

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION

IPR2025-00655

U.S. Patent No. 7,640,160

**DECLARATION OF DR. HENRY HOUH
UNDER 37 C.F.R. § 1.68 IN SUPPORT OF
PETITION FOR *INTER PARTES* REVIEW**

TABLE OF CONTENTS

	Page
I. INTRODUCTION	8
II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE	10
III. LEVEL OF ORDINARY SKILL IN THE ART	16
IV. RELEVANT LEGAL STANDARDS	17
V. OVERVIEW OF THE 160 PATENT	22
A. The Disclosures of the 160 Patent.....	22
VI. CLAIM CONSTRUCTION	24
VII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE.....	25
VIII. THE PRINCIPAL PRIOR ART	25
A. Maes	25
B. Ross	27
C. The Combination of Maes (with Coffman) and Ross	29
IX. CLAIMS 12 AND 13 ARE UNPATENTABLE	36
A. Ground 1: Maes in view of Ross renders obvious claims 12 and 13.....	36
1. Invalidity Analysis of claims 12 and 13.....	36
a. Independent Claim 12.....	36
i. [12.0] A method for interpreting natural language utterances using knowledge-enhanced speech recognition engine, wherein the knowledge-enhanced speech recognition engine is configured to determine an intent and correct false recognitions of the natural language utterances, comprising:	36
ii. [12.1] receiving a transcription of a natural language utterance at a computer comprising the knowledge-enhanced speech recognition engine;	54
iii. [12.2.0] identifying one or more contexts that completely or partially match one or more	

	text combinations contained in the transcription,.....	63
iv.	[12.2.1] wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;	73
v.	[12.3] scoring each of the identified matching contexts;.....	88
vi.	[12.4] selecting the matching context having a highest score to determine a most likely context for the utterance; and.....	91
vii.	[12.5] communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.	92
2.	Independent Claim 13	101
i.	[13.0] A method for processing natural language utterances, comprising:	101
ii.	[13.1] receiving a natural language utterance at a computer comprising a multi-pass speech recognition module;.....	101
iii.	[13.2] transcribing the utterance using the multi-pass speech recognition module, the multi-pass speech recognition module configured to transcribe the utterance into text;	104
iv.	[13.3.0] identifying one or more contexts that completely or partially match one or more text combinations contained in the text of the transcribed utterance, ...	105

v.	[13.3.1] wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;	105
vi.	[13.4] scoring each of the identified matching contexts;.....	105
vii.	[13.5] selecting the matching context having a highest score to determine a most likely context for the utterance; and.....	105
viii.	[13.6] communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.....	105
B.	Ground 2: Maes, Coffman and Ross Render Obvious claims 12 and 13	106
X.	OBJECTIVE INDICIA OF NON-OBVIOUSNESS	106
XI.	CONCLUSION.....	107
	CLAIM LISTING	108

EXHIBIT LIST

No.	Exhibit Description
1001	U.S. Patent No. 7,640,160
1002	File History of U.S. Patent No. 7,640,160
1003	Declaration of Dr. Henry Houh
1004	CV of Dr. Henry Houh
1005	U.S. Patent No. 6,964,023 (“Maes”)
1006	RESERVED
1007	Second Amended Complaint
1008	RESERVED
1009	RESERVED
1010	RESERVED
1011	RESERVED
1012	International Patent Application Publication No. WO 00/20962 (“Coffman”) filed as PCT/US99/22927
1013	U.S. Patent No. 7,137,126 (“ <u>Coffman126</u> ”)
1014	RESERVED
1015	RESERVED
1016	RESERVED
1017	Excerpts from Dafydd Gibbon et al., “Handbook of Multimodal and Spoken Dialogue Systems : Resources, Terminology, and Product Evaluation” (2000) (“Gibbon”)
1018	RESERVED
1019	RESERVED
1020	U.K. Patent Application GB2162347A
1021	D. Walters “Deterministic Context-Sensitive Languages: Part I*” (“Walters”), INFORMATION AND CONTROL 17, 14-40 (1970)
1022	U.S. Patent Application Publication No. 2002/0133354 (“ <u>Ross</u> ”)
1023	M. Mao et al. “Automatic training set segmentation for multi-pass speech recognition,” IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005
1024	Hetherington, I. L. (2005, September). A multi-pass, dynamic-vocabulary approach to real-time, large-vocabulary speech recognition. In <i>INTERSPEECH</i> (pp. 545-548).
1025	RESERVED
1026	Excerpts from Microsoft Computer Dictionary, 5 th edition (2002)

1027	RESERVED
1028	RESERVED
1029	EDTX Calendar, Judge Gilstrap

TABLE OF ABBREVIATIONS AND CONVENTIONS

Abbreviation	Meaning
160 Patent	Ex.1001: U.S. Patent No. 7,640,160
IPR	<i>inter partes</i> review
Petitioner	Microsoft Corporation (“Microsoft”)
Patent Owner or PO	Dialect, LLC (“Dialect”)
First Amended Infringement Contentions	IC: Ex.1007
<i>xx.yy–zz</i>	column <i>xx</i> , lines <i>yy</i> to <i>zz</i>

I, Henry Houh, declare as follows:

I. INTRODUCTION

1. I am making this declaration in support of Petitioner’s request for an *Inter Partes* Review of U.S. Patent No. 7,640,160 (“the 160 Patent”) to Cristo.

2. I am being compensated for my work in this matter at my standard hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this proceeding. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

3. I have been asked to provide my opinions regarding whether the subject matter of claims 12 and 13 (“the Challenged Claims”) of the 160 Patent are anticipated or otherwise would have been obvious to a person having ordinary skill in the art (“POSITA”) at the time of the alleged invention, in light of the prior art. It is my opinion that the Challenged Claims would have been obvious to a POSITA.

4. In the preparation of this declaration, I have studied:

- the 160 Patent, Ex.1001
- the prosecution history of the 160 Patent (“160 File History”), Ex.1002
- U.S. Patent No. 6,964,023 (“Maes”), Ex.1005
- International Patent Application Publication No. WO 00/20962 (“Coffman”), Ex.1012

- Patent Owner's Second Amended Complaint, Ex.1007
- U.S. Patent Application Publication No. 2002/0133354 ("Ross"), Ex.1022
- the additional references I refer to below and/or identified in the exhibit list.

5. In forming the opinions expressed below, I have considered: the documents listed above; the relevant legal standards, including the standard for obviousness; and my own knowledge and experience based upon my work in the field of network communications and security as described below.

6. Unless otherwise noted, all **emphasis** in any quoted material has been added. Claim terms are italicized.

7. Since I was retained, I have been in frequent contact with counsel while working on this matter to communicate my opinions. My opinions are presented both in this Declaration and in the Petition for Inter Partes Review, which it supports. In many, if not most, cases the language used to express my opinions in both documents is substantially the same. This is because I worked with counsel to capture my opinions, and generally speaking we captured them in the Petition itself as opposed to developing both the Petition and Declaration in tandem. My opinions, as reflected in the Petition we prepared, were then incorporated into my Declaration.

I make this statement to explain the work process and underscore that the opinions in this Declaration are my own, were formed by me and were documented with the assistance of counsel in the manner described. As detailed herein, it is my opinion that each of the challenged claims is rendered obvious by prior art references that predate the earliest priority date of the 160 Patent. If requested, I am prepared to testify about my opinions expressed herein.

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

8. I am over the age of 18 and am competent to write this declaration. I have personal knowledge, or have developed knowledge, of these technologies based upon education, training, or experience, of the matters set forth herein.

9. My complete qualifications and professional experience are described in my Curriculum Vitae, a copy of which can be found in Ex.1004. The following is a brief summary of my relevant qualifications and professional experience.

10. I received a Ph.D. in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology (“MIT”) in 1998. Beforehand, I received a Master of Science degree in Electrical Engineering and Computer Science in 1991, a Bachelor of Science degree in Electrical Engineering and Computer Science in 1989, and a Bachelor of Science degree in Physics in 1990, all from MIT.

11. I am currently self-employed as an independent technical consultant. Until 2022, I was also president of a company that provides supplemental science,

technology, engineering, and mathematics (“STEM”) education to children of all ages.

12. I first entered telecommunications in 1987 when I worked as a summer intern at AT&T Bell Laboratories as part of a five-year dual degree program at MIT. I continued to work at AT&T Bell Laboratories as part of this MIT program. While at MIT, I was a teaching assistant (“TA”) in the Electrical Engineering and Computer Science Department’s core Computer Architectures course. I first was a TA as a senior for a role typically reserved for graduate students. I later became head TA. The course covered various topics in computer architectures. As a TA, I helped write homework assignments, lab assignments, and exams. I also taught in the recitation sections.

13. Later, as part of my doctoral research at MIT from 1991-1998, I was a research assistant in the Telemedia Network Systems (“TNS”) group at the Laboratory for Computer Science. The TNS group built a high-speed gigabit network and created applications that ran over the network. Example applications included ones for remote video capture, processing, and display of video on computer terminals. In addition to working on the design of core network components, designing and building the high-speed links, and designing and writing the device drivers for the interface cards, I also set up the group’s web server.

14. I also helped to build the web pages that initiated the above- mentioned video sessions via a web interface. Vice President Al Gore visited our group in 1996 and received a demonstration of—and remotely drove—a radio- controlled toy car with a wireless video camera mounted on it that was built by our group. This toy car device received commands transmitted over a network from a remote computer, and video data from the toy car was transmitted wirelessly then over a computer network back to the user controller. On occasion, we allowed users visiting our web site to drive the toy car from their remote computer while they watched the video on their computer. The video stream was encoded by TNS-designed hardware, streamed over the TNS-designed network, and displayed using TNS-designed software.

15. I defended and submitted my Ph.D. thesis, titled “Designing Networks for Tomorrow’s Traffic,” in January 1998. As part of my thesis research, I analyzed local area and wide area flows to show a more efficient method for routing packets in a network, based on traffic patterns at the time.

16. From 1997 to 1999, I was a Senior Scientist and Engineer at NBX Corporation, a start-up that made business telephone systems for streaming packetized audio over data networks instead of using traditional telephone lines. NBX was later acquired by 3Com Corporation and the phone system is still used today by numerous businesses.

17. As part of my work at NBX, I designed the core audio reconstruction algorithms for the telephones, as well as the packet transmission algorithms. I also designed and validated the core packet transport protocol used by the phone system. The protocol was used for all signaling in the phone system, including for the setup of conference calls.

18. The NBX system also featured a computer interface for initiating phone calls, which could also initiate conference calls. The NBX system also supported the Telephony Application Programming Interface (“TAPI”) that allowed other computer programs to integrate with our system telephony features. We obtained U.S. Patent No. 6,967,963, entitled “Telecommunication method for ensuring on-time delivery of packets containing time-sensitive data,” as part of this work. During my time at NBX, I also evaluated and purchased VPN equipment in order to demonstrate that our system worked over long distances through VPN devices, so that our phones could be located remotely. We used the VPN equipment to connect from various trade shows, for example, in Las Vegas and Los Angeles, remotely into our home network so that phones in the remote network were able to connect to our home facility in Andover, MA.

19. From 1999-2004, I was employed by Empirix or its predecessor company, Teradyne. Empirix was a leader in test tools for telecommunications protocols and systems, providing functional testing tools as well as load testing tools.

From 2000-2001, I conceived and built a test platform for testing Voice- over-IP (VoIP). The first application on this new test platform was a cloud emulator for simulating the effects of transmitting VoIP over a busy network.

20. In 2006, as part of my role at BBN Technologies, I helped found PodZinger Inc., now known as RAMP Inc. PodZinger utilized BBN's speech recognition algorithms to search through the spoken words in audio and video segments. While I was Vice President of Operations and Technology, PodZinger followed its initial prototype with a full streaming audio and video search solution. I also created a social networking web site, which BBN sold to a venture-funded startup company. In the process of creating the web site, I designed and specified the authentication and authorization protocols.

21. I have been awarded several United States patents, and I have several patent applications pending including the following examples:

- U.S. Patent No. 7,975,296, "Automated security threat testing of web pages";
- U.S. Patent No. 7,877,736, "Computer language interpretation and optimization for server testing";
- U.S. Patent No. 7,801,910, "Method and apparatus for timed tagging of media content";

- U.S. Patent 7,590,542, “Method of generating test scripts using a voice-capable markup language”;
- U.S. Patent No. 6,967,963, “Telecommunication method for ensuring on-time delivery of packets containing time-sensitive data”;
- U.S. Patent Application Publication No. 20070106685, “Method and apparatus for updating speech recognition databases and reindexing audio and video content using the same”;
- U.S. Patent Application Publication No. 20070106693, “Methods and apparatus for providing virtual media channels based on media search”;
- U.S. Patent Application Publication No. 20070106760, “Methods and apparatus for dynamic presentation of advertising, factual, and informational content using enhanced metadata in search-driven media applications”;
- U.S. Patent Application Publication No. 20070112837, “Method and apparatus for timed tagging of media content”;
- U.S. Patent Application Publication No. 20070118873, “Methods and apparatus for merging media content”; and
- U.S. Patent Application Publication No. 20090222442, “User-directed navigation of multimedia search results.”

III. LEVEL OF ORDINARY SKILL IN THE ART

22. My opinions are provided based on what a person of ordinary skill in the art (“POSITA”) in the technical field of the invention would have understood at the time of the purported invention of the 160 Patent. I have been asked to assume the date of the purported invention for the 160 patent is August 5, 2005.

23. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

24. Based on the materials and information I have reviewed, and on my experience in the technical areas relevant to the Asserted Patents, a POSITA with respect to the Asserted Patents as of the time of the invention, i.e., the earliest possible priority date of the 160 Patent, August 5, 2005, would have had a bachelor’s degree in electrical engineering, computer science, computer engineering, or an equivalent, and two years of relevant experience involving computer science fundamentals, including natural language processing, speech recognition and transcription, non-speech recognition and transcription that is pertinent to the 160 Patent. Lack of professional experience can be remedied by additional education, and vice versa.

25. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my own experience and what a POSITA would have understood or known generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the relevant field as of the priority date of the 160 Patent.

IV. RELEVANT LEGAL STANDARDS

26. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the 160 Patent, I am relying on certain basic legal principles that counsel has explained to me.

27. I understand that prior art to the 160 Patent includes patents and printed publications in the relevant art that predate the priority date of the 160 Patent. For purposes of this Declaration, I am applying August 5, 2005 as the priority date of the 160 Patent, but I take no position on that issue.

28. I understand and have been informed that in order for an inventor to be entitled to a patent, the invention must be “new.” There are a number of ways that an invention can be “anticipated,” that is, not new.

29. A patent claim is anticipated if each limitation of the patent claim is found either expressly or inherently in a single item of prior art. While a prior art reference need not use the same words as a patent claim to anticipate the claim, the prior art reference must describe the requirements of the claim with sufficient clarity

such that POSITA would have been able to make and use the claimed invention based on the reference and his or her knowledge in the applicable technical field.

30. The disclosure of a feature of a claimed invention can be either “express,” meaning that the text, figures, or other content of a reference actually teaches the aspects of the patent claim, or the disclosure can be “inherent.” An express disclosure does not need to use the same words as the patent claim being considered, and does not need to be depicted in exactly the same way, but the disclosure must actually convey the claimed invention to POSITA using some combination of words, figures, data, or other portions of the patent. I have been informed and understand that in order to establish that an element of a claim is “inherent” in the disclosure in a prior art reference, it must be clear to a person skilled in the art that the missing element is an inevitable part of what is explicitly described in the reference, and that it would have been recognized as necessarily present by a person skilled in the art. Inherency cannot be established by probabilities or possibilities, and the mere fact that something may result from a given set of circumstances is not enough to establish inherency.

31. I further understand that to anticipate a claim, the elements of the claim must be found in the prior art reference “arranged as in the claim.” I have been informed that this means that the elements of the prior art reference may not be recombined in a way that is not taught in the prior art to arrive at the invention;

instead, the prior art must teach the claimed subject matter in the arrangement required by the claims.

32. I understand that if a reference incorporates other documents by reference, the incorporating reference and the incorporated reference(s) should be treated as a single prior art reference for purposes of analyzing anticipation.

33. I understand that it is acceptable to consider evidence other than the information in a particular prior art document to determine if a feature is necessarily present in or inherently described by that reference.

34. I have been informed and understand that subject matter claimed in a patent is obvious under 35 U.S.C. § 103 if a person of ordinary skill in the art at the time the alleged invention was made would have had reason to combine or modify the disclosures of one or more prior art references to arrive at the claimed subject matter. A claim may be unpatentable even if each claim limitation is not present or disclosed in a single prior-art item.

35. I have been informed and understand that, under the doctrine of obviousness, a claim is unpatentable if the differences between the invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. A person of ordinary skill in the art is presumed to have knowledge of the relevant prior art at the time of the claimed invention.

36. I have been informed and understand that obviousness is based on the scope and content of the prior art, the differences between the prior art and the claim, the level of ordinary skill in the art and secondary indicia of obviousness and nonobviousness to the extent such indicia exist. I have been advised that the following evidence, when present, must be considered before forming a conclusion that a claimed invention is obvious: (1) the invention's commercial success, (2) long felt but unresolved needs, (3) the failure of others, (4) skepticism by experts, (5) praise by others, (6) teaching away by others, (7) recognition of a problem, and (8) copying of the invention by competitors. In this instance, however, I am not aware of any evidence of any of these factors. I will of course consider any evidence on these issues which is presented to me.

