi-modal input from a user,

including speech (from phone 600)—an “**utterance**”—and “**non-speech input**”
(from, e.g., keyboard 601):

Referring now to Fig. 6, a diagram illustrates a conversational input/output interface in accordance with one embodiment of the present invention. 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.

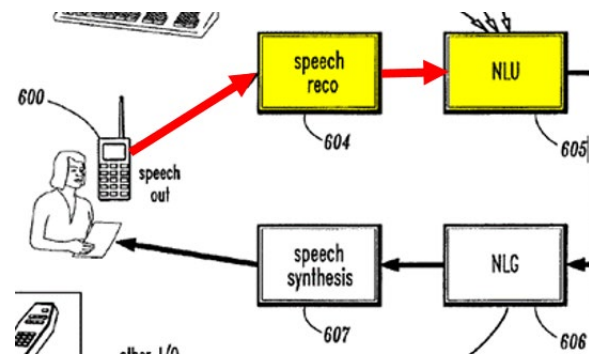
(Coffman, 37:14-23.) Coffman in fact provides a specific example of receiving multi-modal input that includes both an “**utterance**” and “**non-speech input**”:

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.

(Coffman, 6:30-7:3.)

66. Coffman further discloses receiving “**multi-modal natural language input**” from a user because the input includes “**a natural language utterance**.” In the context of the ’607 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.*, Gibbon [EX1008], p.444 (definition of “natural language”).) For example, English is a natural language, while C++ and Java are programming languages. Consistent with the ’607 patent, 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!”).)

67. Coffman's receipt (and use) 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).”).)

68. With respect to non-speech input, I further note that Coffman describes how non-speech input is also “subject to NLU processing 605” (Coffman, 37:14-22, Fig. 6) thereby disclosing another way that Coffman teaches receiving “**multi-modal natural language input**” from a user.

(b) “generate a non-speech transcription from the non-speech input;” (1[b])

69. The written description of the ’607 patent mentions transcribing speech input a number of times, but it does not appear to have any mention of transcribing non-speech input. (’607, e.g., 19:54 (“transcribes the verbal utterance to a text message”), 22:16-17 (“transcribe the verbal annotation to text”).) Transcribing (i.e. generating a “transcription”) is broadly understood as some form of recording, such as making a copy, whether verbatim or not, or converting information or data from one form to another. (EX1010, p.1253 (entry for “transcribe” stating “**1 a** : to make a written copy of **b** : to make a copy of (dictated or recorded matter) in longhand or on a machine (as a typewriter) **c** : to paraphrase or summarize in writing **d** : WRITE

DOWN, RECORD **2 a** : to represent (speech sounds) by means of phonetic symbols
b : TRANSLATE **2a c** : to transfer (data) from one recording form to another **d** :
to record (as on magnetic tape for later broadcast)...”).)

70. Coffman discloses “**generat[ing] a non-speech transcription from the non-speech input.**” Consistent with this ordinary understanding of “transcription,” Coffman describes how multi-modal input is transcribed into ASCII or binary input:

[T]he CVM provides the capability to have a natural dialog with NLU, NLG, contexts and mixed-initiatives sorted across multiple tasks, processes and discourses (with multiple domains). A conversational input interface is provided whereby a set of multi-mode input streams are each transcribed into an ASCII command or query (i.e., lists of attribute-value pairs or n-uples). Each input entity (command, NLU query field or argument unit (isolated letter, word, etc.) is associated with time-marks and appended accordingly to a compounded input stream.

(Coffman, 6:6-12; *see also id.*, 6:27-7:3 (“Using the conversational sub-systems, each input is converted into a binary or ASCII input (lists of attribute-value pairs or n-uples), which can be directly processed by the programming language as built-in objects.”), 38:26-29 (“In summary, the input procedure provides a set of multi-mode input streams, each transcribed into an ASCII command, query, or list of attribute

value n-uples. Each input entity (command, NLU query field or argument unit (isolated letter, word etc.) are associated to time-marks and appended accordingly to a compounded input stream.”).⁴ As shown in the excerpt above, Coffman describes transcribing “a set of multi-mode input streams.” Because, as discussed for 1[a], Coffman teaches that the “multi-mode input streams” include non-speech input, such as keyboard and mouse input, a person of ordinary skill in the art would have understood Coffman’s description of transcribing “a set of multi-mode input streams” to include transcribing non-speech input. In fact, Coffman provides a specific example of how speech input and non-speech input (a mouse click) are transcribed (“converted into a binary or ASCII input”) and ultimately “disambiguated into a set of attribute value pairs”:

Using the conversational sub-systems, each input is converted into a binary or ASCII input (lists of attribute-value pairs or n-uples), which can be directly processed by the programming language as built-in objects. Calls, flags and tags are automatically included to transmit between object and processes the conversational meta-information required to correctly interface with the different objects. Indeed, any

⁴“ASCII” refers to text formatted using the American Standard Code for Information Interchange (“ASCII”). (EX1009, p.36 (definition of “ASCII”).)

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.

(Coffman, 6:27-7:3.)

(c) “identify the user who provided the multi-modal natural language input;” (1[c])

71. Coffman discloses **“identify[ing] the user who provided the multi-modal natural language input”** because 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:

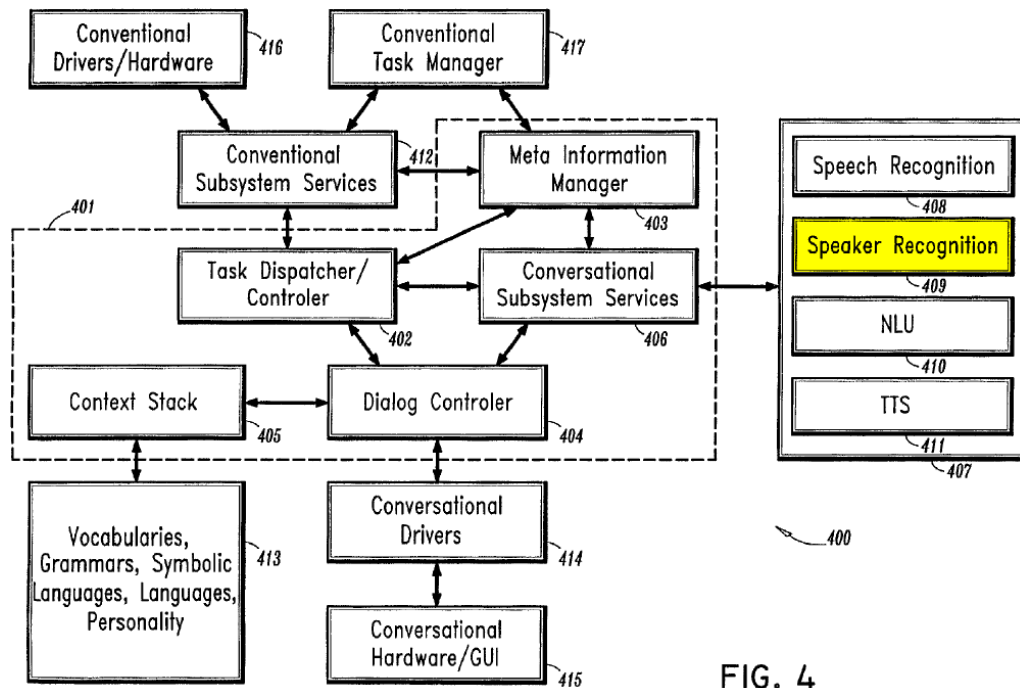


FIG. 4

(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.”), 48:7-9 (“[T]he user identity could be obtained through alternative procedures such as manual selection or input by the user of his or her identity.”), 46:28-29 (“[T]he stack of context should contain the identity of the user as the most recently authenticated identity.”).) “Text-

independent speaker identification,” as its name suggests, is a form of speaker identification using some 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”).)

