UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION

IPR2025-00657

U.S. Patent No. 9,263,039

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 15
IV. RELEVANT LEGAL STANDARDS 17
V. OVERVIEW OF THE 039 PATENT 22
A. The Disclosures of the 039 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
IX. CLAIMS 13-15 AND 17-18 ARE UNPATENTABLE 28
A. Ground 1: Maes and Ross render obvious claims 13-15 and 17-18 28
1. The Combination of Maes and Ross 28
2. Invalidity Analysis of claims 13-15 and 17-18 40
a. Independent Claim 13 40
i. [13.0] A method of processing speech and non-speech communications, comprising: 40
ii. [13.1] receiving the speech and non-speech communications; 46
iii. [13.2] transcribing the speech and non-speech communications to create a speech-based textual message and a non-speech-based textual message; 53
(A) Transcribing the speech communications 53
(B) Transcribing the non-speech communications 61

iv. [13.3] merging the speech-based textual message and the non-speech-based textual message to generate a query;.....67

(A) Merging the speech-based and non-speech-based textual messages 67

(B) Merging generates a query.....75

v. [13.4] searching the query for text combinations; 76

vi. [13.5] comparing the text combinations to entries in a context description grammar; 78

vii. [13.6] accessing a plurality of domain agents that are associated with the context description grammar;.....85

viii. [13.7] generating a relevance score based on results from comparing the text combinations to entries in the context description grammar;.....87

ix. [13.8] selecting one or more domain agents based on results from the relevance score;.....91

x. [13.9] obtaining content that is gathered by the selected domain agents; and.....93

xi. [13.10] generating a response from the content, wherein the content is arranged in a selected order based on results from the relevance score. 95

b. Dependent Claim 14 97

i. [14.0] The method according to claim 13, further comprising generating an aggregate response that includes the content that is gathered by the selected domain agents. 97

c. Dependent Claim 15 99

i. [15.0] The method according to claim 13, further comprising: receiving a follow-up speech and non-speech communications; 99

ii. [15.1] transcribing the follow-up speech and non-speech communications to create a follow-up speech-based textual message and a follow-up non-speech-based textual message; and....99

iii. [15.2] merging the follow-up speech-based textual message and the follow-up non-speech-based textual message to generate a follow-up query. 99

iv. [15.1] 102

v. [15.2] 103

d. Dependent Claim 17 105

i. [17.0] The method according to claim 13, further comprising generating a context stack that includes one or more contexts that are selected based on the query. 105

(A) Generating a context stack including contexts 105

(B) Contexts selected based on the query 106

e. Dependent Claim 18 109

i. [18.0] The method according to claim 17, wherein the one or more contexts are generated based on applying prior probabilities or fuzzy possibilities to (i) keyword matching, (ii) user profiles, (iii) a dialog history, or any combination of (i) to (iii). 109

X. OBJECTIVE INDICIA OF NON-OBVIOUSNESS 112

XI. CONCLUSION....112

CLAIM LISTING 114

EXHIBIT LIST

No.	Exhibit Description
1001	U.S. Patent No. 9,263,039
1002	File History of U.S. Patent No. 9,263,039
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	RESERVED
1013	RESERVED
1014	RESERVED
1015	RESERVED
1016	RESERVED
1017	RESERVED
1018	RESERVED
1019	RESERVED
1020	RESERVED
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 ("Ross")
1023	RESERVED
1024	RESERVED

1025	RESERVED
1026	Excerpts from Microsoft Computer Dictionary, 5th edition (2002)
1027	Cattaneo, Marco EGV. "Fuzzy probabilities based on the likelihood function." *Soft Methods for Handling Variability and Imprecision*. Springer Berlin Heidelberg, 2008.
1028	Shdaifat, I., Grigat, R.R. and Lütgert, S., 2001. Viseme recognition using multiple feature matching. In *INTERSPEECH* (pp. 2431-2434).
1029	EDTX Calendar, Judge Gilstrap
1030	RESERVED
1031	RESERVED
1032	RESERVED

TABLE OF ABBREVIATIONS AND CONVENTIONS

Abbreviation	Meaning
039 Patent	Ex.1001: U.S. Patent No. 9,263,039
IPR	*inter partes* review
Petitioner	Bank of America, N.A. ("BofA")
Patent Owner or PO	Dialect, LLC ("Dialect")
Second Amended Complaint	Compl.: Ex.1007
xx:yy–zz	column *xx*, lines *yy* to *zz*

1. I, Henry Houh, declare as follows:

I. INTRODUCTION

2. I am making this declaration in support of Petitioner's request for an *Inter Partes* Review of U.S. Patent No. 9,263,039 ("the 039 Patent") to Criso.

3. 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.

4. I have been asked to provide my opinions regarding whether the subject matter of claims 13-15 and 17-18 ("the Challenged Claims") of the 039 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.

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

- the 039 Patent, Ex.1001
- the prosecution history of the 039 Patent ("039 File History"), Ex.1002
- U.S. Patent No. 6,964,023 ("Maes"), Ex.1005
- 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.

6. 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.

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

8. 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 039 Patent. If requested, I am prepared to testify about my opinions expressed herein.

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

9. 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.

10. 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.

11. 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.

12. 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.

13. 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.

14. 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.

15. 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.

16. 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.

17. 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.

18. 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.

19. 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.

20. 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.

21. 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.