37. I have been informed and understand the determination of whether the asserted claims would have been obvious to a person of ordinary skill in the art and, therefore, invalid, is not governed by any rigid test or formula. A determination that a claim is obvious is, instead, based on a common-sense determination that the claimed invention is merely a combination of known limitations to achieve predictable results. Any of the following rationales are acceptable justifications to conclude that a claim would have been obvious: (1) the claimed invention is simply a combination of known prior art methods to yield predictable results; (2) the claimed invention is a simple substitution of one known element for another to

obtain predictable results; (3) the claimed invention uses known techniques to improve similar devices (methods, or products) in the same way; (4) the claimed invention applies a known technique to a known device (method, or product) ready for improvement to yield predictable results; (5) the claimed invention was “obvious to try”—choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; (6) there is known work in one field of endeavor that may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations would have been predictable to one of ordinary skill in the art; or, (7) there is some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill in the art to modify the prior art reference to combine prior art teachings to arrive at the claimed inventions.

38. I have been informed and understand that an analysis of whether a claimed invention is obvious must not rely on a hindsight combination of prior art. The analysis must proceed in the context of the time of the invention or claimed priority date and consider whether the invention as a whole would have been obvious to a person of ordinary skill in the art, taking into consideration any interrelated teachings of the prior art, the effects of demands known to the design community or present in the marketplace, and the background knowledge possessed by a person having ordinary skill in the art, all in order to determine whether there was an

apparent reason to combine any known elements in the fashion claimed by the patent at issue.

V. OVERVIEW OF THE 160 PATENT

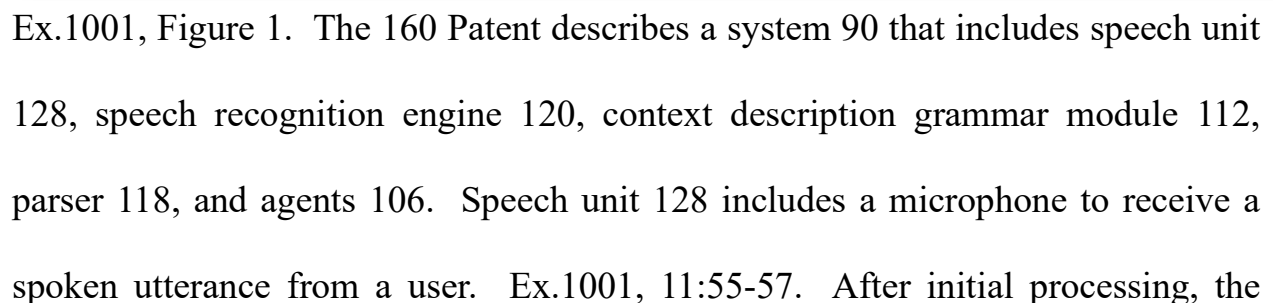
39. The 160 Patent was based on U.S. Patent Appl. No. 11/197,504 filed on August 5, 2005. Ex.1001, Cover.

40. I have been asked to assume that it has an earliest priority date of August 5, 2005.

A. The Disclosures of the 160 Patent

41. The 160 Patent explains that it “relates to retrieval of information or processing of commands through a speech interface and/or a combination of a speech interface and a non-speech interface” and that “[m]ore specifically,” it “provides a fully integrated environment that allows users to submit natural language questions and commands via the speech interface and the non-speech interface. Information may be obtained from a wide range of disciplines, making local and network inquiries to obtain the information and presenting results in a natural manner, even in cases where the question asked or the responses received are incomplete, ambiguous or subjective.” Ex.1001, 1:8–18. The 160 Patent states that it “creates, stores and uses extensive personal profile information for each user, thereby improving the reliability of determining the context of the speech and non-

42. Figure 1 of the 160 Patent (reproduced below) “is an overall diagrammatic view according to one embodiment of the invention” (Ex.1001, 9:64–65):



speech is passed to speech recognition engine 120 for processing using the context description grammar module 112. Ex.1001, 12:23-25. 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.” Ex.1001, 12:25-28. “The knowledge agents may then process the commands or questions.” Ex.1001, 12:32-36. To select the correct agent to process the command or question, “[t]he parser 118 uses a scoring system to determine the most likely context or domain for a user’s question and/or command. The score is determined from weighing a number of factors including, the user profile 110, the domain agent’s data content and previous context. Based on this scoring, the system 90 invokes the correct agent,” which returns results to the user. Ex.1001, 21:10-15. 12:41-42.

VI. CLAIM CONSTRUCTION

43. It is my understanding that in order to properly evaluate Challenged Claims, the terms of the claims must first be interpreted. It is my understanding that for the purposes of this inter partes review, the claims are to be construed under the so-called Phillips standard, under which claim terms are given their ordinary and customary meaning as would have been understood by a POSITA in light of the specification and prosecution history, unless the inventor has set forth a special

meaning for a term. I have also been informed that claim terms only need to be construed to the extent necessary to resolve the unpatentability inquiry.

VII. IDENTIFICATION OF HOW THE CLAIMS ARE UNPATENTABLE

44. The discussion in this Declaration provides a detailed analysis of how the asserted prior art teaches each limitation of the Challenged Claims.

45. As part of my analysis, I have considered, and discuss in detail, the scope and content of the prior art and any differences between the alleged invention and the prior art.

46. It is my opinion that the alleged invention recited in the Challenged Claims would have been obvious in view of the teachings of the asserted prior art and the knowledge of a POSITA, as I discuss in detail below.

VIII. THE PRINCIPAL PRIOR ART

A. Maes

47. U.S. Patent No. 6,964,023 to Maes et al. (Ex.1005) was filed on February 5, 2001, and issued on November 8, 2005. I understand that it is prior art to the 160 Patent.

48. Maes explains that although “multi-modal systems would appear to have inherent advantages over systems that use only one data input mode, the existing multi-modal techniques fall significantly short of providing an effective conversational environment between the user and the computing system with which

the user wishes to interact.” Ex.1005, 1:60–63. Maes thus describes “systems and methods are provided for performing focus detection, referential ambiguity resolution and mood classification in accordance with multi-modal input data, in varying operating conditions, in order to provide an effective conversational computing environment for one or more users.” Ex.1005, Abstract.

49. In particular, Maes explains that “the multi-modal conversational computing system 10 of the present invention receives multi-modal input in the form of audio input data, video input data, as well as other types of input data (in accordance with the I/O subsystem 12), processes the multi-modal data (in accordance with the I/O manager 14), and performs various recognition tasks (e.g., speech recognition, speaker recognition, gesture recognition, lip reading, face recognition, etc., in accordance with the recognition engines 16), if necessary, using this processed data. The results of the recognition tasks and/or the processed data, itself, is then used to perform one or more conversational computing tasks, e.g., focus detection, referential ambiguity resolution, and mood classification (in accordance with the dialog manager 18, the context stack 20 and/or the classifier 22).” Ex.1005, 4:7–22.

50. Moreover, Maes expressly incorporates by reference Coffman in its entirety. In particular, Maes states that “[i]t is to be appreciated the conversational virtual machine disclosed in PCT patent application identified as PCT/US99/22927

filed on Oct. 1, 1999 and entitled “Conversational Computing Via Conversational Virtual Machine,” [*i.e.*, Coffman or Ex.1012] the disclosure of which is incorporated by reference herein, may be employed to provide a framework for the I/O manager, recognition engines, dialog manager and context stack of the invention.” Ex.1005, 9:30–37. I therefore understand that Coffman is considered effectively part of Maes.

51. As the entire disclosure of Coffman is directed to “Conversational Computing Via Conversational Virtual Machine,” I understand that Maes is incorporating with detailed particularity the entire disclosure of Coffman. At a minimum, the disclosures related to the “conversational virtual machine” that “provide a framework for the I/O manager, recognition engines, dialog manager and context stack,” are expressly incorporated with particularity, which is the material I rely upon below.

B. Ross

52. U.S. Pub. No. 2002/0133354 to Ross et al. was filed on August 16, 2001 and published on September 19, 2002. I understand that it is prior art to the 160 Patent.

53. As the title suggests, Ross entitled “System and Method for Determining Utterance Context in a Multi-Context Speech Application.” describes techniques for determining a context associated with a user’s spoken command or question to determine an application to invoke to process the command or question.

Ex.1022, [0010] and [0013]. In Ross, determining a context involves evaluating the user's recognized spoken command against grammars for applications in which the grammars describe potential contexts (e.g., keywords or phrases) related to the utterance. Ex.1022, [0033]-[0034]. Ross teaches "test[ing]...against the active grammars" and finding a successful "match" involves matching text combinations in a user's transcribed utterance to grammar expressions in a grammar. Ross explains that "a grammar is defined for each application 26" and describes two example grammars, one for an electronic mail application and another for a calendar application. Ex.1022, [0035], [0040], and [0046]. Ross describes how identifying context(s) involves matching transcribed text to the grammar expressions in the grammars: "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." Ex.1022, [0052]; *see also* [0053] ("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.")

C. The Combination of Maes (with Coffman) and Ross

54. To the extent one asserts that Coffman discloses a different embodiment from that of Maes, a POSITA would have been motivated to combine Coffman's teachings related to searching a context stack to find entries matching transcribed user input and merging/combining speech and non-speech transcriptions (Ex.1012, 41:2–5, 42:1–3, and 38:1–19) with Maes's teachings that receives and processes multi-modal inputs (e.g., by producing decoded text or script) as described below.

55. As explained above, Maes expressly incorporates by reference Coffman in its entirety, names Stephane Maes as a common inventor, and has IBM as the original assignee. Ex.1005, Abstract, 9:30–37 and 19:43–49 and Ex.1012, Cover, Abstract. Accordingly, a POSITA would have been motivated to combine the teachings of Coffman with those of Maes for at least these reasons.

56. Additionally, it would have been obvious to combine Ross, another IBM reference (Ex.1022, Cover), with Maes (and Coffman) because Ross and Coffman each provide motivation to combine their teachings with those of Maes. For example, Ross's teachings of maintaining a prioritized list of context grammars associated with applications/application programs, comparing a spoken utterance¹¹

¹¹ As used herein, it will be understood that a “spoken utterance” received using Maes's system would encompass both an audio portion (e.g., speech) and non-speech portion of the utterance. Ex.1005, 6:43–47.

to identify matches with context grammars, and selecting applications capable of handling the user's spoken utterance combined with Maes would enhance its ability to determine target applications. Ex.1022, Abstract, [0002], [0013], [0021], [0033]–[0038], [0052]–[0054], [0059]–[0060], and Figure 4 and 6; Ex.1005, 36:54–66.

57. Combining Coffman's teachings of organizing/updating its context stack such that the most-recent context is at the top of the context (Ex.1012, 21, 23–28, 41:1–5, 40:15–19, and 42:7–9) with the teachings of Maes would benefit Maes's system by providing efficient identification and retrieval of the most-recent context, allowing for faster executions for matching/decoding of keywords and improving the responsiveness of Maes's system aligning with its goal of faster executions (Ex.1005, 11:64–12:30, 5:29–34), and would be inline with Ross's goal of providing quick or faster support in connection with determining “target applications” (Ex.1022, [0031] and [0035]–[0036]). A POSITA would have understood that by keeping the context stack updated, e.g., with the most-recent context associated with a user's dialog, would allow an application (“*domain agent*”) to use the most-recent context, which is likely to be relevant to the user's utterance, in determining the user's intent. Ex.1005, 36:64–37:3.

58. A POSITA would be motivated to combine Maes (and Coffman) and Ross because they teach strikingly similar techniques. For example, Ross's teachings of testing/comparing a recognized utterance against data included in a

context list (Ex.1022, [0013], [0034]–[0037], [0053], and Figure 4) is similar to Maes’s technique of associating recognized events against data in a context stack (Ex.1005, 7:60–8:4 and Figures 1 and 8), and Coffman’s technique of searching a context stack to find entries matching transcribed user input (Ex.1012, 42:1-9). Notably, Maes’s context stack disclosures are almost verbatim as Coffman’s. Ex.1012, 21, 23–28 and Ex.1005, 37:51–62; *compare also* Ex.1005, Figure 8B and Ex.1012, Figure 2.

59. Additionally, a POSITA would have understood that Maes’s and Coffman’s “context stack,” which is similar to Ross’s “context list” involves the application of known techniques (e.g., identifying contexts by performing comparisons of spoken utterances against data stored in a context list/ stack) to improve a similar system (e.g., Maes’s system) in the same way. A POSITA would have understood that Ross’s description of priority ordering of its context list (Ex.1022, [0037]) is consistent with the well-known concept of a “stack.” For example, it is well-known that in a stack, the most recent item added to the stack will be the first item considered from the stack in a later operation and termed in the art as “last in, first out” (LIFO) processing a queue. *See* Ex.1026, 305 (dictionary definition for “last in, first out”); *compare with* Ex.1026, 215 (dictionary definition for “first in, first out”). .

60. Maes, Coffman, and Ross are analogous art as the 160 Patent for at least three reasons. *First*, all three references are directed toward the same field of endeavor as the 160 Patent—computer-implemented systems interpreting user utterances or otherwise natural language processing systems. Ex.1001, Abstract, 1:51–61, 3:25–45; Ex.1005, Abstract, 4:7–17; Ex.1012, 1–4 and 29:1–4; and Ex.1022, Abstract, [0021]. *See also* Ex.1005, 7:60–8:4; Ex.1022, [0053].

61. *Second*, Maes, Coffman, and Ross are reasonably pertinent to a problem described by the 160 Patent: an environment for **reliably processing** a user’s language queries, which is a problem of conventional systems identified in the 160 Patent. Ex.1001, 1:35–44; Ex.1005, Abstract, 4:7–17 and 6:39–55; Ex.1012, 4:4–9; Ex.1022, [0033]–[0034] and Figure 5. *Third*, similar to the 160 Patent, Maes, Coffman, and Ross teach the use of **grammars** and **vocabularies** in recognizing speech inputs, which indicate compatibility of their teachings and therefore a POSITA would have had a reasonable expectation of success combining them. Ex.1001, 13:35–38; Ex.1005, 31:15–18, 34:61–67, 33:11–21, 39:59, 41:13–20, and 41:35–38; Ex.1012, 29:21–32, 25:6–9; Ex.1022, [0028], [0033]–[0035], and Figure 4.

62. A POSITA would be motivated to combine Maes (and Coffman), and Ross because to do so would have been the arrangement of old elements (a speech transcription and recognition system comprising modules for speech transcription

and recognition, speech-enabled applications handling a user's spoken utterance input, context grammars comprising keywords/phrases pertaining to applications, context stack storing current and historical data related to user interactions) with each performing the same function it has been known to perform (e.g., recognizing spoken utterances, converting spoken utterances into computer-readable format based on decoding the utterances, matching the decoded text of the utterance against grammars describing potential contexts such as keywords/phrases included in the utterance, searching a context stack to find matches between data stored on the context stack and transcribed user input) and yielding no more than what one would expect from such an arrangement (an improved computer-implemented system interpreting user utterances), as Maes seeks to implement. Ex.1005, 11:62–63, 2:54–67.

63. Furthermore, a POSITA would have been motivated to combine Maes (and Coffman), and Ross because the references make clear that their systems require no specialized hardware or software. In fact, these references teach using conventional, commercially-available systems. *See* Ex.1005, 2:38–53 and 45:50–60, 12:43–49 (“processing ... may be accomplished via any **conventional acoustic information recognition system**”) (emphasis added¹²); Ex.1022, [0002] and [0025]

¹² Emphasis added throughout unless otherwise specified.

(identifying examples of commercially-available speech recognition products);
Ex.1012, Abstract (“The conversational computing system may be built on top of a
conventional operating system and API’s ... and conventional device hardware.”).

64. A POSITA would have had a reasonable expectation of success in combining the teachings of these references because to do so would be a simple combination of analogous teachings that would be well within the skill of a POSITA. For example, modifying Maes’s system producing decoded text to include Coffman’s teachings related to searching a context stack to find entries that match transcribed user input and merging/combining speech and non-speech transcriptions and to further include Ross’s teachings of maintaining a priority list of context grammars for applications would enhance the functionality of Maes’s system and further, as explained below, such modifications would have been simple implementations predictably resulting in a recognition system with improved accuracy Maes seeks to implement (Ex.1005, 11:62–63). For example, in one implementation, a POSITA would be motivated to modify Maes’s “grammar” / “grammar database” to additionally include Ross’s teachings of context grammars (e.g., each context grammar comprising entries such as keywords, phrases, and operators used in an application) yielding Maes’s modified “grammar” / “grammar

database.” Ex.1005, 31:15–18, 34:61–67, and 33:11-21; Ex.1022, [0028] and [0033]-[0035].

65. And in another implementation, for instance, a POSITA would have understood to modify Maes’s context stack (Ex.1005, 37:53–55, 7:60–8:4, 8:37–42, and Figure 1) to additionally include Ross’s context list (Ex.1022, [0013], [0034]-[0035] and [0052]-[0059], and Figure 4) comprising context grammars (such as 70-1, 70-2, 70-3). Thus, Ross’s context grammars (each including appropriate keywords and phrases used in an application) would be implemented within Maes’s context stack yielding Maes’s modified context stack having improved context identification functionality.