- (d) **“generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;”**
(1[d])

72. Coffman discloses **“generat[ing] a speech-based transcription.”** As discussed above for 1[b], Coffman describes generating transcriptions of multi-modal input, including transcriptions of both speech-based input and non-speech input. (Coffman, 6:6-12 (“A conversational input interface is provided whereby a set of multi-mode input streams are each transcribed into an ASCII command or query (i.e., lists of attribute-value pairs or n-uples).”); *see also id.*, 38:26-29 (“[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.”).) For example, Coffman states, “[b]ased 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.” (Coffman, 41:2-4.) And as noted, Coffman provides an example of how both speech input and non-speech input (a mouse click) are transcribed and ultimately “disambiguated into a set of attribute value pairs.” (Coffman, 6:27-7:3 (“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.”).) As noted for 1[c] above, Coffman further teaches that the transcription can comprise information associating the recognized text with an identified user. (*See also* Coffman, 48:1-4 (“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).”), 8:22-23 (“Ambiguity is resolved at the level of each context or the context stack using the user identity.”).)

73. As explained below, it would have further been obvious to a person of ordinary skill in the art to generate a speech-based transcription “**based on a**

cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device.”

74. With respect to a “**cognitive model**,” the ’607 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.

(’607, 24:1-11.)

75. 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 on one or more prior interactions between the user and the device,**” 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 past interactions with the device that the system may use in connection with creating a transcription associated with a user of the user’s input.

76. **Kennewick** (EX1004) further confirms and renders obvious the claimed “**cognitive model.**” Kennewick, U.S. Patent Application Publication No. 2004/0044516 A1, is entitled “Systems and Methods for Responding to Natural Language Speech Utterance.” The ’607 patent shares a number of named inventors with Kennewick (e.g., Robert A. Kennewick and Richard Kennewick), and appears to incorporate a substantial amount of Kennewick’s disclosures. (See EX1007.) Figure 1 of Kennewick, reproduced below, is nearly identical to Figure 5 of the ’607 patent. It depicts “an overall diagrammatic view according to one embodiment of the invention” (Kennewick, ¶0073):

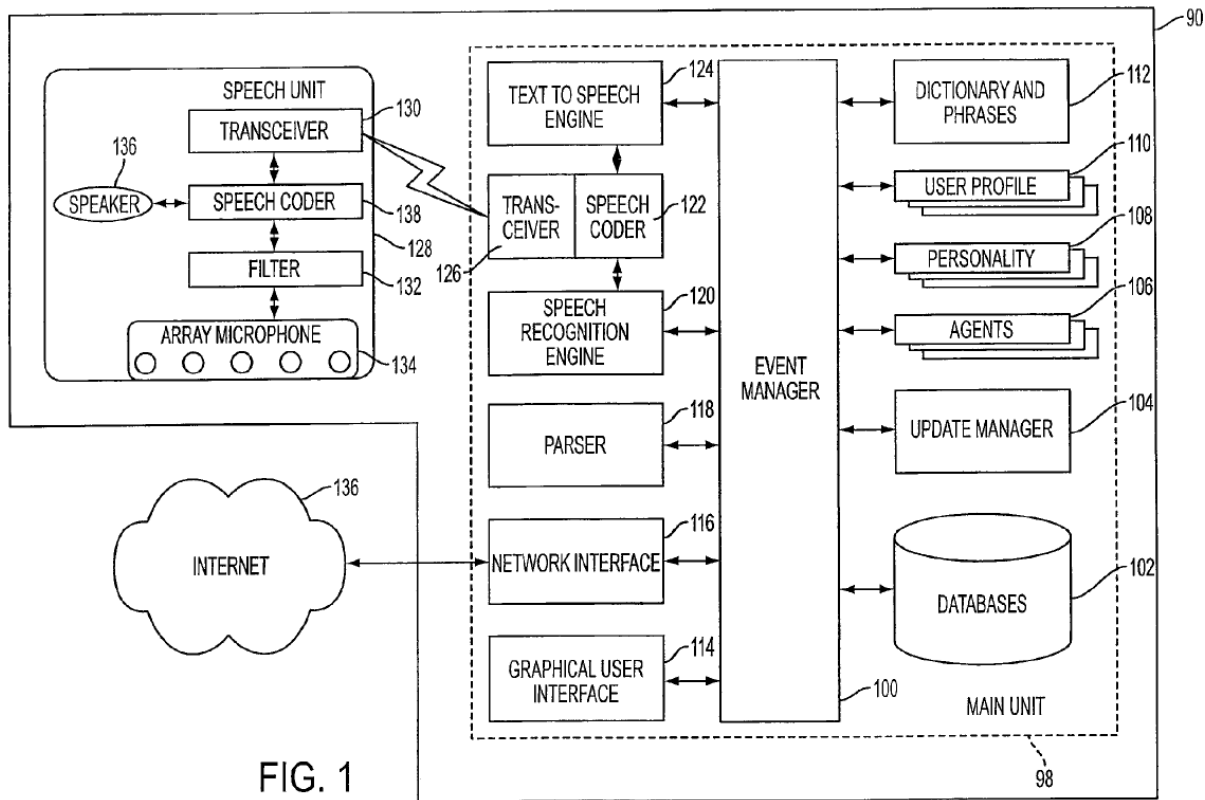


FIG. 1

(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. 8,447,607 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.) “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.)

77. 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 on one or more prior interactions between the user and the device**,” 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.”.)

78. Kennewick further discloses that the collected user profile information may be used in connection with speech recognition—i.e., “**generat[ing] a speech-based transcription**”—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.”).)

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

It would have been obvious to a person of ordinary skill in the art to 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 **“generat[ing] a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device”** as claimed.

80. Coffman and Kennewick are analogous references in the same field of speech recognition and processing as the ’607 patent. (’607, 1:20-25; Coffman, e.g., Abstract; Kennewick, ¶0003.) Coffman and Kennewick also would have been

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

reasonably pertinent to problems facing the '607 patent inventors, including collecting and using user profile information in connection with speech recognition and processing.

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

82. 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 ’607 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.

(e) **“generate a merged transcription from the speech-based transcription and the non-speech transcription;” (1[e])**

83. Coffman further discloses **“generat[ing] a merged transcription from the speech-based transcription and the non-speech transcription,”** describing how transcribed inputs from multi-mode input streams are “appended accordingly

to a compounded input stream”:

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). A conversational input interface is provided whereby a set of multi-mode input streams are each transcribed into an ASCII command or query (i.e., lists of attribute-value pairs or n-uples). Each input entity (command, NLU query field or argument unit (isolated letter, word, etc.) is associated with time-marks and appended accordingly to a compounded input stream.

(Coffman, 6:6-12; *see also id.*, 38:26-29 (“In summary, the input procedure provides a set of multi-mode input streams, each transcribed into an ASCII command, query, or list of attribute value n-uples. Each input entity (command, NLU query field or argument unit (isolated letter, word etc.) are associated to time-marks and appended accordingly to a compounded input stream.”).) Coffman further describes an example of how “transcripts are merged”:

In accordance with the present invention, the input problem is treated as a merge of the output of multiple decoders, ASCII transcription or a list of attribute value n-uples.. Each input stream is converted into its ASCII transcription and aligned with input time- marks via speech recognition processing 604. When different input stream are associated to the same task, the transcripts are merged as follows. First, commands and queries are sorted based on the time marks and appended to a single

data stream. Command formulation can be checked against FSG rules and re-sorted to satisfy the grammar rules. NLU queries do not necessarily require re-sorting. For NLU queries, the symbolic fields are filled for each stream, then compounded at the level of the final input stream.

(Coffman, 38:8-16.)

- (f) **“identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the merged transcription;” (1[f])**

84. As explained below, 1[f] would have been obvious in further view of **Ross (EX1005)**.

85. To start, 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.”).)

86. Coffman further makes clear that its “context stack” includes “**a plurality of entries...that are each indicative of context.**” Coffman explains, for example, that “each new task, process, or thread creates a new stack entry and is associated with a discourse.” (Coffman, 30:4-10; *see also id.*, 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.”).) Coffman further explains how context stack entries are searched when determining a context matching a user’s input. (Coffman, 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.”).)

87. Coffman further teaches “**identify[ing]...an entry in the context stack that matches information in the merged transcription.**” As noted above, Coffman explains that context stack entries are searched to identify a context matching the user’s input. (Coffman, 40:25-27, 42:1-5.) The search involves

comparing the transcribed input against information in the context stack entries:

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.

(Coffman, 41:2-5.) A context stack entry is identified when an acceptable match between the user's input and context stack is found:

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.