66. Moreover, in yet another example implementation, a POSITA would have understood to incorporate Ross’s teachings related to context grammars for applications as an additional stand-alone database/module within Maes’s system. A POSITA would have further understood that any of the above implementations would have yielded a system in which the results of the recognized input/output (I/O) events produced as a result of processing the user’s spoken utterance using Maes’s system would be compared, per Ross, against data (e.g., keywords/phrases used in applications) included in context grammars (for instance, stored within Maes’s modified grammar database, or within Maes’s modified context stack). Ex.1002, [0034]-[0035] and [0053]-[0054].

67. Furthermore, a POSITA would recognize that grammars (such as Backus Naur Form (BNF)) describing context associated with applications was well-known and utilized in the art. Ex.1022, [0013] and [0060]; Ex.1026, 49; *see also* Ex.1021, 15–16 (providing definitions of context-sensitive and context-free grammars)).

IX. CLAIMS 12 AND 13 ARE UNPATENTABLE

A. Ground 1: Maes in view of Ross renders obvious claims 12 and 13.

68. 56. Independent claims 12 and 13 of the 160 Patent are rendered obvious over Maes in view of Ross.

1. Invalidity Analysis of claims 12 and 13

a. Independent Claim 12

- i. [12.0] A method for interpreting natural language utterances using knowledge-enhanced speech recognition engine, wherein the knowledge-enhanced speech recognition engine is configured to determine an intent and correct false recognitions of the natural language utterances, comprising:

69. Maes teaches or suggests this claim element.

70. Initially, the 160 Patent describes “*knowledge-enhanced speech recognition*” as follows: “According to another embodiment of the invention, if a match is not found, or only a partial match is found, between the text message and active grammars, then a knowledge-enhanced speech recognition system may be

used to semantically broaden the search. The knowledge-enhanced speech recognition system may be used to determine the intent of the request and/or to correct false recognitions. The knowledge-enhanced speech recognition may access a set of expected contexts that are stored in a context stack to determine a most likely context.” Ex.1001, 13:60–14:2. Therefore, as described in the 160 Patent, the functionality of a knowledge-enhanced speech recognition system is to determine the intent of the request and/or to correct false recognitions, which is analogously disclosed in Maes as ambiguity resolution or disambiguating the user’s intent, as explained below.

71. First, Maes discloses “[a] method for interpreting natural language utterances.” Maes describes “[s]ystems and methods ... provided for performing focus detection, **referential ambiguity resolution** and mood classification **in accordance with multi-modal input data [in the form of audio input data, video input data, as well as other types of input data]**.” Ex.1005, Abstract (emphasis added), 4:10–11. Maes continues: “The present invention relates to **multi-modal data processing techniques** and, more particularly, to systems and methods for performing focus detection, referential ambiguity resolution and mood classification in accordance **with multi-modal input data**. ... Systems that employ such ‘multi-modal’ input techniques have inherent advantages over systems that use only one

data input mode.” Ex.1005, 1:9–23. Therefore, Maes discloses “[a] *method for interpreting natural language utterances.*”

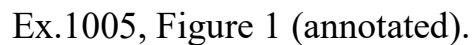
72. Initially, a “*knowledge-enhanced speech recognition engine*” in the context of speech recognition systems was a well-known concept at the time of the 160 Patent. As early as 1998, a patent issued to Komissarchik (entitled “**Knowledge-enhanced speech recognition system and methods** having frame length computed based upon estimated pitch period of vocalic intervals) discloses “knowledge-enhanced speech recognition system and methods ... includ[ing] a series of ... steps which provide real-time analysis and transcription of spoken utterances.” Ex.1018, 8:4–7. Komissarchik describes that its invention includes “**not only grammatical, but also semantic control of the recognition results.**” Ex.1018, 6:45–47; *see also* Ex.1018, 2:60–3:62 (describing several prior art references implementing knowledge-enhanced systems). Furthermore, for the reasons explained below, Maes teaches or suggests “*a knowledge-enhanced speech recognition engine.*”

73. Referencing Figure 1, Maes generally discloses use of one or more recognition engines, e.g., for performing speech recognition:

a master dialog manager may be implemented as part of the dialog manager 18 of FIG. 1, **while the engines would include the one or more recognition engines of FIG. 1.**

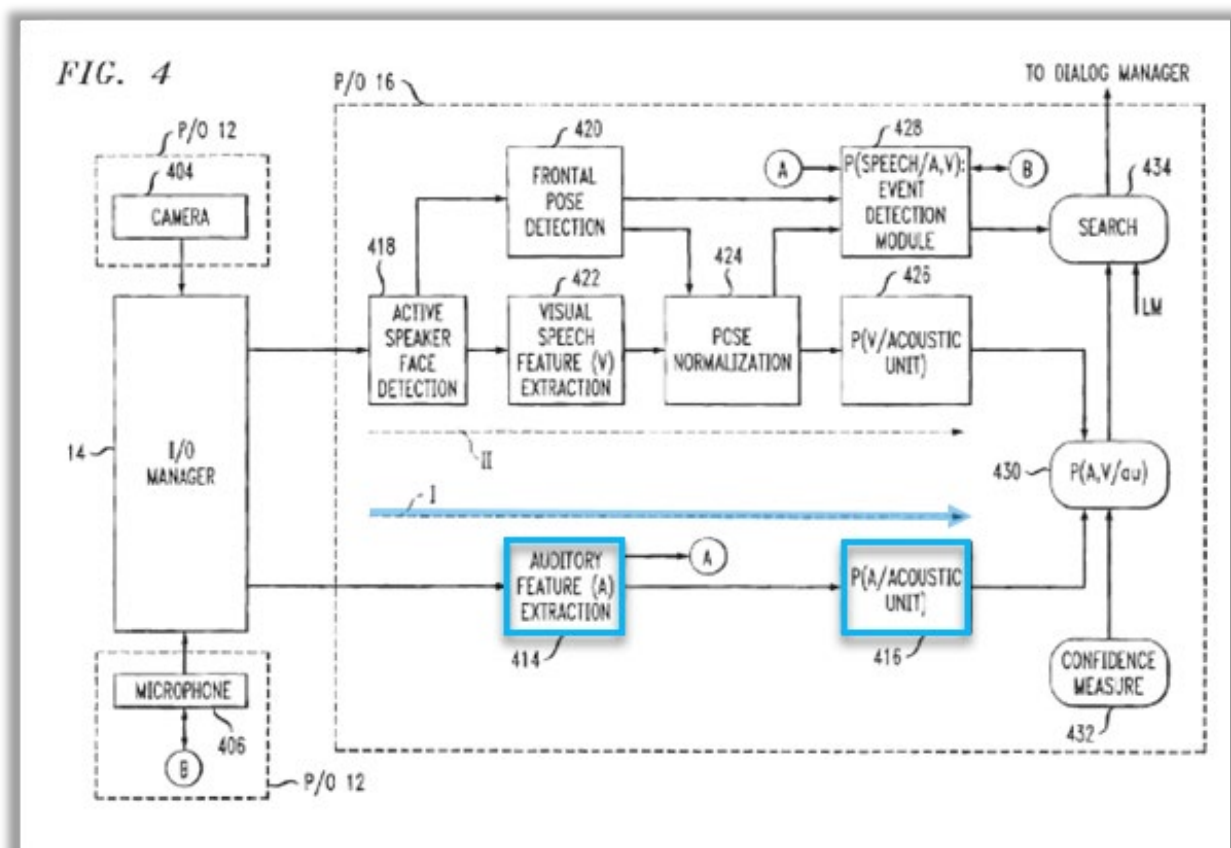
Data files of the foundation classes, as well as data needed by any recognition engine (e.g., grammar, acoustic models, video patterns, etc.), are present on CVM (loadable for embedded platforms or client platforms).

Ex.1005, 34:61–35:1. Maes then describes that its system “**performs various recognition tasks (e.g., speech recognition, speaker recognition, gesture recognition, lip reading, face recognition, etc., in accordance with the recognition engines 16).**” Ex.1005, 4:13–17. “[D]epending on the nature of the [I/O (input/output)] event, **one or more recognition engines may be used to recognize the event. For example, if the event is some form of spoken utterance wherein the microphone picks up the audible portion of the utterance and a camera picks up the visual portion (e.g., lip movement) of the utterance, the event may be sent to an audio-visual speech recognition engine to have the utterance recognized using both the audio input and the video input associated with the speech.**” Ex.1005, 6:42–50.



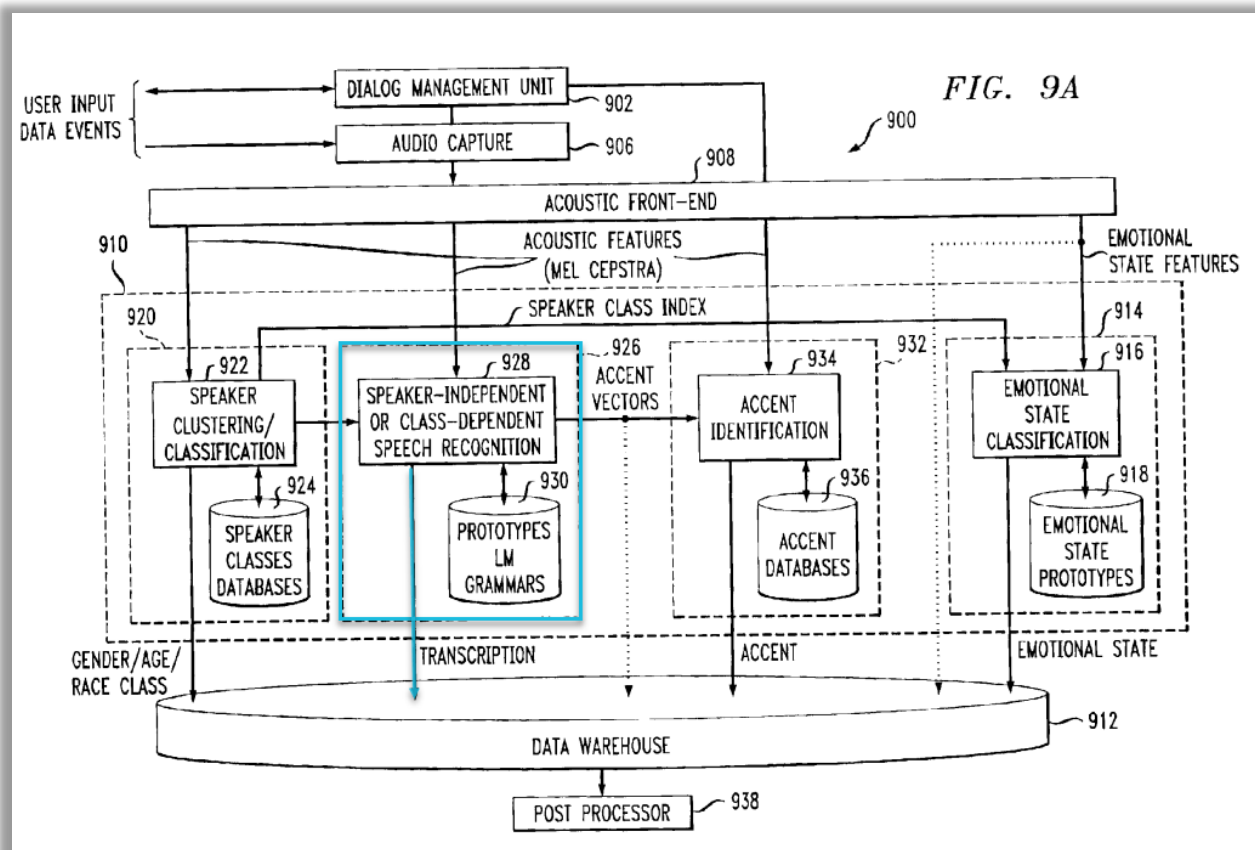
¹³ Maes uses the term “block” and “module” interchangeably. For example, see 12:43-38 and 17:46-50 referring to “414” as “block[] 414” and “module 414.”

utterance”) and a video signal (e.g., the speaker’s face including lip movement and/or background objects in the environment) and processes the multi-modal input data by passing through several processing modules (e.g., the video signal is processed by blocks/modules 418, 422, 424, 426 and the audio signal is processed by blocks/modules 414, 416). Maes states that “the processing performed in blocks 414 and 416 may be accomplished via any conventional acoustic information recognition system [i.e., “*speech recognition engine*”] capable of extracting and labeling acoustic feature vectors, e.g., Lawrence Rabiner, Biing-Hwang Juang, “Fundamentals of Speech Recognition,” Prentice Hall, 1993.” Ex.1005, 12:43–49.



Ex.1005, Figure 4 (showing audio information processing path (Path I) annotated in blue).

75. Referencing Figures 1 and 9A (reproduced below), Maes provides details of “*speech recognition*” functionality explaining that that “via the I/O manager 14 of FIG. 1” “user-provided input data events are ... provided to” “apparatus 900”—an “apparatus for collecting data associated with a voice of a user,” which includes “a dialog management unit 902 for conduct[ing] a conversation with the user,” depicted in Figure 9A. Ex.1005, 39:24–26, Figure 9A. Ex.1005, 39:22–29) “Dialog management unit 902,” Maes explains, can “include natural language understanding (NLU), natural language generation (NLG), finite state grammar (FSG), and/or text-to-speech Syntheses (TTS) for machine-prompting the user in lieu of, or in addition to, the human operator.” Ex.1005, 41:13–20. Maes continues: “Apparatus 900 further includes a processing module 910” which “can further include a speech recognizer 926 which can, in turn, itself include a speech recognition module 928,” providing “*speech recognition*” *functionality* and “a speech prototype, language model and grammar database 930.” Ex.1005, 39:59, 41:35–38.



Ex.1005, Figure 9A (annotated).

76. Maes continues by stating that “Apparatus 900 can further include a **post processor 938** which is coupled to the data warehouse 912 and which is **configured to transcribe user utterances and to perform keyword spotting thereon**. Although shown as a separate item in FIG. 9A, the post processor can be a part of the processing module 910 or of any of the sub-components thereof. For example, it can be implemented as part of the speech recognizer 926. Post processor 938 can be implemented as part of the processor of an IVR, as an application specific circuit, or on a general purpose computer with suitable software modules. Post

processor 938 can employ speech recognizer 926.” Ex.1005, 41:48–59. Therefore, Maes’s system comprises functionality for “*speech recognition*.”

77. Maes further teaches or suggests “*a knowledge-enhanced speech recognition engine*.” Specifically, Maes describes that apparatus 900 which receives “user-provided input data events” “via the I/O manager 14 of FIG. 1 includes a **“semantic module ... to interpret meaning of phrases”** (e.g., “*knowledge-enhanced*”) such that it is **“used by speech recognizer 926 to indicate that some decoding candidates in a list are meaningless and should be discarded/replaced with meaningful candidates.”** Ex.1005, 41:59–64. Ambiguity resolution or disambiguating the user’s intent, which is a goal of Maes (see Ex.1005, Abstract, 4:10–11, and 36:59–63), is handled by the semantic module because the semantic module indicates that some decoding candidates in a list are meaningless and should be discarded/replaced with meaningful candidates. Ex.1005, 41:61–64. These ambiguity resolution disclosures in Maes are analogous to the 160 Patent’s description of “knowledge-enhanced speech recognition system ... used to determine the intent of the request and/or to correct false recognitions.” Ex.1001, 13:64–66. See also Gibbon describing “knowledge-based speech recognition system” as a system that “specifies explicit acoustic-phonetic rules that are robust enough to allow recognition of linguistically meaningful units and that ignore irrelevant variation in these units.” Ex.1017, 429. For at least the reason above, the

semantic module in Maes is “*knowledge-enhanced*” And as explained earlier, Maes’s speech recognition module 928 provides “*speech recognition*” functionality, and, therefore taken together, the speech recognition module 928 and the semantic module, both comprised in Maes’s system, jointly provide functionality that make Maes’s system a “*knowledge-enhanced speech recognition engine.*” Accordingly Maes teaches or suggests that the “*method for interpreting natural language utterances*” is “*using knowledge-enhanced speech recognition engine*”.

78. Indeed, Maes’s system is “*knowledge-enhanced*” because Maes’s provides “**functions needed to adapt ... to the capabilities and constraints of the device, application and/or user preferences,**” uses “a-priori knowledge, ... information about the user,” and Maes teaches training methods based on “**speaker profile features (accent, dialect, gender, age, speaking rate ...**” which are consistent with Patent Owner’s description of “*knowledge-enhanced*” in its Complaint. Ex.1005, 36:44–48, 37:22–28, and 19:7–19; Ex.1007, ¶83 (“*Erica* is also knowledge-enhanced in that it identifies the user and uses knowledge and information relating to the user in interpreting natural language utterances.”).

79. Additionally, for the reasons explained below, the semantic module in Maes is “*knowledge-enhanced*” which therefore makes Maes’s system (comprising “speech recognition functionality) a “*knowledge-enhanced speech recognition engine.*” See generally Ex.1005, 4:1–17.