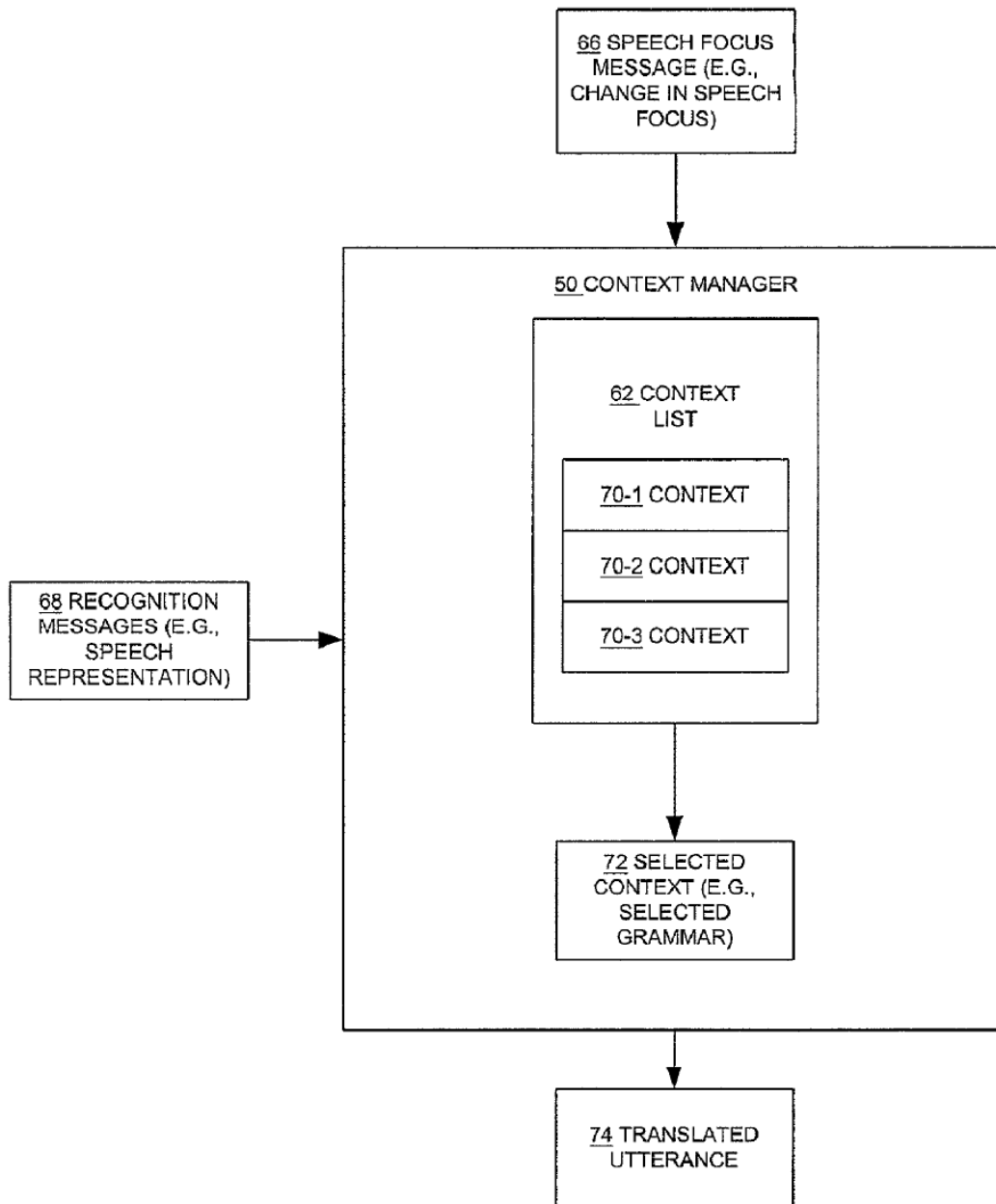
(Coffman, 42:1-5.) For instance, taking Coffman's example involving the utterance "I would like to open this" and mouse click input, Coffman teaches that the system matches the recognized word "open" in the user's utterance and the mouse click information to a context entry for an application's "open" 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.").)

This is therefore an example of “**match[ing] information in the merged transcription,**” because as discussed above for 1[e], Coffman teaches generating a merged transcription that would include transcriptions of the utterance and the mouse click input. (Coffman, e.g., 6:6-12, 38:8-16, 38:26-29.)

88. To the extent there is any question whether Coffman alone discloses or renders obvious 1[f], it would have been obvious in view of Ross, as explained next.

89. Ross, U.S. Patent Application Publication No. 2002/0133354 A1, is entitled “System and Method for Determining Utterance Context in a Multi-Context Speech Application.” (Ross, face page.) Ross, as its title suggests, describes techniques for determining a context associated with a user’s spoken command or question in order to determine an application to invoke to process the command or question, just like Coffman, Kennewick and the ’607 patent. (Ross, e.g., ¶¶0010, 0013.)

90. Ross describes a technique where determining a context involves evaluating grammars associated with potential contexts against a user’s recognized speech input—Figure 4 of Ross illustrates a general overview:



(Ross, Fig. 4.) As shown above, Ross includes a context manager **50** that includes a context list **62**. (Ross, ¶0033.) Context list **62**, in turn, “includes contexts **70** (e.g., **70-1**, **70-2**, **70-3**, etc.) for speech-enabled applications **26**, which represent grammars for the applications **26**.” (Ross, ¶0033.) A user’s transcribed utterance (a

“recognition message”) is tested against the grammars in the context list in order to determine an application to invoke:

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

(Ross, ¶0034.)

91. Ross further makes clear that “test[ing]...against the active grammars” and finding a successful “match” involves matching information in a user’s transcribed utterance to grammar information of the context entries in the context list. Ross explains that “a grammar is defined for each application **26**” and describes two example grammars, one for an electronic mail application:

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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

(Ross, ¶¶0035, 0040) and one for a calendar application:

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

(Ross, ¶0046). A grammar expression can generally comprise word tokens, rule references, and grammar operators. For example, in the grammar for the electronic mail application shown above, the line “<message> = the? <nth> message | it | this,” defines a rule named “message.” The rule specifies a grammar expression that includes word tokens (e.g., “the,” “message”), a reference to another rule (“<nth>”) and a grammar operator (“?”). As shown above, each grammar includes a number of grammar expressions. Therefore, as Ross explains, the grammar for the electronic mail application allows spoken phrases such as “open the first message,” “create a message,” “send this,” and “print it” to be matched, while the grammar for the calendar application matches phrases such as “open the first appointment,” “create an appointment,” “print the fourth appointment,” and “print it.” (Ross, ¶¶0041-0051.) The phrases that can be matched can be easily recognized by substituting the term in angle brackets with text (or an expression) to which it can correspond. For example, in the grammar for the electronic mail application, “<message>” can

correspond to “the? <nth> message,” “it,” or “this.” Therefore, the grammar expression “print <message>” can match “print it” (as well as “print this” or “print the first message” and so on). Ross discloses, therefore, that text in a user’s transcribed utterance, such as “print it,” matches the “print <message>” expression defined by the “mail” rule as well as the “it” expression defined by the “message” rule in the grammar for the electronic mail application. With respect to the calendar application grammar, “print it” would also match the “print <appointment>” and “it” expressions in the “appointment” rules.

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

If both of these grammars were loaded into the context manager **50**, (ignoring foreground/background issues in this example, assuming that the grammars are the same for both foreground and background), the speech center system **20** is listening for any of the phrases accepted by either grammar. If the speech center **20** hears a phrase such as “print the first message” or “print the first appointment,” the context manager **50** can readily figure out the intended target application **26** for the uttered sentence. Only one grammar will accept the phrase, which thus indicates the selected context **72** for that phrase and that associated application **26** is the one that should be targeted to receive the corresponding command.

If the sentence is “print it” however, both grammars are capable of accepting the utterance. The context manager **50** therefore has to make a choice by referring to the context list **62** of applications in order of recency of access. The context manager **50** tests the utterance against these grammars (indicated by the contexts **70** in the context list **62**) in priority order, and passes the commands on to the first application **26** that has a grammar that will accept the phrase.

(Ross, ¶¶0052-0053.)