80. Maes explains that “[a]ll data stored in the data warehouse 912 [which is shown in Figure 9A as coupled to postprocessor 938 that includes the semantic module] can be stored in a format to facilitate subsequent data mining thereon.” Ex.1005, 43:38–41; *see also id.* 39:63–67 (data warehouse 912 ... stores the at least one user attribute, together with at least one identifying indicia, in a form for subsequent data mining thereon.”). Maes further explains that as part of the data mining performed on the data stored in the data warehouse 912, “dialect, socioeconomic classification, and the like can be estimated based on vocabulary and word series used by the user. **Appropriate key words, sentences, or grammatical mistakes to detect can be compiled via expert linguistic knowledge.**” Ex.1005, 43:16–20; *see also* Ex.1005, 41:48–51 (“a post processor 938 [that includes a semantic model] ... coupled to the data warehouse 912 and which is configured to ... perform keyword spotting thereon”), 42:60–61 (“**a keyword spotter to detect insults.**”). The functionality of detecting appropriate key words, sentences, insults, and/or grammatical mistakes fall within the purview of a semantic model/module. *See Gibbon* defining “*semantic[s]*” as “the study of the meaning in language.” Ex.1017, 468. And, as part of the detecting functionality, using “expert linguistic knowledge” (i.e., knowledge defined by human experts in the field) makes clear that Maes’s system, including its semantic model is “*knowledge-enhanced.*” *See e.g.*, Ex.1019, 1157 (McGraw-Hill dictionary defining a “knowledge-enhanced system”

as “[a] computer system whose usefulness derives primarily from a data base containing human knowledge in a computerized format”).

81. Moreover, Maes teaches “modify[ing] behavior of [its] system ... based on at least one user attribute which has been determined based on at least one acoustic feature extracted from the user’s speech” and further that the “[f]eatures can be extracted and **decisions dynamically returned by the models**”—which indicate that the behavior of Maes’s system is modified on-the-fly based on the knowledge (e.g., decisions) incorporated by its models (such as the semantic model), further confirming that Maes’s semantic model is “*knowledge-enhanced*.” Ex.1005, 43:53-54 and 44:7–11. Komissarchik, reflecting the POSITA’s knowledge, proposed knowledge-enhanced speech recognition system and methods which provide real-time analysis and transcription of spoken utterances. Ex.1018, 8:4–14.

82. Moreover, a POSITA would have understood Maes’s models (including its semantic model) and techniques, which include acoustic feature extraction, labeling the extracted vectors with previously stored phonemes, and natural language processing of spoken commands are—“*knowledge-enhanced*” See e.g., Ex.1005, 1:53–57, 2:1–4, and 11:64–12:28. For example, Komissarchik defines “knowledge-based systems” as it applies to “knowledge-enhanced systems” as “involv[ing] the application of acoustic, phonetic and natural language processing (‘NLP’) information theory to the speech recognition process.” Ex.1018, 2:39-43.

83. Therefore, in view of the above-described teachings, Maes teaches or suggests that the “*method for interpreting natural language utterances*” is “*using knowledge-enhanced speech recognition engine.*”

84. Additionally, Maes teaches or suggests “*wherein the knowledge-enhanced speech recognition engine*” is “*configured to determine an intent and correct false recognitions of the natural language utterances.*”

85. Specifically, Maes discloses: “When an abstract [input/output (I/O)] event occurs, **the [dialog manager arbitrator] DMA [in Maes’s system] determines the target of the event and, if needed, seeks confirmation, disambiguation, correction, more details, etc., until the intent is unambiguous and fully determined.**” Ex.1005, 36:59–63. For example, by seeking disambiguation and correction, Maes’s system “*correct[s] false recognitions.*” Maes describes its system includes functionality to “prompt or display for missing, ambiguous or confusing information, asks for confirmation or launches the execution of an action associated to a fully understood multi-modal request from the user) of that application based on the multi-modal input.” Ex.1005, 8:12–16. Therefore, the DMA function in Maes’s system is capable of handling multi-modal I/O events to “**(a) understand the intent of the user; (b) follow up with a dialog to disambiguate, complete, correct or confirm the understanding [of the intent of the user]; (c) or, dispatch a task resulting from fall [sic:full] understanding**

of the intent of the user.” Ex.1005, 36:64–37:3. The above disclosures in Maes are analogous to the 160 Patent’s description of “knowledge-enhanced speech recognition system ... to determine the intent of the request and/or to correct false recognitions.” Ex.1001, 13:64–66..

86. The above disclosures in Maes are analogous to the 160 Patent’s description of “knowledge-enhanced speech recognition system ... to determine the intent of the request and/or to correct false recognitions.” Ex.1001, 13:64–66.

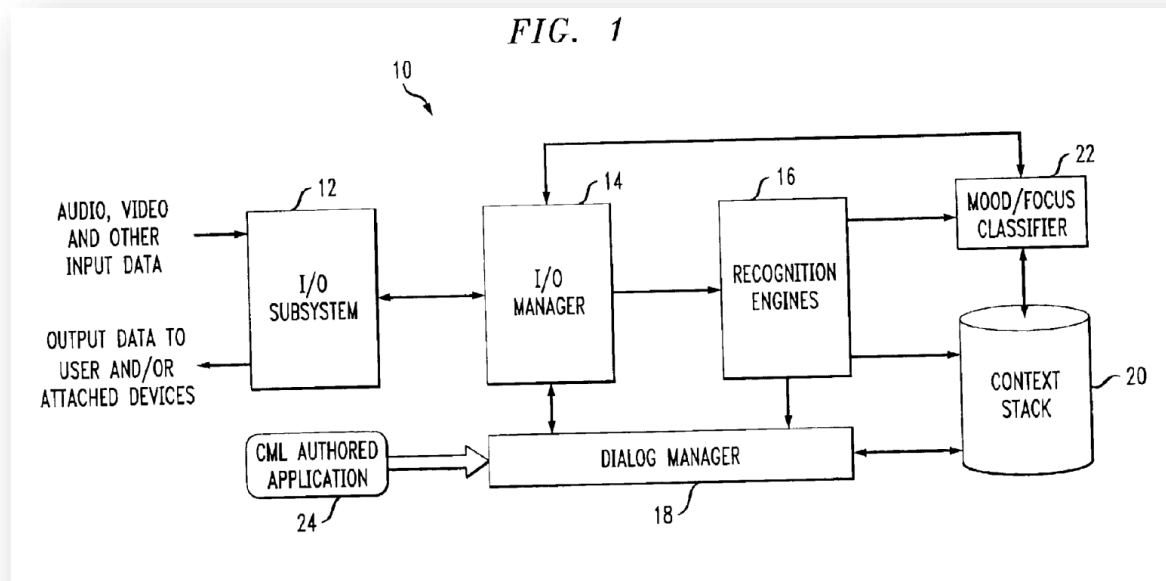
87. Further, Maes provides details of “*interpreting natural language utterances*,” including for instance, a description of how a natural language utterance is received at Maes’s system. While Maes discusses processing multi-modal inputs, the multi-modal input may comprise, at least in part, “audio input data” (*e.g.*, “*natural language utterances*”) and with reference to Figure 1 (reproduced below), Maes discloses:

[A] multi-modal conversational computing system ...
compris[ing] an input/output (I/O) subsystem 12, an I/O
manager module 14, one or more recognition engines 16,
a dialog manager module 18, a context stack 20 and a
mood/focus classifier 22.

**Generally, the multi-modal conversational computing
system 10 of the present invention receives multi-
modal input in the form of audio input data, video input**

data, as well as other types of input data (in accordance with the I/O subsystem 12), processes the multi-modal data (in accordance with the I/O manager 14), and performs various recognition tasks (e.g., speech recognition, speaker recognition, gesture recognition, lip reading, face recognition, etc., in accordance with the recognition engines 16), if necessary, using this processed data. The results of the recognition tasks and/or the processed data, itself, is then used to perform one or more conversational computing tasks, e.g., focus detection, referential ambiguity resolution, and mood classification (in accordance with the dialog manager 18, the context stack 20 and/or the classifier 22).

Ex.1005, 3:66–4:21.



Ex.1005, Figure 1.

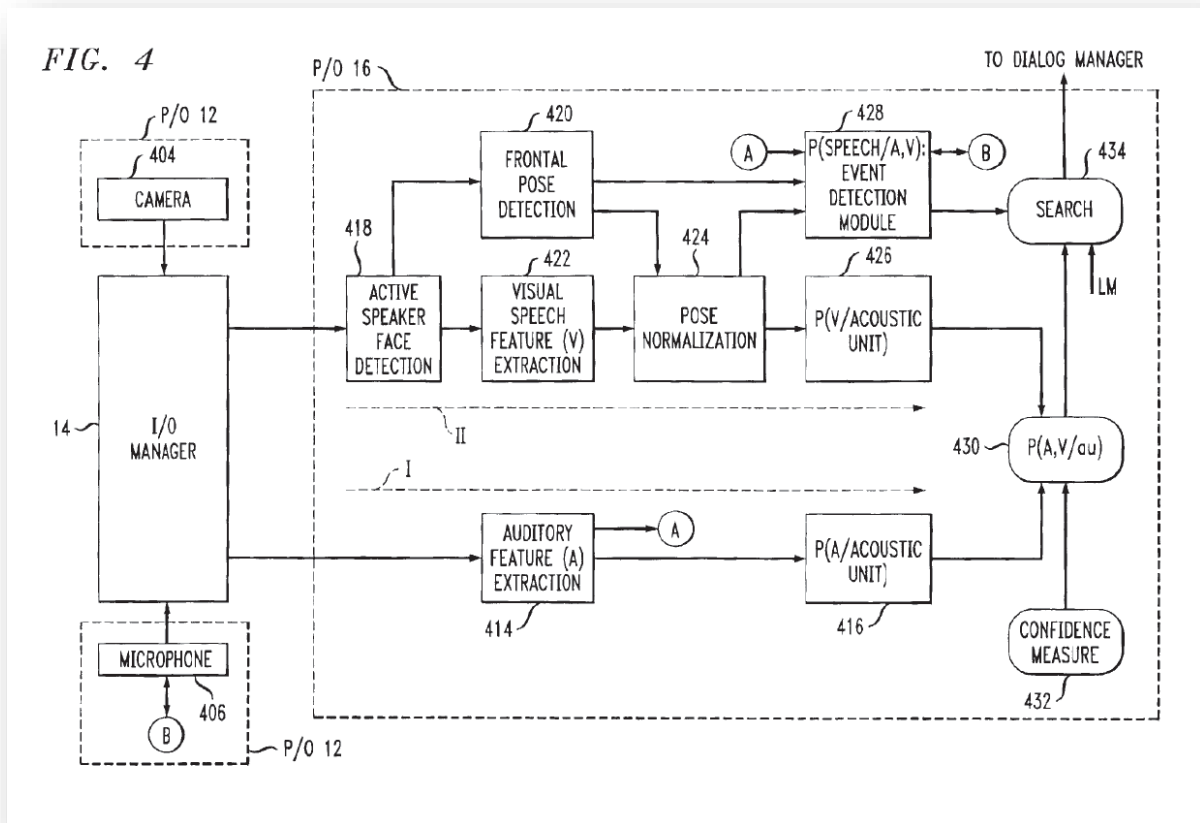
88. With reference to Figures 1 and 4, Maes describes Figure 4 (reproduced below) as “an audio-visual speech recognition module¹⁴ that may be employed as one of the recognition modules¹⁵ of FIG. 1 **to perform speech recognition using multi-modal input data received** in accordance with the invention.” Ex.1005, 10:50–52. Maes notes that the “microphone 406” of Figure 4 is “part of the I/O subsystem 12” (Ex.1005, 11:24–26) which Maes describes that “I/O subsystem 12 may comprise one or more microphones for capturing audio input data from the environment in which the system is deployed.” Ex.1005, 6:5–8. Maes also teaches its system “may comprise one or more cameras or sensors for capturing video input data representing the environment in which the system (or, at least, the I/O subsystem) is deployed.” Ex.1005, 5:60–64.

89. Specifically, Figure 4 shows that the audio-visual speech recognition module receives multi-modal input data—notably, including an audio signal (e.g., speech provided by the speaker and/or background noise) (“*natural language utterances*”) and a video signal (e.g., the speaker’s face including lip movement and/or background objects in the environment) and processes the audio signal and

¹⁴ As used herein, the terms “Maes’s system” and “audio-visual speech recognition module” are used interchangeably.

¹⁵ Maes uses the term “block” and “module” interchangeably. For example, see 12:43-38 and 17:46-50 referring to “414” as “block[] 414” and “module 414.”

the video signal by passing through several processing modules (e.g., the video signal is processed by blocks/modules 418, 422, 424, 426 and the audio signal is processed by blocks/modules 414, 416). Therefore, Maes discloses “*interpreting natural language utterances using ... speech recognition engine.*”



Ex.1005, Figure 4.

90. Maes describes examples in which its methods are applicable to a broadcast news system which uses spoken utterance inputs which further confirms that its method applies to “*interpreting natural language utterances.*” Maes’s

system provides, among others, functions of “natural language understanding (NLU), natural language generation (NLG)” and “NL parsing.” Ex.1005, 41:14–16, 35:58, 10:63–11:15). For example, Maes states that “the audio-visual speech recognition module ... process[es] arbitrary content video [such as] in the context of broadcast news ... [which] contain a newsperson speaking at a location where there is arbitrary activity and noise in the background.” Ex.1005, 10:63–11:15. *See* Gibbon defining “*natural language*” as “a language used by humans for communication.” Ex.1017, 444.

91. As another example, Maes states that “the multi-modal conversational computing system 10 may be employed within a vehicle” in which a user may say the “spoken utterance ‘turn it on’” and Maes’s system processes inputs relating to an I/O event representative of the user’s spoken utterance (audio portion) and the accompanying lip movement of the spoken utterance (video portion). Ex.1005, 4:30–32, 4:46–53, 5:16, 6:43–50, 7:63–8:29.

92. Maes describes an example of how the disclosed system works. For example, Maes discloses its multi-modal conversational computing system is employed within a vehicle. For instance, a user in the vehicle “may say ‘turn it on,’ while pointing at the vehicle radio. Maes’s system receives multi-modal inputs relating to an I/O event representative of the user’s spoken utterance and lip movement corresponding to the spoken utterance. *See* Ex.1005, 6:43–47. For

example, “the microphone picks up the audible portion of the utterance and a camera picks up the visual portion (e.g., lip movement) of the utterance, [and] the event may be sent to an audio-visual speech recognition engine to have the utterance recognized using both the audio input and the video input associated with the speech.” Ex.1005, 6:43–50. “The I/O manager [of the multi-modal conversational computing system] receives the raw multi-modal data and abstracts the data into a form that represents ... a spoken utterance [A] data abstraction operation may involve generalizing details associated with all or portions of the input data ... to yield a more generalized representation of the data for use in further operations.” Ex.1005, 6:35–38. “The dialog manager would therefore receive the results of the recognized events associated with the spoken utterance ‘turn it on’” including the accompanying lip movement. Ex.1005, 7:63–67.

93. Therefore, Maes teaches or suggests this claim element.

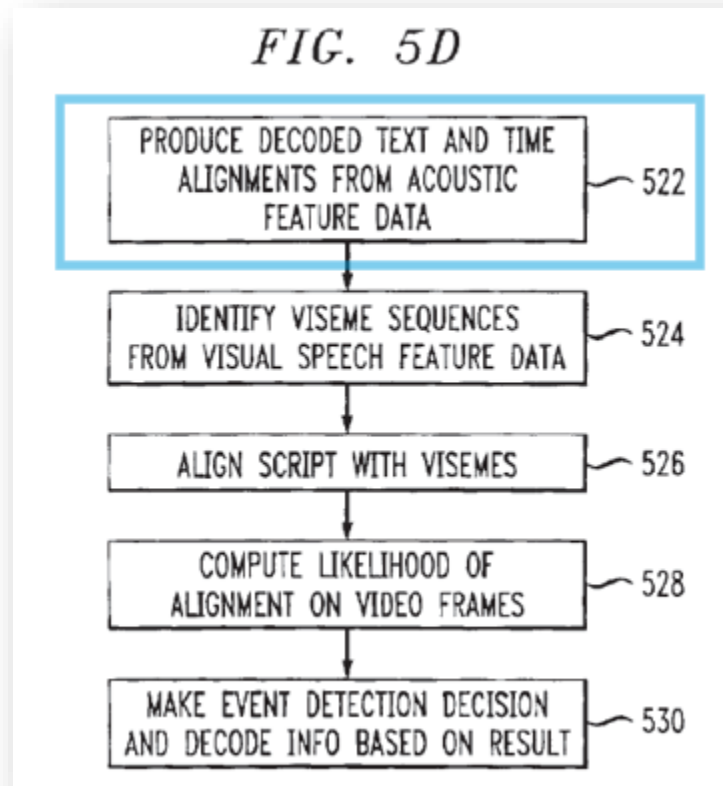
ii. [12.1] receiving a transcription of a natural language utterance at a computer comprising the knowledge-enhanced speech recognition engine;

94. Maes teaches or suggests this claim element.

95. Because “*receiving a transcription of a natural language utterance*” requires “*a transcription of a natural language utterance*” before the transcription is received, the step of creating “*a transcription of a natural language utterance*” is first described below.

96. As explained in [12.0], Maes's system corresponds to a "*knowledge-enhanced speech recognition engine*."

97. Further, with reference to Figure 5D (reproduced below), Maes discloses classical speech recognition techniques producing decoded text (or, script) ("*a transcription of a natural language utterance*") based on processing the spoken utterance using the feature data from the acoustic feature extractor 414 included in an audio-visual recognition module ("*at a computer comprising the knowledge-enhanced speech recognition engine*"). See generally Ex.1005, 21:43–47, Figure 4, and 10:47–48 (identifying Figure 4 as "a preferred embodiment of an audio-visual speech recognition module that may be employed as one of the recognition modules" of Figure 1 to perform speech recognition). In Maes, the outcome of the "*transcription of a natural language utterance*" produces decoded text or script. Therefore, producing decoded text (or, script) based on processing the spoken utterance by an audio-visual speech recognition module serves as "*receiving a transcription of a natural language utterance at a computer comprising the ... speech recognition engine*" and as explained in detail in connection with element [12.0], the ... *speech recognition engine* in Maes is "[a] *knowledge-enhanced speech recognition engine*."