93. In addition, with respect to the claimed “**context stack**,” Ross as noted describes a context manager **50** that includes a context list **62**. (Ross, e.g., Fig. 4, ¶0033.) The context list **62**, in turn, includes contexts **70**. The contexts in the context list **62** are sorted in priority order, such that the most-recently-accessed context will be the first context evaluated when processing the next user utterance. (Ross, ¶¶0035 (“[W]hen an application **26** which is not the top priority application **26** is chosen as the target for a speech command, the application **26** indicated by the selected context **72** will move to the head of the list **62**. In this way, the last application **26** that the user touched or talked to will get the first opportunity at interpreting the next user utterance, and the other applications **26** will be ordered in a most-recently-accessed way.”), 0037 (“[T]he context manager **50** receives the recognition message **68** and selects a context **70** for a speech enabled application **26** by checking the speech representation against contexts **70**, starting at top of context list **62** to

determine the highest priority context 70 that can process the speech representation”).) Ross’s priority ordering description of a context list is consistent with the well-known concept of a “stack,” in which the most recent item added to the stack will be the first item considered from the stack in a later operation. (EX1009, p.305 (definition of “last in, first out”).) This “last in, first out” (“LIFO”) behavior of a stack contrasts with, for example, the “first in, first out” (“FIFO”) behavior of a queue. (EX1009, pp.215 & 305 (definitions of “first in, first out” and “last in, first out”).)

94. **Rationale and Motivation to Combine (Coffman and Kennewick with Ross).** It would have been obvious to a person of ordinary skill in the art to further combine Coffman and Kennewick with Ross’s teachings regarding matching transcribed user utterances with grammar information in context entries in a context stack. The combination would have predictably resulted in Coffman’s conversational system identifying a context stack entry that matches information in a transcribed user’s utterance. The combination would have therefore disclosed **“identify[ing], from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the merged transcription”** as claimed.

95. Ross is analogous reference in the same field of speech recognition and

processing as the '607 patent. ('607, 1:20-25; Ross, e.g., ¶¶0010, 0013.) Ross would have also been pertinent to a number of problems facing the '607 patent inventors, including techniques for using a grammar to identify a context for processing a user command or question and techniques for choosing an application to invoke to process a user's a command or question in a user's transcribed utterance.

96. A person of ordinary skill in the art would have been motivated to combine. The combination would have involved the application of known techniques to improve a similar system in the same way. Coffman, for its part, explains that context stack entries include information "needed for input understanding," including "possible queries for a speech input" and "grammar[s]." (Coffman, 29:18-30:3.) Coffman does not detail how transcribed user input is matched against information in context stack entries, such as "possible queries" and "grammars." But as shown above, Ross provides a detailed explanation and examples of such a matching process. Ross's matching process similarly involves using grammar information in context entries stored in a context list that, as explained, is organized as a "stack." Therefore, a person of ordinary skill in the art, looking to implement or adapt Coffman's conversational system, would have been motivated to look to Ross's more detailed disclosures for further context and understanding of Coffman's teachings.

97. A person of ordinary skill in the art would have had a reasonable expectation of success. The use of grammars and grammar expressions to match text was well-known, and implementing Ross's techniques in Coffman's system would have involved the straightforward application of conventional techniques and technologies. For example, Ross explains that the grammars can be specified using the well-known "Backus Naur Form (BNF)." (Ross, ¶0060 ("In a preferred embodiment, the grammar used for each speech-enabled application 26 may be a Backus Naur Form (BNF) grammar as shown above."); EX1009, p.49 (dictionary entry for "Backus Naur Form").) Moreover, both the '607 patent's, as well as Coffman's, cursory descriptions of grammars confirm that their use was well-known, as I understand that a patent need not teach, and preferably omits, what is well-known in the art. A person of ordinary skill in the art would have found the disclosures of Coffman, Kennewick and Ross sufficiently detailed to implement the techniques on which I rely, without undue experimentation.

(g) "identify a domain agent associated with the entry in the context stack;" (1[g])

98. 1[g] recites a "**domain agent**." Generally speaking, in the software 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 ’607 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.” (’607, 25:34-36, 28:11-12.)

99. 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 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,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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”).)

100. 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 ’607 patent’s description of “domain agent,” also usually provides functionality related to a specific domain, such as a particular application/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.”).)

101. A conversationally-aware application in Coffman is also “**associated with [an] entry in the context stack.**” As discussed for 1[f], 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.)

102. Coffman therefore teaches “**identify[ing] a domain agent associated with the entry in the context stack**” because, by identifying a context stack entry matching the user’s input, the associated conversationally-aware application is selected to process a command/query based on the user’s input. (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.”).)

(h) “determine a request based on the merged transcription; and” (1[h])

103. I addressed 1[h] in my analysis of 1[f] above. As explained, both Coffman and Ross describe determining a command/query (“**determin[ing] a request**”) for an application to carry out based on transcribed user input—and in Coffman, the user input is included in a “**merged transcription.**” (Coffman, e.g., 6:32-7:3 (describing “I would like to open this” example), 37:2-4 (“The first task to have its dialog completed by producing a non-ambiguous query is executed”), 41:2-5 (“The output of these subsystems is run down the context stack 405 to extract the active query and complete it.”), 42:1-5 (“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.”); Ross, e.g., ¶¶0052-0053 (e.g., “associated application **26** is the one that should be targeted to receive the corresponding command,” “passes the commands on to the first application **26** that has a grammar that will accept the phrase”), 0021 (“The output **16** is a command or other input based on the spoken utterance **14** directed to the speech enabled external application **26** (see. FIG. 2) selected by the conversation manager **28** based on the context **70.**”), 0034 (“When a successful match is found, the corresponding translation **74** is dispatched to the script engine **38** for execution”).)

- (i) **“communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.” (1[i])**

104. As discussed above for 1[g] and 1[h], Coffman and Ross both describe how an identified application is invoked to process a command/query based on the user’s input. A person of ordinary skill in the art would have understood that, as part of invoking the identified application, the command/query (**“request”**) would be communicated to the application. For example, as noted, Coffman describes how “[t]he first task to have its dialog completed by producing a non-ambiguous query is executed” (Coffman, 37:2-4) and Ross similarly teaches that the identified application “receive[s] the corresponding command” (Ross, ¶0052). A person of ordinary skill in the art therefore would have understood and found it obvious that a command/query (**“request”**) would be communicated to a conversationally-aware application in Coffman (**“the domain agent”**).

105. Coffman further makes clear that a conversationally-aware application (**“domain agent”**) **“is configured to generate a response to the user.”** Coffman discloses that a conversationally-aware application returns output as a result of processing the 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 the user using, for

example, text-to-speech, as Coffman illustrates in Figure 6:

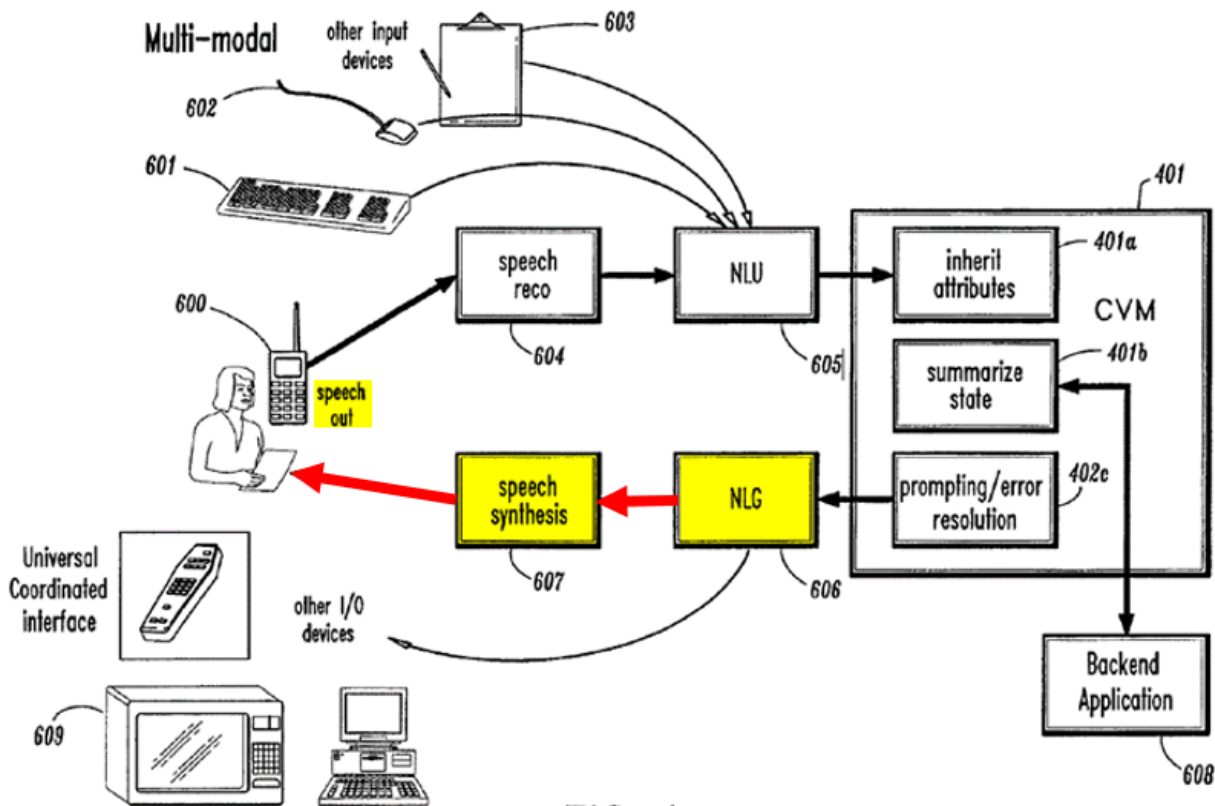


FIG. 6

(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”), 41:23-24 (“[C]omplete and non-ambiguous commands will result in certain results (e.g., outputs or actions).”) 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.”).)

2. Claim 2: “The device of claim 1, wherein the response includes an aggregation of content gathered when the identified domain agent processes the request.”

106. The additional limitation of claim 2—“**wherein the response includes an aggregation of content gathered when the identified domain agent processes the request.**”—would have been obvious over Coffman alone or in view of Kennewick.

107. As discussed above for 1[i], 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 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] includes an aggregation of content gathered when the identified domain agent processes the request.”**

108. The additional limitation of claim 2 would have also been obvious over Coffman in view of Kennewick. As discussed above, 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 to a user that **“includes an aggregation of content gathered when [designated agent[s]] process[] [a] request.”**

109. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis of 1[d], which applies here as well. Moreover, with respect to claim 2 in particular, 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 content 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 content would allow the system to provide more complete and useful responses to the user. 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. (Coffman, e.g., 12:28-30 & 57:23-27 (describing conversationally aware applications (“agents”); Kennewick, e.g., ¶¶0088-0094 (describing “agents”).)

3. Claim 3: “The device of claim 1, wherein the multi-modal natural language input is a first multi-modal natural language input and is received during a first session, the one or more processors further configured to receive a second multi-modal natural language input from another user during the first session.”

110. Coffman explains that its conversational computing system “allows multiple users while actively running” and describes “multi-user” discourses, which Coffman describes as “conversational interaction[s] associated with a given task/process/thread.” (Coffman, 47:25-26 (“Conversational VM can adapt dynamically to the preferences of multiple users and to the active context. It allows multiple users while actively running.”), 47:27-48:7 (“Any change of user automatically implies the creation of a new active context which pushes the previous

context down the context stack, unless the new active context is waived explicitly by the new user or the active application....Exceptions to this rule while traversing the context stack may automatically imply that the discourse becomes flagged as multi-users.”), 29:22-23 (“activated discourse (i.e., conversational interaction associated with a given task/process/thread)”).) A person of ordinary skill in the art would have understood a discourse, as a conversational interaction associated with a given task/process/thread, would correspond to a “**session.**” (EX1009, p.475 (defining “session” as including “**1.** The time during which a program is running. In most interactive programs, a session is the time during which the program accepts input and processes information. **2.** In communications, the time during which two computer maintain a connection.”).) By describing multi-user discourses—i.e., multiple users participating in a single discourse—a person of ordinary skill in the art would have understood and found it obvious for Coffman’s conversational system and user device (such as a PVA or PC (and its processors)) to be “**configured to receive a second multi-modal natural language input from another user during the first session.**” And, as I explain in my analysis of claim 4 below, to the extent there is any question whether Coffman discloses multiple users interacting with the same during the same session, it would have been obvious in view of Kennewick, which expressly discloses sessions in which multiple users interact with

the system. (Kennewick, ¶0045.) Claim 3 therefore would have also been obvious over Coffman in view of Kennewick.

4. Claim 4: “The device of claim 1, wherein the one or more processors are configured to support interactions with a plurality of users during an interleaved session.”

111. The '607 patent describes an “**interleaved session**” as a session in which multiple users access the system in an “interleaved” manner during a same session. ('607, 8:20-23 (“According to another embodiment of the invention, the system may support multiple users that access the system during a same session, in an interleaved or overlapping manner.”).)

112. As with claim 3, the additional limitations of claim 4 would have been obvious over Coffman. Claim 4 is similar to claim 3, but broader because it recites “supporting interactions” instead of “receiving multi-modal natural language input.” As discussed above for claim 3, Coffman discloses “multi-user discourses”—i.e., sessions in which multiple users may provide system input. A person of ordinary skill in the art would have understood and found it obvious that, in the course of taking part in a multi-user discourse, users would naturally interact with the system in an “interleaved” manner.

113. Moreover, it would have been obvious in view of Kennewick for Coffman’s conversational system to support “interleaved sessions.” Kennewick

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

expressly discloses “interleaved sessions,” providing a description that, as shown below, appears in substantially similar form in the ’607 patent:

Kennewick, ¶0045	’607 Patent, 8:20-34
Multiple users may use the system at different times or during interleaved or overlapping sessions.	[T]he system may support multiple users that access the system during a same session, in an interleaved or overlapping manner.
The system recognizes the user either by name or voice. Once the user is recognized, the system invokes the correct profile.	The system may recognize the multiple users by name, voice, or other characteristic and may invoke a correct profile for each user.
If multiple users are addressing the system in overlapping or interleaved sessions, the system determines which user is stating each question or command and applies the correct profile and context.	If multiple users are addressing the system in overlapping or interleaved sessions, the system may identify the multiple users and may invoke one or more corresponding profiles.
For applications requiring security, the user is verified, typically by using voiceprint matching or requesting a password or pass-phrase from the user.	For applications that require security safeguards, the multiple users may be verified using voiceprint matching, password or pass-phrase matching, or other security safeguards.
When multiple users are engaged in interleaved sessions, the system gracefully resolves conflicts using a probabilistic or fuzzy set decision method.	When multiple users are engaged in interleaved sessions, the system may gracefully resolve conflicts using a probabilistic or fuzzy set decision method for each user.

114. As shown above, by contrasting an “interleaved session” with multiple users using the system at different times, and providing substantially the same descriptions of how the system operates during an interleaved session, Kennewick

makes clear that an “interleaved session,” just as in the ’607 patent, is a session where users interact with the system in an interleaved manner during the same session.

115. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis for 1[d] above, which applies here as well. With respect to claim 4, it would have further been obvious to combine Kennewick’s teachings regarding an “interleaved session” with Coffman, predictably resulting in “[t]he device of claim 1, wherein the one or more processors are configured to support interactions with a plurality of users during an interleaved session,” as claimed. As discussed above, Coffman already teaches that multiple users can interact with the conversational system during the same discourse. Kennewick simply confirms that the interaction can be “interleaved.” Moreover, to the extent there is any question whether Coffman discloses multiple users interacting with the same during the same session, it would have been obvious in view of Kennewick. A person of ordinary skill in the art would have readily appreciated that enabling multiple users to interact during the same session would beneficially allow for more flexible and natural interactions with the system. A person of ordinary skill in the art would have therefore been motivated to incorporate Kennewick’s teachings regarding interleaved sessions into Coffman’s conversational system.

5. **Claim 5: “The device of claim 1, wherein the cognitive model is a first cognitive model, wherein the one or more processors are configured to generate the speech-based transcription based on a second cognitive model that includes information on one or more interactions between the device and a plurality of users of the device.”**

116. As discussed above for 1[d], the written description of the '607 patent describes a “personalized cognitive model **810**.” ('607, 24:1-11.) In addition to that model, the '607 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.

('607, 24:12-21.)

117. 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,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

¶0149.) As discussed above for claim 1[d], 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 device and a plurality of users of the device”** because Kennewick explains that its system supports multiple users in a given session—i.e., 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.)

118. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis for 1[d] and applies here as well. With respect to claim 5, 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.

119. In **Ground 2**, I discuss how generating a speech-based transcription based on a "**second cognitive model**" would also have been obvious in further view of **Lee** (EX1006).

6. **Claim 6: "The device of claim 5, wherein the one or more processors are configured to generate the speech-based transcription further based on an environmental model that has information indicative of an environment of the user."**

120. With respect to an "**environmental model**," the written description of the '607 patent states that an "environmental model" may include "information

associated with the user's environment and surroundings" and makes clear that it encompasses a broad range of information:

According to one embodiment of the invention, the environmental model **808** may include information associated with the user's environment and surroundings. The information may include the type of environment that a user is in (e.g., quiet or noisy); details of a microphone and/or speaker system; the user's current global position and movement, as may be determined by GPS; current system status, such as what song/movie is playing, is the system in the midst of retrieving something, or other system status; details on all voice-enabled devices in the immediate vicinity, such as a presence of a voice-enabled TV, stereo, and DVD player in the same room; a user's credit card information, such as numbers and current balances, wherein the user may ask a mobile telephone to download and pay for a video and the system may respond that there are insufficient funds; or other information. The information may be accessed to invoke a context, domain knowledge, preferences, and/or other cognitive qualities that enhance the interpretation of questions and/or commands.

('607, 24:22-39.)

121. An "**environmental model**" as claimed would have been obvious in view of Kennewick. Kennewick explains "[t]he accurate recognition and parsing of the user's speech requires the best possible signal to noise ratio at the input to the speech recognition engine **120**." (Kennewick, ¶0138.) Kennewick describes using

an “adaptive filter” in order “to optimize the signal to noise ratio at the input to the speech recognition engine **120**.” (Kennewick, ¶0141.) Kennewick explains that the adaptive filter has “band-shaping parameters” that are optimized using “[a]lgorithms comparing the background noise to the signal received from the users speech.” (Kennewick, ¶0141.)

122. Kennewick therefore discloses an “**environmental model**” in the form of an adaptive filter and its band-shaping parameters. Consistent with the written description and claim 6, Kennewick’s environmental model “**has information indicative of an environment of the user**” because, as Kennewick explains, the band-shaping parameters are optimized by using the user’s environment—i.e., by comparing the background noise of the signal received from the user’s speech. Kennewick further teaches that an adaptive filter and its band-shaping parameters are used in generating a speech-based transcription, by explaining that the adaptive filter is used to provide the best possible speech input for the speech recognition engine to process. (Kennewick, ¶¶0138, 0141.) Therefore, the output of the speech recognition engine—i.e., the speech-based transcription—is “**generated...based on an environmental model.**”

123. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis of 1[d] above and applies here as well. With respect to

claim 6, it would have further been obvious to incorporate Kennewick's teachings about using an adaptive filter into Coffman's conversational system. The combination would have predictably resulted in Coffman's conversational system processing spoken input from the user using an adaptive filter to optimize the input for speech recognition and generating a speech-based transcription. The combination therefore would have disclosed **"the one or more processors are configured to generate the speech-based transcription further based on an environmental model that has information indicative of an environment of the user"** as claimed.

124. Kennewick provides express motivation to combine. As noted, Kennewick explains that "[f]iltering and noise elimination may be important in facilitating the various functionalities of the system 90" and that [t]he accurate recognition and parsing of the user's speech requires the best possible signal to noise ratio at the input to the speech recognition engine 120." (Kennewick, ¶0138.) Kennewick further explains that one way to optimize signal to noise ratio is to apply an adaptive filter as explained above. (Kennewick, ¶0141.) With respect to Coffman, a person of ordinary skill in the art would have recognized that the devices in the system would operate in a variety of different and dynamic sonic environments and therefore would have appreciated the benefits of using an adaptive filter. A

person of ordinary skill in the art, therefore, would have been motivated to incorporate Kennewick's adaptive filter teachings into Coffman's conversational system (or at least found it obvious to try) to improve a similar system in the same way.

7. Claim 7: “The device of claim 6, wherein the environmental model has information indicative of how noisy the environment of the user is.”

125. My analysis of claim 6 addressed claim 7. A person of ordinary skill in the art would have understood that Kennewick's adaptive filter and band-shaping parameters (“**environmental model**”) would have “**information indicative of how noisy the environment of the user is.**” This is because, as explained, the parameters would be optimized using “[a]lgorithms comparing the background noise to the signal received from the users speech.” (Kennewick, ¶0141.)

8. Claim 8: “The device of claim 1, wherein the one or more processors are configured to receive one or more subsequent multi-modal inputs, the subsequent multi-modal inputs including at least one of a subsequent natural language utterance or a subsequent non-speech input.”

126. The limitations of claim 8 describe nothing more than the normal, expected use of a multi-modal conversational system, and are disclosed and obvious over Coffman.

127. A person of ordinary skill in the art would have understood and found

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

it obvious that Coffman's multi-modal conversational system, as a system for user(s) to carry out back-and-forth dialog with a computer system, would receive one or more subsequent multi-modal inputs, including both natural language utterances and non-speech input. (Coffman, 4:6-8 ("The conversational computing paradigm prescribes that systems dialog with a user to complete, disambiguate, summarize or correct queries and the result of their executions."), 6:6-8 ("[T]he 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).").) For example, Coffman explains that "[m]ixed initiative allows a user to naturally complete, modify, or correct a request via dialog with the system." (Coffman, 11:17-18.) In fact, Coffman provides a lengthy example of an ongoing interaction between a user and the conversational system that includes at least "subsequent natural language utterances," such as "grocery list" and "it should be synchronized with my PDA!":

You can instruct the PVA to dial in your PC using the phone. Once the connection is established, you are authenticated by voice and you find by voice the memo by simply requesting the "grocery list" that you had previously created, without having to remember the file name or the directory or browse through your directory to eventually select the appropriate file. You may need to confirm the selection if your PC CVM requests it. You can issue another query – "it should be

synchronized with my PDA! - and the file is appropriately linked to be transferred to your PDA at the next synchronization. One last command – “Do it!” - and your PC gives up and lets the PVA handle that ambiguous query. The PVA understands your desire to synchronize the PDA and the PC based on your previous conversation. After possible confirmation, the synchronization is performed and the grocery list is stored on your PDA, ready for later use. 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. At any time, you can request navigation information about where you are, what to do next, how far to go or even request a different itinerary. Pressed by time, you instruct the PVA to dial the store drive-through server. This may involve an intermediate dialog with a directory assistance service IVR. Once connected to the store IVR, an illustration of the concept of a small business or personal consumer IVR built similarly to current home pages, through the dialog with its conversational interface, you place your order. For this, you ask the PVA to slowly browse through the grocery list and read it to you item by item. You then rephrase the request to the IVR and pursue the dialog until each order is appropriately taken.

By the time you reach the store, your order is ready. You can now drive home and while driving quietly listen to your e-mail or check the news or stock quotes. If needed, you can dial in your PC to consult or modify some spreadsheet information; the same way that you would have

consulted it by voice on your desktop while processing your mail.

(Coffman, 56:26-57:20.)

9. Claim 9: “The device of claim 1, wherein the identified domain agent is configured to update the context stack in response to processing the request.”

128. Claim 9 is obvious over Coffman. Coffman discloses that a conversationally-aware application can update a context stack in response to processing a request, explaining for example, that “[e]ach action, completed query and conversational argument of an active task as well as each returned value/result are stored on the context stack 405.” (Coffman, 40:27-29; *see also id.*, 30:6-8 (“Each context associated with a given discourse comprises the latest requests made to the corresponding process/task/thread as well as the latest output.”), 21:23-26 (“A context stack 217 is managed by the dialog manager 219. The context stack 217 comprises all the information associated with an application. 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.”).)

10. Claim 10: “The device of claim 1, wherein the one or more processors are configured to synchronize the context stack with a context stack of another device.”

129. Claim 10 is obvious over Coffman. Coffman explains that its

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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.)