Ex.1005, Figure 5D (block 522). “[I]n step 522, the uttered speech to be verified may be **decoded** by classical speech recognition techniques so that a decoded script and associated time alignments are available. This is accomplished using the feature data from the acoustic feature extractor 414.” Ex.1005, 21:43–47. A POSITA would have understood that the “decoded script” described in Maes is a essentially the tangible result of a speech recognition system’s interpretation of spoken language and provides a text representation of the multi-modal content. *E.g.*, Ex.1005, 20:61-64 (“If the buffered data is tagged as speech, in step 518, the buffered data is sent on

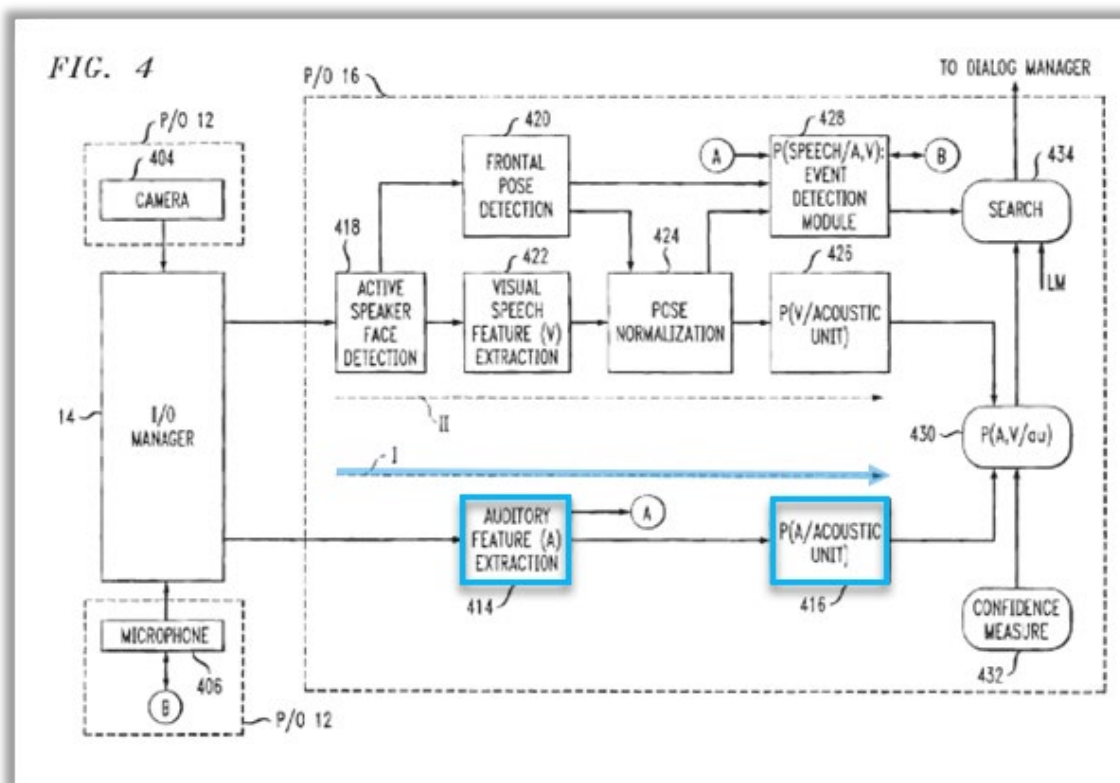
through the acoustic path so that the buffered data may be recognized, in step 520, ... to yield a **decoded output**.”); *see also* 20:23-33.

98. With reference to Figures 1 and 4 (reproduced below), Maes provides additional details explaining the process of “*transcription of a natural language utterance*” as follows: Maes begins by describing Figure 4 as “an audio-visual speech recognition module that may be employed as **one of the recognition modules**¹⁶ of FIG. 1 to perform speech recognition using multi-modal input data **received** in accordance with the invention.” Ex.1005, 10:50–52. Specifically, Figure 4 shows that the audio-visual speech recognition module receives multi-modal input data including an audio signal (e.g., speech provided by the speaker and/or background noise) and a video signal (e.g., the speaker’s face including lip movement and/or background objects in the environment). For example, the auditory feature extractor 414 “receives an audio or speech signal and, as is known in the art, extracts spectral features from the signal at regular intervals. The spectral features are in the form of acoustic feature vectors (signals) which are then passed on to a probability module 416.” Ex.1005, 11:64–12:2. In the audio-visual speech recognition module shown above in Figure 4, the acoustic feature vectors are

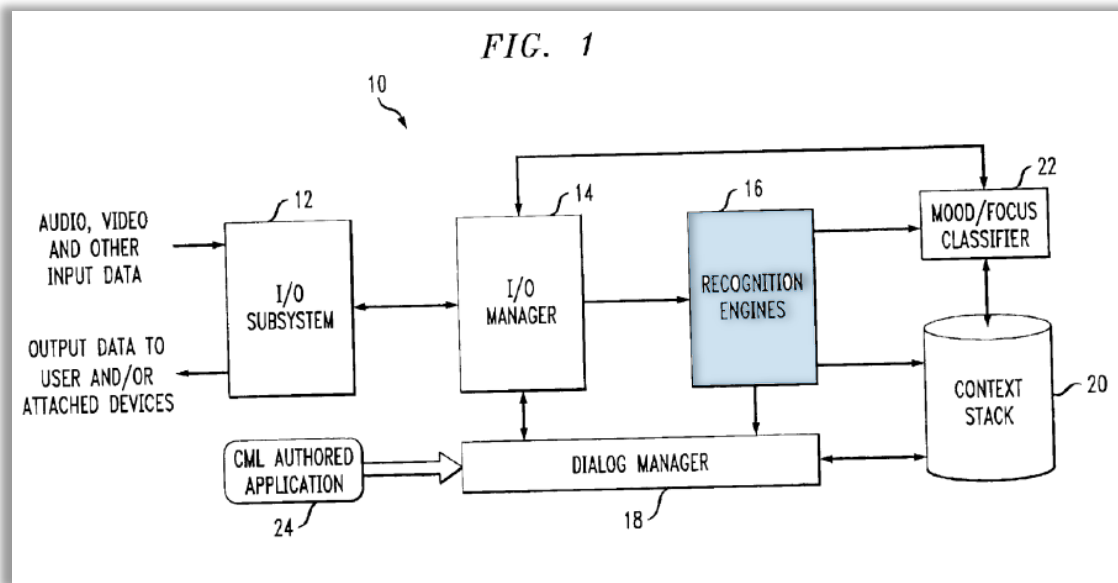
¹⁶ Maes uses the term “block” and “module” interchangeably. For example, see 12:43-38 and 17:46-50 referring to “414” as “block[] 414” and “module 414.”

denoted by the letter “A” and “[a] phantom line denoted by Roman numeral I represents the processing path the audio information signal takes within the module.”

Ex.1005, 11:55–64. As a result of processing the spoken utterance by Maes’s audio-visual speech recognition module (“*the speech recognition engine / module*”), the spoken utterance is transcribed into decoded text (or, simply text).



Ex.1005, Figure 4 (showing audio information processing path (Path I) annotated in blue).



Ex.1005, Figure 1 (annotated).

99. Maes explains how acoustic features are extracted:

[I]n accordance with a preferred acoustic feature extraction process, magnitudes of discrete Fourier transforms of samples of speech data in a frame are considered in a logarithmically warped frequency scale. Next, these amplitude values themselves are transformed to a logarithmic scale. The latter two steps are motivated by a logarithmic sensitivity of human hearing to frequency and amplitude. Subsequently, a rotation in the form of discrete cosine transform is applied. One way to capture the dynamics is to use the delta (first-difference) and the delta-delta (second-order differences) information.

Ex.1005, 12:12–28.

100. Maes further describes sampling the signal prior to extracting the acoustic feature vectors. For example, Maes explains that:

Before acoustic vectors are extracted, the speech signal may be sampled at a rate of 16 kilohertz (kHz). A frame may consist of a segment of speech having a 25 millisecond (msec) duration. In such an arrangement, the **extraction process preferably produces 24 dimensional acoustic cepstral vectors** via the process described below. Frames are advanced every 10 msec to obtain succeeding acoustic vectors. Note that other acoustic front-ends with other frame sizes and sampling rates/signal bandwidths can also be employed.

Ex.1005, 12:2–12. Also, Maes states that “the processing performed in blocks 414 and 416 may be accomplished via any conventional acoustic information recognition system capable of extracting and labeling acoustic feature vectors, e.g., Lawrence Rabiner, Biing-Hwang Juang, “Fundamentals of Speech Recognition,” Prentice Hall, 1993.” Ex.1005, 12:43–49.

101. Maes provides several examples of acoustic feature vectors, or alternatively cepstral vectors in connection with the step of extracting acoustic feature vectors. Maes describes “[i]t is to be understood that other variations on [acoustic] features may be used, e.g., [linear prediction coefficients] LPC cepstra, [Perceptual Linear Prediction] PLP and that the invention is not limited to any particular type.” As yet another example of an acoustic feature vector, Maes describes “MEL cepstra.” Ex.1005, 23:40–43, 39:55–58.

102. Furthermore, Maes's system or equivalently, its audio-visual speech recognition module corresponds to "*a computer comprising the knowledge-enhanced speech recognition engine.*" See [12.0]. And further, Maes's "**system ... comprises at least one processor, operatively coupled to the user interface subsystem, and configured to receive at least a portion of the multi-modal input data from the user interface subsystem.** The processor is further configured to then make a determination of at least one of an intent, a focus and a mood of at least one of the one or more users based on at least a portion of the received multi-modal input data. The processor is still further configured to then cause execution of one or more actions to occur in the environment based on at least one of the determined intent, the determined focus and the determined mood. The system further comprises a memory, operatively coupled to the at least one processor, which stores at least a portion of results associated with the intent, focus and mood determinations made by the processor for possible use in a subsequent determination or action."

Ex.1005, 2:38–53.

103. Maes explains that "[I]t should be understood that the elements illustrated in FIGS. 1 through 9C may be implemented in various forms of hardware, software, or combinations thereof, e.g., one or more digital signal processors with associated memory, application specific integrated circuit(s), functional circuitry, one or more appropriately programmed general purpose digital

computers with associated memory, etc. Given the teachings of the invention provided herein, one of ordinary skill in the related art will be able to contemplate other implementations of the elements of the invention.” Ex.1005, 45:50–60.

104. Maes makes clear that the “*transcription of a natural language utterance*” by Maes’s system results in computer-recognizable data, and consequently, the “*transcription*” can be “*receiv[ed]*” by a computer or sent to another computer. Ex.1005, 11:55–64.

105. Further, Maes provides an example of a vehicular system in which a user may say “turn it on,” while pointing or gesturing at the vehicle radio.

The dialog manager would ... receive the results of the recognized events associated with the spoken utterance “turn it on” and the gesture of pointing to the radio. **Based on these events, the dialog manager does a search of the existing applications, transactions or “dialogs,” or portions thereof [stored on the context stack], with which such an utterance [including the accompanying lip movement] and gesture could be associated.**

Ex.1005, 7:65–8:4.

[T]his recognized spoken utterance event is stored on the context stack. Then, when the recognized gesture event (e.g., pointing to the radio) is received, the dialog manager takes this event and **the previous spoken utterance event stored on the context stack** and makes a

determination that the user intended to have the radio turned on.

Ex.1005, 8:37–42. Ex.1005, 6:29–38 (“Next, in step **204**, the raw multi-modal input data is abstracted into one or more events. In terms of the computing system **10** in FIG. 1, the data abstraction is performed by the I/O manager **14**. The I/O manager receives the raw multi-modal data and abstracts the data into a form that represents one or more events, e.g., a spoken utterance, a visual gesture, etc. As is known, a data abstraction operation may involve generalizing details associated with all or portions of the input data so as to yield a more generalized representation of the data for use in further operations.”) Therefore, Maes teaches or suggests this claim element.

iii. [12.2.0] identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription,

106. Maes teaches or suggests this claim element.

107. As explained in connection with element [12.1], in Maes, the “decoded text” (or, script) processed from the spoken utterance constitutes “*a transcription of a natural language utterance.*”

108. Maes discloses implementing a context stack (such as context stack 817 in Figure 8 or context stack 20 in Figure 1) storing “*one or more contexts.*” Maes describes that “the context stack 817 may be implemented ...” as part of the context

stack 20 of Figure 1. Ex.1005, 37:53–55. Referencing Figure 1 (reproduced below),

Maes further describes:

Referring initially to FIG. 1, a block diagram illustrates a multi-modal conversational computing system according to an embodiment of the present invention. **As shown, the multi-modal conversational computing system 10 comprises** an input/output (I/O) subsystem 12, an I/O manager module 14, one or more recognition engines 16, a dialog manager module 18, **a context stack 20** and a mood/focus classifier 22.

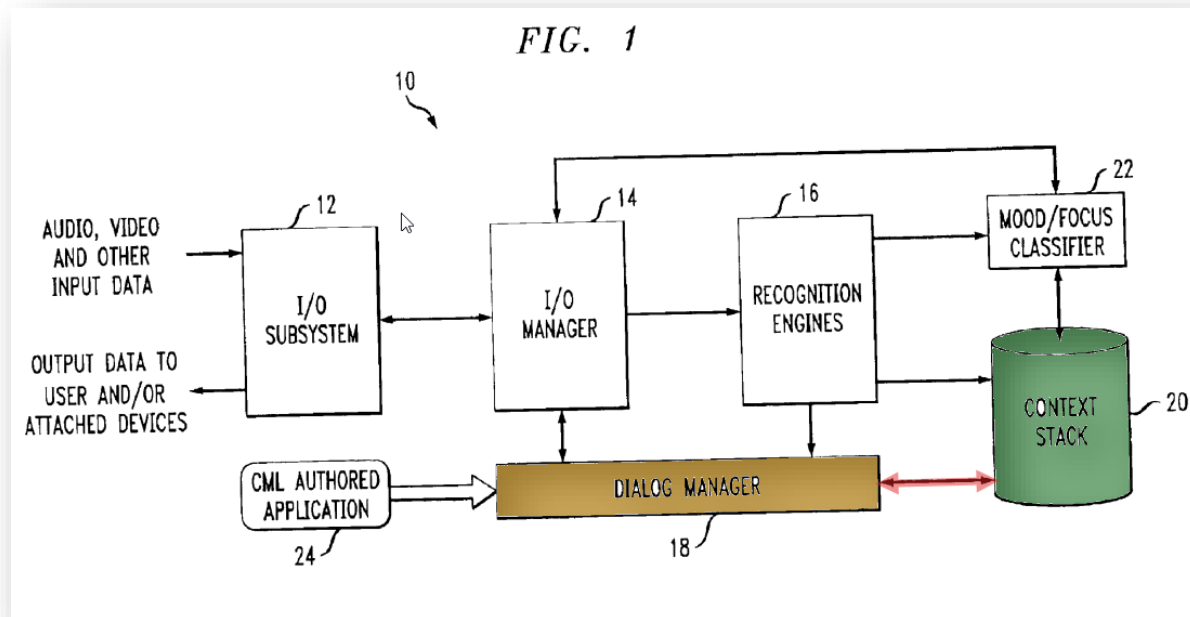
Ex.1005, 3:66–4:6.

109. Maes further describes its context stack storing “historical information (e.g., past [input/output (I/O)] events” or “*one or more contexts*” generated previously in the context of a dialog.

[H]istorical information (e.g., past events) stored in the context stack [such that] the context stack ... is associated with the organized/sorted context corresponding to each active dialog.

Ex.1005, 7:62–63, 37:60–61; Figure 1 (reproduced below); *see also* Ex.1005, 7:40–45 (disclosing “the system 10 attempts to determine the **user intent based on the current event** and the **historical interaction information stored in the context stack** and then determine and execute one or more application programs that

effectuate the user's intention and/or react to the user activity.”); 5:16–19 (disclosing “an I/O (input/output) event generated previously and stored in a context manager/history stack (e.g., if a beeper rang and the user asked ‘turn it off’)”).



Ex.1005, Fig. 1 (annotated).

110. Specifically, Maes provides an example of a vehicular system in which a user may say “turn it on,” while pointing or gesturing at the vehicle radio. Maes describes its system identifying words in a recognized utterance from contexts—such as by matching the utterance “turn it on” to the context “radio”:

The dialog manager would ... receive the results of the recognized events associated with the spoken utterance “turn it on” and the gesture of pointing to the radio. **Based on these events, the dialog manager does a search of the**

existing applications, transactions or “dialogs,” or portions thereof [stored on the context stack], with which such an utterance and gesture could be associated.

Ex.1005, 7:65–8:4.

[T]his recognized spoken utterance event is stored on the context stack. Then, when the recognized gesture event (e.g., pointing to the radio) is received, the dialog manager takes this event and the previous spoken utterance event stored on the context stack and makes a determination that the user intended to have the radio turned on.