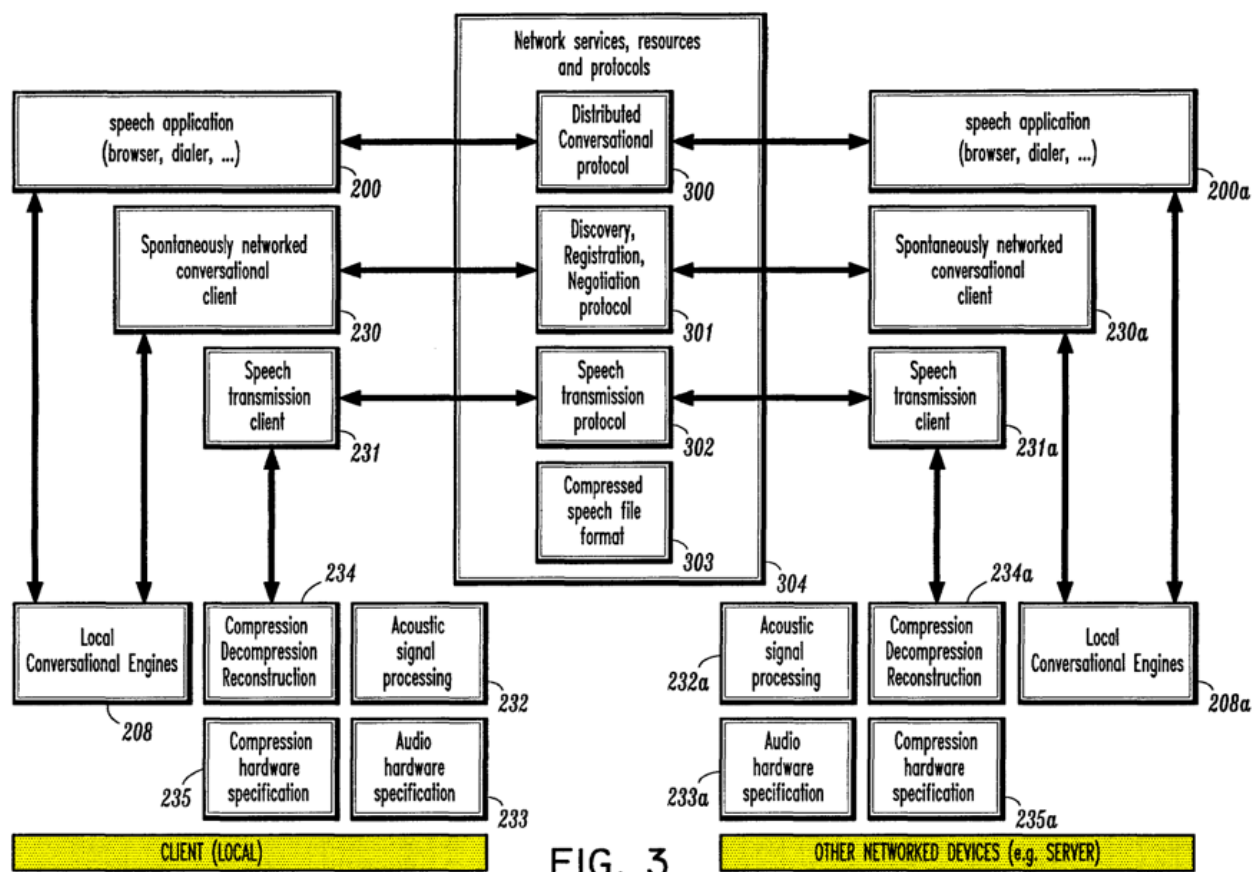


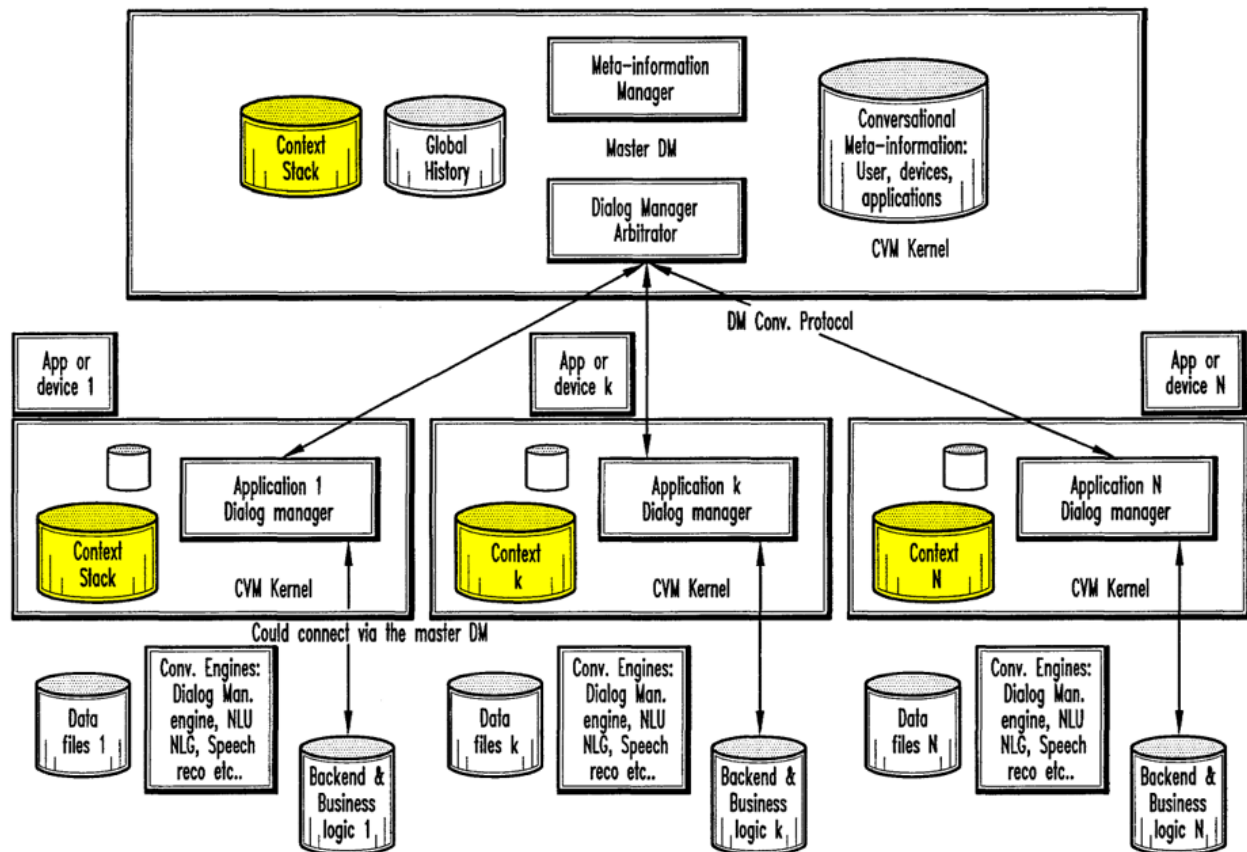
FIG. 3

(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”:

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 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).”).)

130. Coffman further makes clear that a device’s context stack can be “synchronized” with a context stack of another 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.)

131. A person of ordinary skill in the art would have therefore further understood and found it obvious for a user’s device (such as PC or PVA) in Coffman’s conversational system to be **“configured to synchronize the context stack with a context stack of another device.”**

11. Claim 11: “The device of claim 10, wherein the one or more processors are configured to synchronize the cognitive model with a cognitive model of another device.”

132. As discussed above for claim 1, Coffman and Kennewick disclose the claimed **“cognitive model”** in the form of user profile information. With respect to claim 11, Kennewick further discloses that user profile information can be synchronized between devices:

The system may be used in conjunction with a wide range of platform environments. These environments include, as a peripheral to a PC, a portable or wearable computer or other computer, embedded in a PC or

other computer, on a personal digital assistant (e.g., a PDA with wireless networking capability, in a specialized device such as a bar code scanner or payment terminal, on a wireless telephone, or other platforms....A user may use the system on one or more devices. In this case history and profile information for the user may be synchronized between the multiple devices on a periodic basis or other basis.

(Kennewick, ¶0043.)

133. I discussed the rationale and motivation to combine Coffman and Kennewick in my analysis of 1[d] above, which applies here as well. With respect to claim 11, it would have obvious to a person of ordinary skill in the art to combine Kennewick's teachings about synchronizing profile information with the conversational system of Coffman and Kennewick discussed above for claim 1. The combination would have predictably resulted in the conversational system as described, further configured to synchronize a user's profile information ("**the cognitive model**") with a user profile ("**a cognitive model**") of another device. Kennewick provides express motivation to combine, explaining that "[a] user may use the system on one or more devices." (Kennewick, ¶0043.) A person of ordinary skill in the art would have understood that the ability to synchronize user profile information across devices would beneficially enable the system to provide a more consistent and predictable experience for the user.

12. Independent Claim 12

134. As shown below, independent claim 12 recites the method counterpart to claim 1 (underlining added to indicate differences):

Claim 1	Claim 12
1. A <u>device</u> for processing natural language inputs, comprising <u>one or more processors configured to</u> :	12. A <u>method</u> for processing natural language inputs, comprising:
receive a multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;	<u>receiving, by one or more processors, a</u> multi-modal natural language input from a user, the multi-modal natural language input including a natural language utterance and a non-speech input;
generate a non-speech transcription from the non-speech input;	<u>generating, by the one or more processors, a</u> non-speech transcription from the non-speech input;
identify the user who provided the multi-modal natural language input;	<u>identifying, by the one or more processors, the user who provided the</u> multi-modal natural language input;
generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;	<u>generating, by the one or more processors, a</u> speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;
generate a merged transcription from the speech-based transcription and the non-speech transcription;	<u>generating, by the one or more processors, a</u> merged transcription from the speech-based transcription and the non-speech transcription;
identify, from among a plurality of entries that are in a context stack and	<u>identifying, by the one or more processors, an entry in a context stack,</u>

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

Claim 1	Claim 12
that are each indicative of context, <u>an entry in the context stack that matches information in the merged transcription</u> ;	from among a plurality of entries that are in the context stack and that are each indicative of context, <u>wherein the identified entry matches information in the merged transcription</u> ;
identify a domain agent associated with the entry in the context stack;	<u>identifying, by the one or more processors</u> , a domain agent associated with the entry in the context stack;
<u>determine</u> a request based on the merged transcription; and	<u>determining, by the one or more processors</u> , a request based on the merged transcription; and
<u>communicate</u> the request to the domain agent, wherein the domain agent is configured to generate a response to the user.	<u>communicating, by the one or more processors</u> , the request to the domain agent, wherein the domain agent is configured to generate a response to the user.

135. As shown in the table above, the preamble of claim 1 recites “one or more processors,” while claim 12 recites “one or more processors” in each step. This is not a meaningful difference because, as explained for claim 1, it would have been obvious for the “one or more processors” (such as processors in a PC or PVA) to perform each of the recited steps. Therefore, claim 12 is disclosed and obvious for the same reasons as claim 1.

13. Claim 13: “The method of claim 12, wherein the cognitive model is a first cognitive model, and wherein the generating the speech-based transcription is based on a second cognitive model that includes information on one or more interactions between the device and a plurality of users of the device.”

136. Claim 13 is substantially the same as claim 5, which similarly recites: “The device of claim 1, wherein the cognitive model is a first cognitive model, wherein the one or more processors are configured to generate the speech-based transcription based on a second cognitive model that includes information on one or more interactions between the device and a plurality of users of the device.” Therefore, for the reasons discussed for claim 5, claim 13 is also disclosed by and obvious over the prior art. In **Ground 2**, I discuss how generating a speech-based transcription based on a “**second cognitive model**” would also have been obvious in further view of **Lee** (EX1006).

14. Independent Claim 14:

137. The side-by-side table below lists claim 1 and independent claim 14, with underlining added to indicate differences:

Claim 1	Claim 14
1. A device for processing natural language inputs, comprising one or more processors configured to:	14. A device for processing natural language inputs, comprising one or more processors configured to:
receive a <u>multi-modal</u> natural language <u>input</u> from a user, <u>the multi-modal natural language input including a</u>	receive a natural language <u>utterance</u> from a user;

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

Claim 1	Claim 14
<u>natural language utterance</u> and a <u>non-speech input</u> ;	
generate a non-speech transcription from the non-speech input;	[blank]
identify the user who provided the <u>multi-modal natural language input</u> ;	identify the user who provided the <u>natural language utterance</u> ;
generate a speech-based transcription based on a cognitive model associated with the user, wherein the cognitive model includes information on one or more prior interactions between the user and the device;	generate a speech-based transcription based on a <u>personal</u> cognitive model associated with the user and a <u>general cognitive model</u> , wherein the <u>personal</u> cognitive model includes information on one or more prior interactions between the device and the user, and wherein the <u>general cognitive model includes information on one or more prior interactions between the device and a plurality of users</u> ;
generate a merged transcription from the speech-based transcription and the non-speech transcription;	[blank]
identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry in the context stack that matches information in the <u>merged</u> transcription;	identify, from among a plurality of entries that are in a context stack and that are each indicative of context, an entry that matches information in the <u>speech-based</u> transcription;
identify a domain agent associated with the entry in the context stack;	identify a domain agent associated with the entry in the context stack;
determine a request based on the <u>merged</u> transcription; and	determine a request based on the <u>speech-based</u> transcription; and
communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.	communicate the request to the domain agent, wherein the domain agent is configured to generate a response to the user.

138. As shown above, claim 14 does not recite limitations related to “non-speech.” For example, claim 14 recites only “receiv[ing] a natural language utterance” while claim 1 recites “receiv[ing] a multi-modal natural language input” that includes “a non-speech input.” Similarly, claim 14 recites a “speech-based transcription” rather than a “merged transcription” generated from a speech-based transcription and a non-speech transcription.

139. While claim 14 omits claim 1’s limitations related to “non-speech,” it adds limitations related to “general cognitive model” that are substantially similar to limitations related to a “second cognitive model” recited in claim 5. Like the “second cognitive model” of claim 5, which “includes information on one or more interactions between the device and a plurality of users of the device,” the “general cognitive model” of claim 14 “includes information on one or more prior interactions between the device and a plurality of users.”

140. As discussed above for claim 5, the “second cognitive model” corresponds to the collection of other users’ profile information, and the claimed “prior interactions” include activities 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.) With respect to claim 5, therefore, the claimed “**personal**

cognitive model” corresponds to the specific user’s profile information, and the **“general cognitive model”** corresponds to the collection of other users’ profile information.

141. Therefore, for the reasons discussed above for claim 1 and claim 5, claim 14 is disclosed by and obvious over the prior art. In **Ground 2**, I discuss how generating a speech-based transcription based on a **“general cognitive model”** would have also been obvious in further view of **Lee** (EX1006).

15. Claim 15: “The device of claim 14, wherein the one or more processors are configured to generate the speech-based transcription further based on an environmental model, wherein the environmental model includes information indicative of how noisy the environment of the user is.”

142. Claim 15 is an amalgam of claims 6 and 7, which together recite “generat[ing] the speech-based transcription...based on an environmental model” (claim 6) that “has information indicative of how noisy the environment of the user is” (claim 7). Therefore, for the reasons discussed for claims 6 and 7, claim 15 is disclosed by and obvious over the prior art.

C. Ground 2: Claims 5-7, 13-15 Are Obvious Over Ground 1 Prior Art In Further View Of Lee.

143. As discussed above in my Ground 1 analysis, claims 5, 13 and 14 recite using a **“second cognitive model”** or **“general cognitive model”** to generate a speech-based transcription. The **“second cognitive model”** as recited in claims 5

and 13 “**includes information on one or more interactions between the device and a plurality of users of the device,**” and the “**general cognitive model**” similarly “**includes information on one or more prior interactions between the device and a plurality of users.**” I explained that Kennewick discloses the claimed “**second cognitive model**” and “**general 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.)

144. Lee offers more detail and provides 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, U.S. Patent Application Publication No. 2002/0087315, entitled “Computer-Implemented Multi-Scanning Language Method and System,” 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.)

145. **Rationale and Motivation to Combine (Lee with Ground 1 Prior Art).** 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, Kennewick and Ross described in connection with Ground 1. 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 generating a speech-based transcription using a "**second cognitive model**" (claims 5 and 13) and "**general cognitive model**" (claim 14) as claimed.

146. Lee is an analogous reference in the same field of speech recognition and processing as the '607 patent. ('607, 1:20-25; Lee, Abstract, ¶0002.) Lee would have also been reasonably pertinent to problems facing the '607 patent inventors, including techniques for improving speech recognition accuracy, including in multi-user systems.

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

148. 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 ’607 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

149. 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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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

150. 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.

151. 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

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,447,607 B2

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.