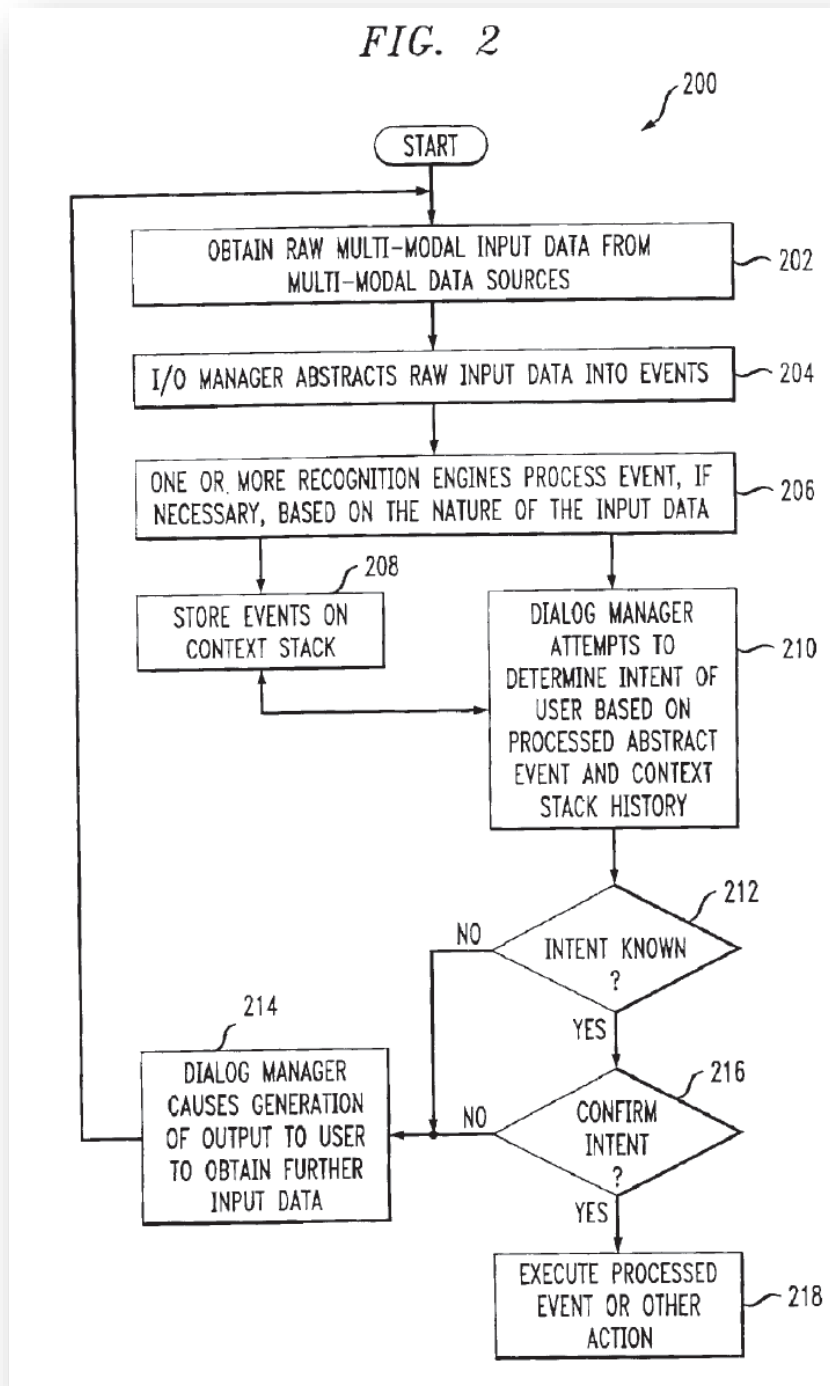
Ex.1005, 8:37–42. In the above example, “this event” corresponds to the current event of the user pointing to the radio, which occurred few seconds after “the previous spoken utterance event” of the user saying “turn it on. Data for both events [“one or more contexts”] are stored on the context stack because Maes discloses **“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”** are stored on the context stack. Ex.1005, 37:55–62. Maes clarifies that “the dialog” includes **“conversational dialog comprising speech and other multi-modal I/O such as GUI keyboard, pointer, mouse, as well as video input, etc.”** Ex.1005, 36:49–54. Therefore, Maes discloses the dialog manager associating the results of the recognized events (“*the one or more text combinations*”) included in the decoded text/script produced as a result of the transcription

(*“contained in the transcription”*) with an organized/sorted context in a context stack such as context stack 817 in Figure 8 or context stack 20 in Figure 1 corresponding to in an active dialog which serves as *“identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription”* and consequently, the associated context(s) stored in the context stack corresponding to the user’s intent in the active dialog constitutes the *“one or more contexts that completely or partially match”* the results of the recognized events (*“the one or more text combinations”*) included in the decoded text/script (*“contained in the transcription”*) associated with the spoken utterance, e.g., “turn it on”. See Ex.1005, 7:65–8:4 and 37:53–55. As explained above, Maes teaches identifying matches between associated context(s) stored in the context stack corresponding to the user’s intent in the active dialog (or, simply contexts stored on the context stack) and results of recognized events (or, simply transcribed user inputs). A POSITA would have understood that in speech recognition systems, it may be the case that—the associated context(s) stored in the context stack corresponding to the user’s intent in the active dialog may not have a full or 100% (i.e., *“complete”*) match with the results of the recognized events (included in the decoded text/script produced as a result of the transcription) but instead have a partial match—and Maes provides for such a possibility describing it as **“could be associated.”** See generally Ex.1005, 7:60–8:4 and 37:53–55.

111. Maes provides an example that teaches or suggests “*contexts that completely or partially match one or more text combinations contained in the transcription.*” Specifically, Maes discloses an example in which the user first says “turn it on” (i.e., a recognized spoken utterance event that is stored on the context stack”) and then a few seconds later points to the radio. Maes’s system first attempts to determine user intent based solely on the turn it on” command. However, since there are likely other devices in the vehicle that could be turned on, “the system would likely not be able to determine with a sufficient degree of confidence [i.e., “*partially match*”] what the user was referring to.” Ex.1005, 8:30–36. Maes continues to explain “[t]hen, when the recognized gesture event (e.g., pointing to the radio) is received, the dialog manager takes this event **and** the previous spoken utterance event stored on the context stack” [e.g., together the contexts of both events “*completely ... match*”] the user’s intent to have the radio turned on. Ex.1005, 8:37–42. Accordingly, with this example, Maes demonstrates “*contexts that ... partially match one or more text combinations contained in the transcription.*”

112. Therefore, a POSITA would have understood that the outcome of the “*identifying ...*” step would generate one or more candidate contexts selected from the context stack that potentially (e.g., completely or partially) match the results of the recognized events (e.g., included in the decoded text/script produced as a result of the transcription). Therefore, Maes teaches or suggests this claim element.

113. For example, Figure 2, below, is described in Maes as depicting “a flow diagram illustrating a referential ambiguity resolution methodology performed by a multi-modal conversational computing system.” Ex.1005, 3:24–28.



Maes, Figure 2. Referencing Figure 2, Maes explains that “[i]n step 208, the recognized events ... are stored in a storage unit referred to as the context stack 20.

The **context stack** is used to create a history of interaction between the user and the system ... **to assist the dialog manager 18 in making referential ambiguity resolution determinations** when determining the user's intent. Next, in step 210, the system 10 attempts to **determine the user intent based on the current event and the historical interaction information stored in the context stack** and then determine and execute one or more application programs that effectuate the user's intention and/or react to the user activity." Ex.1005, 7:33–45. Therefore, Maes teaches or suggests *"identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription."*

114. To the extent that Patent Owner contends that Maes does not teach or suggest the *"identifying ... ,"* step, Coffman discloses this element. First, Maes incorporates Coffman by reference: "It is to be appreciated the conversational virtual machine disclosed in PCT patent application identified as PCT/US99/22927 filed on Oct. 1, 1999 and entitled 'Conversational Computing Via Conversational Virtual Machine,' the disclosure of which is incorporated by reference herein, may be employed to provide a framework for the I/O manager, recognition engines, dialog manager and context stack of the invention." Ex.1005, 9:30–37; see §VIII.C.

115. Coffman explains that context stack entries are searched to identify a context matching one or more words transcribed/determined from the user's spoken utterance. **"Completion/search for the active context is performed from context**

to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained” Ex.1012, 42:1–3. Coffman further explains that searching for the context includes comparing words of the user’s spoken utterance against information stored in the context stack:

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.**

Ex.1012, 41:2–5. Thus, in Coffman, the outputs of conversational subsystems or conventional subsystems which constitute one or more words transcribed/determined from the user’s spoken utterance, are compared against data stored in the context stack until a match 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

Ex.1012, 42:1–3. Thus, Coffman’s teachings of searching the context stack until an acceptable match is found between one or more words transcribed/determined from the user’s spoken utterance and data stored on the context stack serve as the “*identifying ...*” functionality. Indeed, Coffman teaches updating the context stack in a manner such that “[w]hen a message is returned to the user, the context is updated and then pushed to the top of the context stack ... under the active context. The active context is updated to inform of the existence of a returned value.” Ex.1012, 42:7–9. This additionally confirms that Coffman provides the “*identifying ...*” functionality. Accordingly, decoded text (or, script) (“*the transcription*”) comprising “*one or more text combinations*” produced as a result of processing the user’s spoken utterance, as Maes teaches, would be compared against data stored on the context stack, per Coffman’s teachings, to “*identify[] one or more contexts that completely or partially match one or more text combinations contained in the transcription.*” Ex.1012, 41:1–5, 40:15–19; Ex.1005, 7:60–8:4, 8:37–42, and 37:53–55.

116. Therefore, Maes teaches or suggests this element.

- iv. **[12.2.1] wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;**

117. At the outset, the 160 Patent suffers from distinct antecedent basis issues re “the text combinations” in this claim element. First, there is no mention of “text combinations” prior to this claim element. While there is recitation of “one or more text combinations” in a previous claim element, there is simply no reference to “text combinations.” Second, there is no antecedent basis for either “the grammar expression entries” or “the context description grammar.” For purposes of my analysis, I have interpreted “the text combinations” to mean “the one or more text combinations,” and “the grammar expression entries in the context description grammar” to mean “grammar expression entries in a context description grammar.” Notwithstanding these antecedent basis issues, as explained in the analysis below, Maes in combination with Ross renders obvious “wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack.”

118. First, Maes teaches or suggests “wherein identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.”

119. As explained in connection with element [12.2.0], Maes discloses “one or more ... contexts stored in a context stack,” and, moreover, as explained below, the “contexts stored in a context stack” in Maes are “expected contexts.”

120. Specifically, Maes provides an example of a vehicular system in which a user may say “turn it on,” while pointing or gesturing at the vehicle radio:

The dialog manager would ... receive the results of the recognized events associated with the spoken utterance “turn it on” and the gesture of pointing to the radio. **Based on these events, the dialog manager does a search of the existing applications, transactions or “dialogs,” or portions thereof [stored on the context stack], with which such an utterance and gesture could be associated.**

Ex.1005, 7:65–8:4. The contexts stored in Maes’s context stack are “*expected contexts*” because Maes teaches such contexts that “could be associated” with transcribed user input, signaling a possibility or expectation of finding such contexts. Maes states that upon searching the context stack, the associated context(s) stored in the context stack corresponding to the user’s intent in the active dialog is provided by the dialog manager when such context “could be associated” which serves as teaching or suggesting that contexts stored on the context stack are “*expected contexts*.” See generally Ex.1005, 7:60–8:4. As explained above, Maes teaches identifying matches between associated context(s) stored in the context stack corresponding to the user’s intent in the active dialog (or, simply contexts stored on the context stack) and results of recognized events (or, simply transcribed user inputs). A POSITA would have understood that in speech recognition systems, it may be the case that—the associated context(s) stored in the context stack

corresponding to the user’s intent in the active dialog may not have a full or 100% match with the results of the recognized events (included in the decoded text/script)—and Maes provides for such a possibility describing it as “**could be associated.**” *See generally* Ex.1005, 7:60–8:4. Additionally, the contexts stored in Maes’s context stack are “*expected contexts*” because Maes describes its context stack used for storing a “**a history of interaction between the user and the system.**” Ex.1005, 7:35–37; *see also* 7:40–42 (“the **historical interaction information stored in the context stack**”).

121. Further, because Maes discloses that information stored in the context stack relate to “context[s] corresponding to each active dialog,” the “*one or more ... contexts stored in a context stack*” comprise contexts corresponding to each active dialog. Ex.1005, 37:60–62. *See also* Ex.1005, 7:60–8:6 (“[T]he dialog manager must ... determine the user’s intent based on the current event and, if available, the historical information (e.g., past events) stored in the context stack. ... The dialog manager would ... receive the results of the recognized events Based on these events, the dialog manager ... determines the appropriate [context].”).

122. Moreover, for the reasons below, Maes discloses “*wherein identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*” For example, Maes teaches or suggests upon searching the context stack, the associated context stored in the

context stack corresponding to the active dialog is provided by the dialog manager when such context “could be associated,” which therefore constitutes “comparing” the results of the recognized events (“*the text combinations*”) included in the decoded text/script produced as a result of the transcription (i.e., [*contained in the transcription*]) against data stored in the context stack (e.g., *one or more expected contexts stored in a context stack*). Ex.1005, 7:60–8:4. Referencing context stack 817 depicted in Figure 8, Maes discloses:

The **context stack 817** 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. The context stack is associated with the organized/sorted context corresponding to each active dialog.

Ex.1005, 37:55–62. Because Maes teaches “all the information” and “any extraneous event” that occurs during the dialog is stored on the context stack, a POSITA would have understood that Maes’s context stack storing **all**—which would include “*one or more expected*”— contexts, and therefore it would have been obvious to a POSITA that in Maes “*identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*” Ex.1005, 37:55–62.

Thus, the dialog manager must first determine the user’s intent based on the current event and, if available, the

historical information (e.g., past events) stored in the context stack. For instance, returning to the vehicle example, the user may say “turn it on,” while pointing at the vehicle radio. The dialog manager would therefore receive **the results of the recognized events** associated with the spoken utterance “turn it on” and the gesture of pointing to the radio. **Based on these events, the dialog manager does a search of the existing applications, transactions or “dialogs,” or portions thereof [stored on the context stack], with which such an utterance and gesture could be associated.**

Ex.1005, 7:60–8:4, 8:37–42.

123. Referring to the example in which Maes’s multi-modal conversational computing system is employed within a vehicle, Maes provides an example of comparing the results of the recognized events (e.g., “*the text combinations*”) against data stored in the context stack (e.g., “*one or more expected contexts stored in a context stack*”) as follows:

Consider the case where the user first says “turn it on,” and then a few seconds later points to the radio. The dialog manager would first try to determine user intent based solely on the “turn it on” command. However, since there are **likely other devices in the vehicle that could be turned on, the system would likely not be able to determine with a sufficient degree of confidence what the user was referring to.** However, this recognized spoken utterance event is stored on the context stack. **Then, when the recognized gesture event (e.g., pointing**

to the radio) is received, the dialog manager takes this event and the previous spoken utterance event stored on the context stack and makes a determination that the user intended to have the radio turned on.

Ex.1005, 8:30–42. In the above example, “this event” corresponds to the current event of the user pointing to the radio, which occurred few seconds after “the previous spoken utterance event” of the user saying “turn it on. Data for both events [“one or more contexts”] are stored on the context stack because Maes discloses “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” are stored on the context stack. Ex.1005, 37:55–62. Therefore, in making a determination of the user’s intent based on data for both events stored on the context, Maes teaches or suggests “*comparing the text combinations ... against one or more expected contexts stored in a context stack*” and the outcome of the “*comparing*” results in identification of “the user[’s] inten[t] to have the radio turned on” (“*the matching context*”). Therefore, Maes discloses “*wherein identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*”

124. To the extent that Patent Owner contends that Maes does not teach or suggest the step of “*wherein identifying the matching contexts includes comparing*

the text combinations ... against one or more expected contexts stored in a context stack,” Coffman (incorporated by Maes) discloses this step as explained below:

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.**

Ex.1012, 41:2–5. Thus, in Coffman, the outputs of conversational subsystems or conventional subsystems which constitute one or more words transcribed/determined from the user’s spoken utterance, are compared against data stored in the context stack until a match is found. Thus, Coffman’s teachings of searching the context stack until an acceptable match is found between one or more words transcribed/determined from the user’s spoken utterance and data stored on the context stack serve as the “*wherein identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*”

125. Coffman explains that identifying context entries stored on the context stack includes “*comparing*” data stored on the context stack to one or more words transcribed/determined from the user’s spoken utterance. “**Completion/search for**

the active context is performed from context to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained” Ex.1012, 42:1–3. Therefore, Maes (based on incorporation of Coffman) teaches or suggests “*wherein identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*”

126. Although, Maes does not expressly disclose “*wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar,*” Maes generally describes use of “grammar” and “grammar database” in its system.

127. Maes explains that “**data needed by any recognition engine (e.g., grammar, acoustic models, video patterns ...),**” such as a speech recognition engine are present on conversational virtual machine (CVM), which Maes describes is a “component for implementing conversational computing according to the conversational computing paradigm described above with respect to the present invention.” Ex.1005, 31:15–18 and 34:61–67. Maes continues to explain that the CVM handles input/output issues with conversational subsystems which “**us[e] the appropriate data files 794 (e.g., contexts, finite state grammars, vocabularies, language models, symbolic query maps, ...).**” Ex.1005, 33:11-21. Maes explains that “via the I/O manager 14 of FIG. 1” “user-provided input data events are ...

provided to” “apparatus 900”—an “apparatus for collecting data associated with a voice of a user,” which includes “a dialog management unit 902 for conduct[ing] a conversation with the user,” depicted in Figure 9A. Ex.1005, 39:24–26, Figure 9A. Ex.1005, 39:22–29) “Dialog management unit 902,” Maes explains, can “include natural language understanding (NLU), natural language generation (NLG), **finite state grammar (FSG)**, and/or text-to-speech Syntheses (TTS) for machine-prompting the user in lieu of, or in addition to, the human operator.” Ex.1005, 41:13–20.

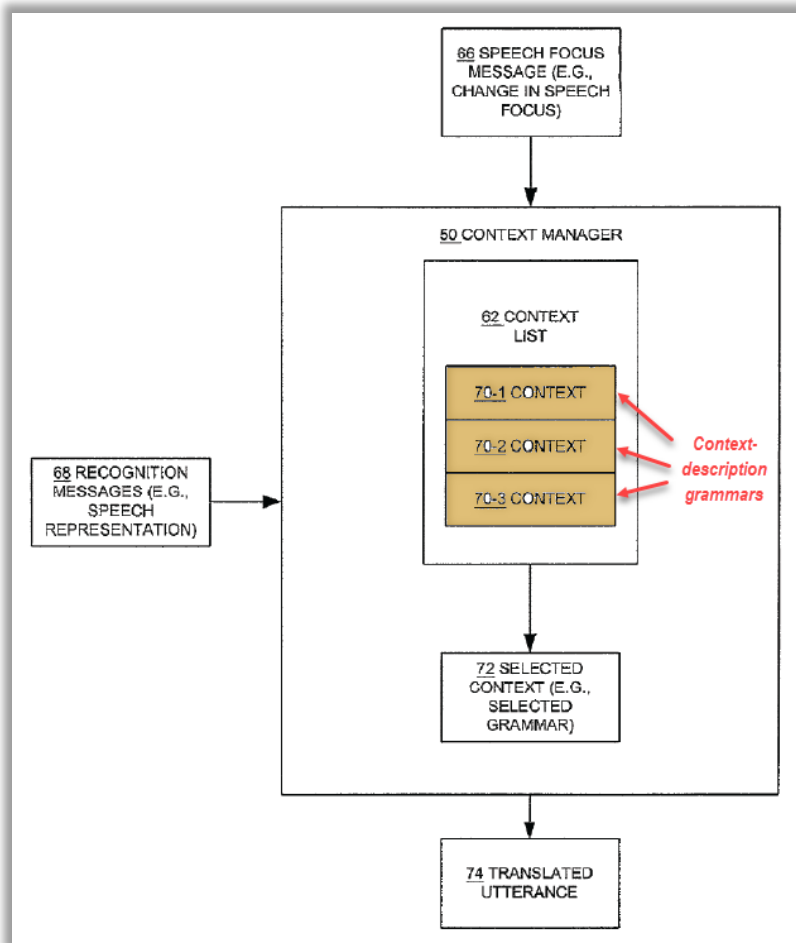
128. Also, Maes describes use of a “grammar database” stating that: “Apparatus 900 further includes a processing module 910” which can further include “a speech recognizer 926 which can, in turn, itself include a speech recognition module 928 and a speech prototype, language model and **grammar database 930**.” Ex.1005, 39:59 and 41:35–38.

129. Although, Maes does not expressly disclose “*wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar*,” Ross teaches or suggests this limitation, as explained below. See Ex.1022, [0034]-[0035], [0041]-[0051].

130. First, Ross teaches a “*grammar expression entries in [a] context description grammar*.” For example, Ross teaches “a grammar is defined for each application 26” and referencing Figure 4 (reproduced below), Ross discloses

“context list includes contexts 70 (e.g., 70-1, 70-1, 70-3, etc.) for speech-enabled applications 26, which represent the grammars for the applications 26.”

Ex.1022, [0033] and [0035]. Because Ross describes that contexts 70-1, 70-1, 70-3 represent the grammars for the applications, Ross’s contexts representing grammars (such as 70-1, 70-1, 70-3) constitute “*context description grammar[s]*” describing contexts for three speech-enabled applications. For instance, context 70-1 is a grammar for a first speech-enabled application, context 70-1 is a grammar for a second speech-enabled application, and context 70-3 is a grammar for a third speech-enabled application.



Ex.1022, Figure 4 (annotated).

131. Furthermore, Ross teaches or suggests examples demonstrating “*grammar expression entries*” included in a “*context description grammar*.” In Ross, “a grammar is defined for each application 26” and Ross describes an example of selection between two grammars that serve as the contexts representing the grammars (“*context description grammar[s]*”) for two applications: 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 | . . .
```

(Ex.1022, [0035] and [0040]-[0041]) and another for a calendar application:

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

(Ex.1022, [0046]-[0047]). The above-mentioned grammars comprise “*grammar expression entries*.” Specifically, in the grammar for the electronic mail application shown above, the line “<message> = the? <nth> message | it | this,” defines a rule named “message” which includes phrases (e.g., “the,” “message”), a reference to another rule (“<nth>”) and a grammar operator (“?”), which serve as examples of “*grammar expression entries*.” Similarly, in the grammar for the calendar application shown above, phrases (e.g., “the,” “appointment”), a reference to another

rule (“<nth>”) and a grammar operator (“?”) serve as examples of “*grammar expression entries*.”

132. Additionally, Ross provides examples explaining how the entries (e.g., the phrases, keywords, and operators) included in a grammar are used. With respect to the grammar for the electronic mail application, Ross describes allowing a user’s spoken phrases (e.g., processed to generate “*the text combinations*”) such as “open the first message,” “create a message,” “send this,” and “print it” to be matched (“*compar[ed]*”) against entries in the grammar for the electronic mail application (“*the grammar expression entries in the context description grammar*”). The grammar for the calendar application allows matching (“*comparing*”) of spoken phrases such as “open the first appointment,” “create an appointment,” “print the fourth appointment,” and “print it” against entries in the grammar for the calendar application. *See generally* Ex.1022, [0041]-[0051].

133. Ross also discloses that “*identifying the matching contexts includes*” the “*comparing ...*” step. For example, Ross discloses:

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 priority of the matching grammar (i.e., selected context 72) is raised.**

Ex.1022, [0034]. Ross notes that “**when an utterance is recognized, it will be tested against each application’s grammar to see if the grammar will accept it.**”

Ex.1022, [0035]. Furthermore, Ross explains “maintain[ing] the priority and state of the various grammars in the context list 62 in the system” so that “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.”

Ex.1022, [0034]. Ross also explains that one goal of testing (“*comparing*”) a recognized utterance (e.g., “*the text combinations*”) against data included in context grammars (“*the grammar expression entries in the context description grammar*”) is to find matches between the utterance and such grammars. “**When a successful match is found** [based on testing the recognition messages against the active grammars], the corresponding translation 74 is dispatched to the script engine 38 for execution, and the priority of the matching grammar (i.e., selected context 72) is raised.” Ex.1022, [0034]. Therefore, Ross “uses a grammar to identify which speech enabled application is to receive the representation of the spoken utterance” and specifically “**to determine if a representation of a spoken utterance from a user is acceptable to (can be processed by) a particular speech-enabled application.**” Ex.1022, [0013]. Ross further explains that “prior to evaluating the contexts,” (the “*comparing*” step), its system “create[s] the contexts for the speech enabled applications in the speech enabled environment.” Ex.1022, [0010].

134. The disclosures in Ross related to contexts representing grammars (*see e.g., Ex.1022, [0033]*) are analogous to the 160 Patent's description of "*context description grammar*" which describes it as including one or more grammar expression entries and "use[d] to process requests in respective contexts." Ex.1001, Cl 1.

135. In view of the afore-mentioned disclosures in Ross, it would have been obvious to a POSITA to modify Maes's system to additionally include Ross's teachings of context grammars for applications such that "*the text combinations*" produced as a result of processing the user's spoken utterance using Maes's system would be compared against data (e.g., keywords/phrases used in applications) included in context grammars, per Ross. *See* §VIII.C.

136. Therefore, Maes in combination with Ross renders this element obvious.

v. [12.3] scoring each of the identified matching contexts;

137. Maes teaches or suggests "scoring each of the identified matching contexts."

138. First, as explained in connection with element [12.2.0], Maes teaches or suggests "*identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription*" and as such, in Maes, the

associated context(s) stored in the context stack corresponding to the user's intent in the active dialog constitutes "*the identified matching contexts*" which equivalently are one or more candidate contexts selected from the context stack that potentially (e.g., completely or partially) match the results of the recognized events (e.g., included in the decoded text/script produced as a result of the transcription). And, furthermore, for the reasons below, Maes teaches or suggests "*scoring each of the identified matching contexts.*"

139. Maes teaches scoring one or more candidate contexts such as phonemes, which Maes describes as "sub-phonetic or acoustic units of speech," (Ex.1005, 12:29–33) or equivalently as "portions [of dialogs]" (Ex.1005, 7:60–8:4). Specifically, Maes discloses generating, for each considered phoneme in a given context, likelihood scores (indicating the likelihood that it was that particular phoneme that was spoken) based, for example, on the audio information and/or the video information received as input to the system. And referencing Figure 4, Maes discloses:

After the acoustic feature vectors, denoted in FIG. 4. by the letter A, are extracted, the probability module labels the extracted vectors with one or more **previously stored phonemes which ... are sub-phonetic or acoustic units of speech. The module may also work with lefemes, which are portions of phones [sic: phonemes] in a given**

context. Each phoneme associated with one or more feature vectors has a probability associated therewith indicating the likelihood that it was that particular acoustic unit that was spoken.

Ex.1005, 12:29–37. Ex.1005, 12:29–37.

[T]he probability module 416 in the audio information path ... labels the acoustic feature vectors with one or more phonemes.

Ex.1005, 18:46–48.

Thus, the probability module yields likelihood scores for each considered phoneme in the form of the probability that, given a particular phoneme or acoustic unit (au), the acoustic unit represents the uttered speech characterized by one or more acoustic feature vectors A or, in other words, $P(A|\text{acoustic unit})$.

Ex.1005, 12:38–43.

Similar to the probability module 416 in the audio information path which labels the acoustic feature vectors with one or more phonemes, the probability module 426 labels the extracted visual speech vectors with one or more **previously stored phonemes**. Again, each phoneme associated with one or more visual speech feature vectors has **a probability associated therewith indicating the likelihood that it was that particular acoustic unit that was spoken in the video segment being considered**. Thus, the probability module yields likelihood scores for each considered phoneme in the form of the probability that, given a particular phoneme or acoustic unit (au), the acoustic unit represents the uttered speech characterized by one or more visual speech feature vectors V or, in other words, $P(V|\text{acoustic unit})$. Alternatively, the visual speech

feature vectors may be labeled with visemes which, as previously mentioned, are visual phonemes or canonical mouth shapes that accompany speech utterances.

Ex.1005, 18:46–60. Therefore, in Maes, calculation of likelihood scores for each considered phoneme in a given context in the form of probabilities serves as “*scoring each of the identified matching contexts.*”

140. Therefore, Maes teaches or suggests this element.

vi. **[12.4] selecting the matching context having a highest score to determine a most likely context for the utterance; and**

141. As explained in detail in connection with element [12.3], Maes teaches or suggests generating likelihood scores for each considered phoneme in a given context (“*scoring each of the identified matching contexts*”). Specifically, and as explained in element [12.3], Maes discloses generating, for each considered phoneme in a given context, likelihood scores (indicating the likelihood that it was that particular phoneme that was spoken) based, for example, on the audio information and/or the video information received as input to the system.

142. After the likelihood scores (e.g., probabilities $P(A|\text{acoustic unit})$ and/or $P(V|\text{acoustic unit})$) are generated, Maes describes using such scores used in identifying (“*selecting*”) “**the acoustic units** [i.e., “*the matching context*” / “*a most likely context*” identified in the context stack as explained in [12.2.0] and [12.2.1]]... **as having the highest probabilities [i.e., “a highest score”] of representing what**

was uttered” (*“for the utterance”*). Ex.1005, 18:64–65 and 20:13–15. Thus, the candidate context (such as a phoneme or an acoustic unit) that has the highest probability of representing what was uttered corresponds to “the matching context” / *“a most likely context for the utterance.”* See Ex.1005, 18:64–65 and 20:13–15.

143. Therefore, Maes teaches or suggests this claim element.

- vii. **[12.5] communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.**

144. Maes in combination with Ross renders this claim element obvious.

145. Maes teaches or suggests *“communicating a request to a domain agent to process requests in the most likely context for the utterance.”*

146. Initially, the 160 Patent describes a “*domain agent*” as follows: “According to another embodiment of the invention, system 90 may include different types of agents 106. For example, generic and domain specific behavior and information may be organized into domain agents. The domain agents provide complete, convenient and re-distributable packages or modules for each application area.” Ex.1001, 14:21–27. Thus, “*domain agent[s]*” in the 160 Patent broadly refer to software modules that are specific to each application area.

147. First, Maes teaches or suggests “*a domain agent configured to process requests in the most likely context for the utterance.*” For example, Maes discloses:

the system 10 attempts to determine the user intent based on the current event and the historical interaction information stored in the context stack and then determine and execute one or more application programs that effectuate the user’s intention and/or react to the user activity. **The application depends on the environment that the system is deployed in.**

Ex.1005, 7:40–46.

148. As such, Maes’s teachings of application programs (effectuating the user’s intention and depending on the environment that the system is deployed) are analogous to the 160 Patent’s description of “*domain agent*,” and therefore “application programs” or “applications” in Maes serve as “*domain agent[s]*.” See also Ex.1005, 30:67 (“**run applications**”), 8:1–2 (“**search of the existing applications**”). Maes explains this further in greater detail. Maes describes that “the conversational dialog manager 819 **determines** what information the user has, which inputs the user presents, and **which application(s) should handle the user inputs.** ... The DMA function [of the dialog manager] handles multi-modal I/O events to: (1) **determine the target application or dialog (or portion of it)** and (2) use past history and context.” Ex.1005, 36:49–37:3. Maes explains that “[t]he DMA

[function of the dialog manager] processes abstracted I/O events (abstracted by the I/O manager) using the context/history to understand the user intent. When an abstract event occurs, the DMA determines the target of the event and ... then launches the action associated to the user's query" which serve as describing "which serve as *"process[ing] requests in the most likely context for the utterance."*

Ex.1005, 36:49–37:3. For example,

[i]f the dialog manager determines with some predetermined degree of confidence that the application it selects is the one which will effectuate the users desire, the dialog manager executes the next step of the multi-modal dialog (e.g., prompt or display for missing, ambiguous or confusing information, asks for confirmation or launches the execution of an action associated to a fully understood multi-modal request from the user) of that application based on the multi-modal input. That is, the dialog manager selects the appropriate device (e.g., radio) activation routine and instructs the I/O manager to output a command to activate the radio.

Ex.1005, 8:8–8:19. For example, in Maes, a matching context corresponding to a fully understood request based on the user's input serves as *"the most likely context for the utterance."* By teaching that Maes's dialog manager (1) determines which

application(s) should handle the user inputs (such as a spoken utterance), (2) determines the target application or dialog (or portion of it), and (3) launches the execution of an action associated to a fully understood request from the user of that application based on the user's input, Maes teaches or suggests *a domain agent configured to process requests in the most likely context for the utterance.*"

149. Maes teaches or suggests "*communicating a request to a domain agent*" with reference to the illustrative example of an in-vehicle system in which the user may say the spoken utterance "turn it on" while pointing at the vehicle radio. Maes discloses that "the dialog manager selects the appropriate device (e.g., radio) activation routine and instructs the I/O manager to output a command to activate the radio." Ex.1005, 7:63–64, 8:15–19. Maes's system processes inputs relating to an I/O event representative of the user's spoken utterance (audio portion) and/or the accompanying lip movement of the spoken utterance (video portion). Ex.1005, 4:30–32, 6:43–50, and 7:63–8:29. Therefore, in this example, "the application" ("*the domain agent*"), such as a "radio activation routine" handles the activation task "*configured to process requests*" "resulting from fall [sic:full] understanding of the intent of the user" ("*in the most likely context*") "*for*" the user's spoken utterance ("*the utterance*"). Ex.1005, 37:2–3. The user's query/command for a particular application corresponds to "*a request to the domain agent*" and launching the execution of an action (e.g., selecting an application such as a radio activation

routine) associated with the user's query (e.g., comprising a request for launching an application such as a radio activation routine) serves as "*communicating a request to a domain agent configured to process requests in the most likely context for the utterance.*" Ex.1005, 7:60–8:29.

150. Also, Maes identifies that its system "**provides the capability to: (i) determine an object, application or appliance addressed by the user; (ii) determine the focus of the user and therefore determine if the user is actively focused on an appropriate application and, on that basis, to determine if an action should be taken; (iii) understand queries based on who said or did what, what was the focus of the user when he gave a multi-modal query/command and what is the history of these commands and focuses; and (iv) estimate the mood of the user and initiate and/or adapt some behavior/service/appliances accordingly.**" Ex.1005, 2:54–64.

151. Therefore, Maes teaches or suggests "*communicating a request to a domain agent to process requests in the most likely context for the utterance.*"

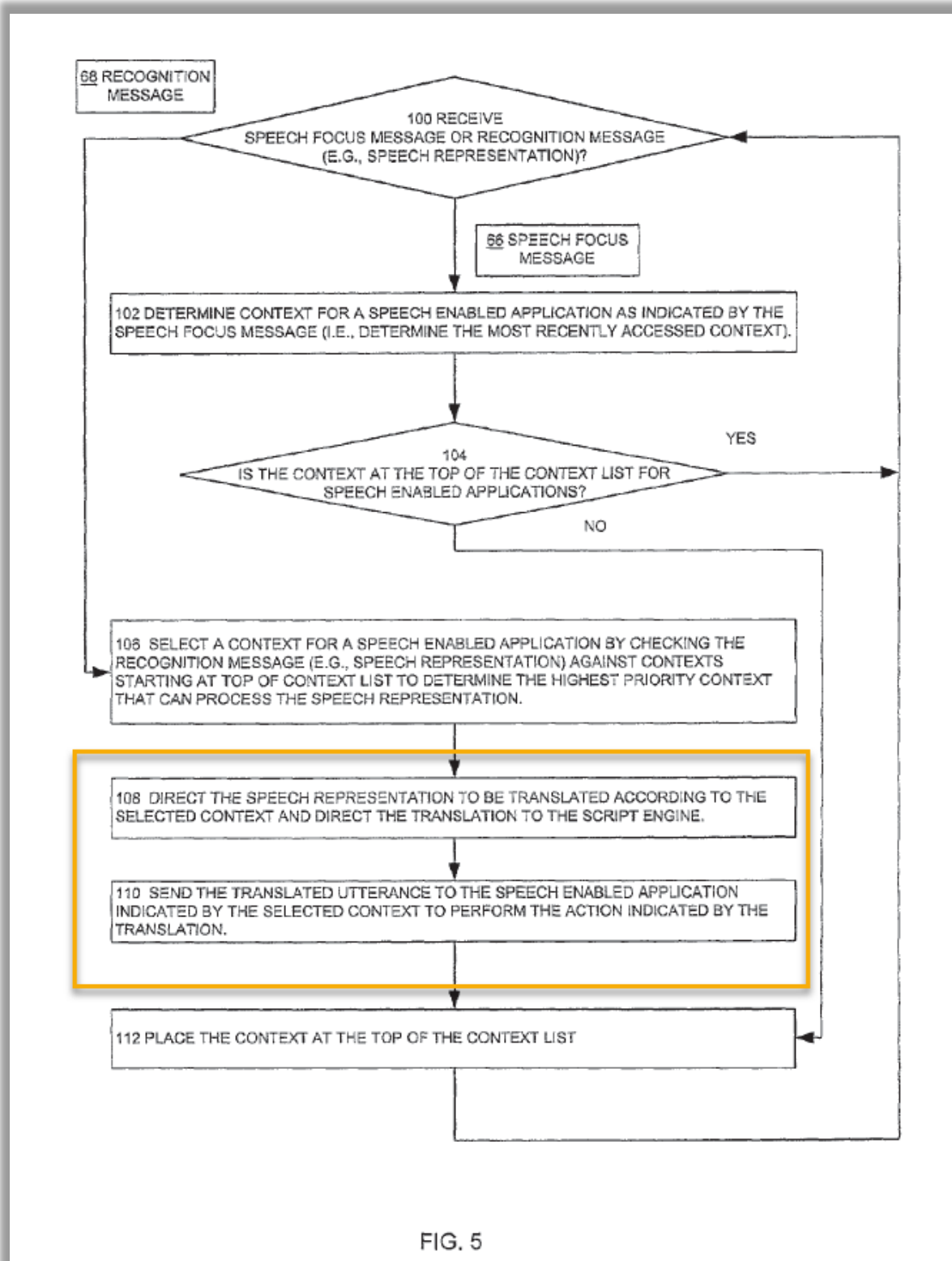
152. Although Maes does not expressly disclose "*the request formulated using at least one grammar expression entry in the context description grammar,*" Ross discloses this feature. Specifically, and as explained in detail in element [12.2.1], Ross provides an example in which a user's query/command (e.g., "[a] request") utilizes spoken phrases ("*at least one grammar expression entry*") "*in*" a

context representing the grammar (“*the context description grammar*”) for a speech-enabled application. Ross describes that the grammar for an electronic mail application allows “phrases to be spoken by a user” of the electronic mail application and the grammar for a calendar application allows “phrases to be spoken by a user.” Ex.1022, [0041]-[0051]. Accordingly, in the example grammars of Ross, the “phrases to be spoken by a user” constitutes as examples of “*request[s] formulated using at least one grammar expression entry in the context description grammar*”

153. Ross provides greater detail explaining how “[a] request [is] formulated using at least one grammar expression entry in the context description grammar.” Ross begins by explaining that “a grammar is defined for each application.” Ex.1022, [0035]. “When a successful match is found [based on testing the recognition messages against the active grammars for applications], **the corresponding translation 74 is dispatched to the script engine 38 for execution, and the priority of the matching grammar (i.e., selected context 72) is raised [within a prioritized list of grammars in context list 62].**” Ex.1022, [0034]-[0035]. Thus, the selected context 72 (comprising keywords and phrases) constitutes the specific grammar (e.g., “*the context description grammar*”) that will accept the user’s spoken utterance. Referencing the process in Figure 5, Ross explains:

In step 108, the context manager 50 **directs the speech representation to be translated according to the selected context 72** and directs the translation to the script engine 38. In step 110, the context manager 50 sends the translated utterance 74 to the speech enabled application 26 indicated by the selected context 72 to perform the action indicated by the translated utterance 74.

Ex.1022, [0037].



Ex.1022, Figure 5 (annotated).

The first application's grammar [e.g., the selected context 72] which will accept the utterance is then used for translation, and the command dispatched to the corresponding application 26.

Ex.1022, [0036]. The act of using the selected context grammar 72 for translating the speech representation (corresponding to the user's spoken utterance) constitutes "*formulat[ing]*" the speech representation (corresponding to the user's spoken utterance) ("*a request*"), for instance, "*using*" keywords or phrases ("*at least one grammar expression entry*") "*in*" the selected context ("*the context description grammar*"). In view of the afore-mentioned disclosures in Ross, it would have been obvious to a POSITA to modify requests sent/communicated to the target application (e.g., to handle the task corresponding to the user's intention and/or react to the user activity) in Maes's system to additionally include Ross's teachings of context grammars for applications so that a request to the target application is constructed utilizing the context grammar for the target application, per Ross's teachings. See §VIII.C.

154. Therefore, Maes in combination with Ross renders this element obvious.

2. Independent Claim 13

- i. **[13.0] A method for processing natural language utterances, comprising:**

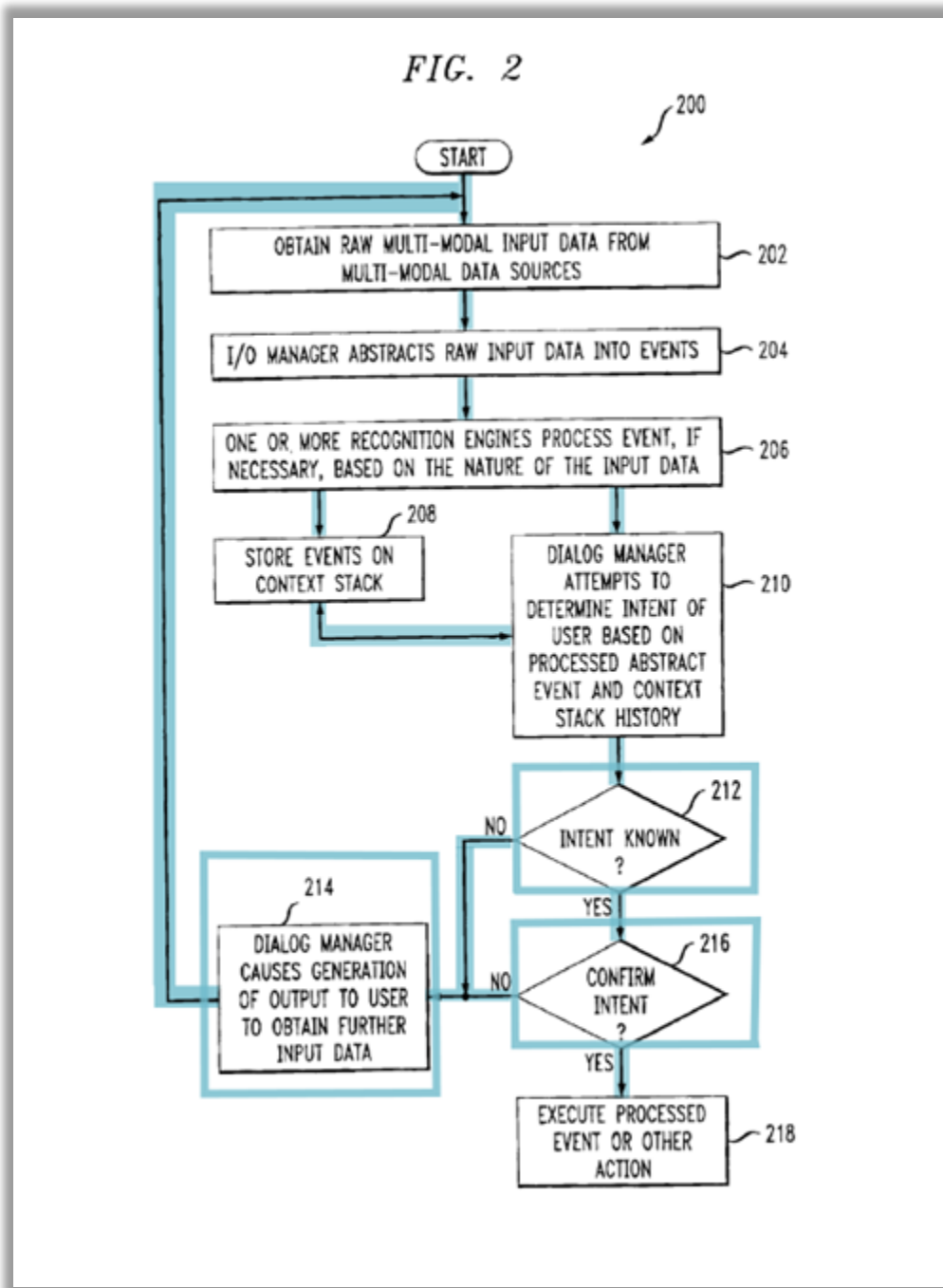
155. *See* claim element [12.0].

- ii. **[13.1] receiving a natural language utterance at a computer comprising a multi-pass speech recognition module;**

156. Other than replacing “*speech recognition engine*” in [12.0] with “*speech recognition module*” in this claim element, this element is substantively similar to [12.0]. *See* [12.0] which provides an explanation with respect to Maes’s system “*receiving a natural language utterance*” and [12.1] which explains Maes’s system corresponds to “*a computer comprising the ... speech recognition module.*” *See also* Ex.1005, 4:1–17, 10:50–52, 11:24–26, 6:5–8.

157. In addition, Maes’s system corresponds to “a computer comprising a multi-pass speech recognition module.” Referencing Figure 2 (reproduced below), Maes teaches an example in which process 200 “performed by a multi-modal conversational computing system” “**iterates based on the new data. Such iteration can continue as long as necessary for the dialog manager to determine the user’s intent.**” Ex.1005, 8:61–65 and 5:52–54. Specifically, Maes discloses that when an abstract event occurs, the system determines the target of the event and, “if needed, seeks confirmation, disambiguation, correction, more details, etc., **until the intent is unambiguous and fully determined.**” Maes’s system (“*speech recognition*

module”) implementing a process that performs multiple iterations (“*multi-pass*”) to decode a natural language utterance until the user’s intent is unambiguous and fully determined therefore would be understood by a POSITA to constitute a “*multi-pass speech recognition module*,” as such techniques were well-known at the time. *See, e.g.* Ex.1023, Abstract (teaching a “common approach to automatic speech recognition uses two recognition passes to decode an utterance: the first pass limits the search to a smaller set of likely hypotheses; and the second pass rescores the limited set using more detailed acoustic models which may target gender or specific channels.”); Ex.1024, 545, 548 (describing “a multi-pass approach to certain types of large-vocabulary recognition tasks by refining the vocabulary between passes, while operating in real time” and stating that “[a] multi-pass approach locating the state name in the first pass, the city name in the second pass, and finally the street in the third pass will almost certainly be preferable.”).



Ex.1005, Figure 2 (annotated to show iterative decoding passes in Maes's system).

Therefore, Maes teaches or suggests a “*multi-pass speech recognition module*” at least for this reason.

158. At the time of filing of the 160 Patent, “*multi-pass speech recognition*” techniques were well-known in the art, and it would have been obvious to implement those techniques in Maes based on such knowledge to the extent one may assert that Maes does not disclose those techniques explicitly or implicitly. For instance, Mao teaches “common approach to automatic speech recognition uses two recognition passes to decode an utterance: the first pass limits the search to a smaller set of likely hypotheses; and the second pass rescores the limited set using more detailed acoustic models which may target gender or specific channels.” Ex.1023, Abstract. As another example, in 2005, Hetherington published a paper describing “a multi-pass approach to certain types of large-vocabulary recognition tasks by refining the vocabulary between passes, while operating in real time.” Ex.1024, 545. Hetherington states that “[a] multi-pass approach locating the state name in the first pass, the city name in the second pass, and finally the street in the third pass will almost certainly be preferable.” Ex.1024, 548. A POSITA would have been motivated to combine the “*multi-pass speech recognition*” based on the knowledge of a POSITA because of the advantages and improvements in results based on multiple iterations.

159. Therefore, Maes teaches or suggests this claim element.

iii. **[13.2] transcribing the utterance using the multi-pass speech recognition module, the multi-pass**

speech recognition module configured to transcribe the utterance into text;

160. See claim element [12.1]. Although element [12.1] recites “receiving a transcription of a natural language utterance,” as explained in element [12.1], “receiving a transcription of a natural language utterance” encompasses the step of “transcribing [an] utterance”

iv. [13.3.0] identifying one or more contexts that completely or partially match one or more text combinations contained in the text of the transcribed utterance,

161. See claim element [12.2.0].

v. [13.3.1] wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;

162. See claim element [12.2.1].

vi. [13.4] scoring each of the identified matching contexts;

163. See claim element [12.3].

vii. [13.5] selecting the matching context having a highest score to determine a most likely context for the utterance; and

164. See claim element [12.4].

viii. [13.6] communicating a request to a domain agent configured to process requests in the most

likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.

165. See claim element [12.5].

B. Ground 2: Maes, Coffman and Ross Render Obvious claims 12 and 13

166. To the extent one asserts that Coffman is not expressly and/or particularly incorporated by reference into Maes as if they were effectively the same document, claims 12 and 13 would have nonetheless been obvious in view of Maes, Coffman and Ross for the reasons articulated above in Ground 1. Moreover, a POSITA would have been motivated to combine the references and had a reasonable expectation of success for such a combination for the reasons articulated in § VIII.C.

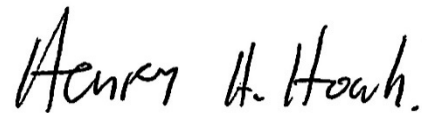
X. OBJECTIVE INDICIA OF NON-OBVIOUSNESS

167. I am not aware of any objective indicia of non-obviousness that would change my opinion that the challenged claims of the 160 Patent are obvious in view of the prior art discussed above in this declaration. It is my opinion that any potential objective indicia of non-obviousness would not overcome the clear disclosures in the prior art showing that the subject matter of these claims would have been obvious to a POSITA as of August 5, 2005. However, I reserve the right to supplement my opinions Patent Owner alleges or offers any purported evidence of objective indicia of non-obviousness.

XI. CONCLUSION

168. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and that these statements were made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code.

Dated: March 10, 2025



CLAIM LISTING

Claim 12	
[12.0]	A method for interpreting natural language utterances using knowledge-enhanced speech recognition engine, wherein the knowledge-enhanced speech recognition engine is configured to determine an intent and correct false recognitions of the natural language utterances, comprising:
[12.1]	receiving a transcription of a natural language utterance at a computer comprising the knowledge-enhanced speech recognition engine;
[12.2.0]	identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription,
[12.2.1]	wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;
[12.3]	scoring each of the identified matching contexts;
[12.4]	selecting the matching context having a highest score to determine a most likely context for the utterance; and
[12.5]	communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.
Claim 13	
[13.0]	A method for processing natural language utterances, comprising: receiving a natural language utterance at a computer comprising a multi-pass speech recognition module;
[13.1]	receiving a natural language utterance at a computer comprising a multi-pass speech recognition module;
[13.2]	transcribing the utterance using the multi-pass speech recognition module, the multi-pass speech recognition module configured to transcribe the utterance into text;
[13.3.0]	identifying one or more contexts that completely or partially match one or more text combinations contained in the text of the transcribed utterance,
[13.3.1]	wherein identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar and against one or more expected contexts stored in a context stack;

[13.4]	scoring each of the identified matching contexts;
[13.5]	selecting the matching context having a highest score to determine a most likely context for the utterance; and
[13.6]	communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.