22. 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

23. 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 039 Patent. I have been asked to assume the date of the purported invention for the 039 Patent is August 5, 2005.

24. 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.

25. 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 039 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 039 Patent. Lack of professional experience can be remedied by additional education, and vice versa.

26. 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 039 Patent.

IV. RELEVANT LEGAL STANDARDS

27. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the 039 Patent, I am relying on certain basic legal principles that counsel has explained to me.

28. I understand that prior art to the 039 Patent includes patents and printed publications in the relevant art that predate the priority date of the 039 Patent. For purposes of this Declaration, I am applying August 5, 2005 as the priority date of the 039 Patent, but I take no position on that issue.

29. 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.

30. 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.

31. 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.

32. 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.

33. 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.

34. 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.

35. 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.

36. 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.

37. 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.

38. 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.

39. 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 039 PATENT

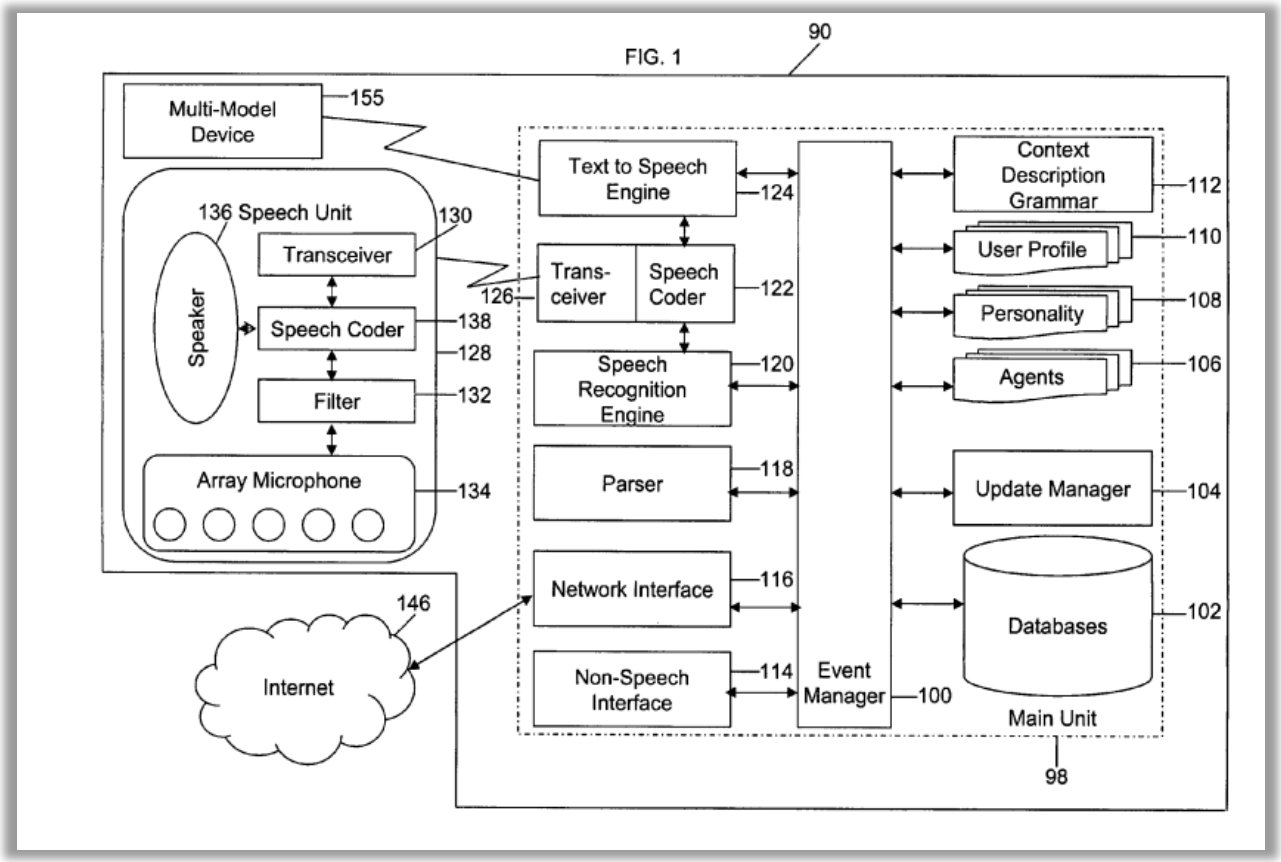
40. The 039 Patent was based on U.S. Patent Appl. No. 14/500,273 filed on September 29, 2014. Ex.1001, Cover.

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

A. The Disclosures of the 039 Patent

42. The 039 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, 1:25–35. The 039 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-speech communications and presenting the expected results for a particular question or command." Ex.1001, Abstract.

43. Figure 1 of the 039 Patent (reproduced below) "is an overall diagrammatic view according to one embodiment of the invention" (Ex.1001, 10:14–15):



Ex.1001, Figure 1. The 039 Patent describes a system 90 that includes speech unit 128, speech recognition engine 120, context description grammar module 112, parser 118, and agents 106. Speech unit 128 includes a microphone to receive a spoken utterance from a user. Ex.1001, 12:4–7. After initial processing, the speech is passed to speech recognition engine 120 for processing using the context

description grammar module 112. Ex.1001, 12:40–42. 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, 12:42–45. "The knowledge agents may then process the commands or questions." Ex.1001, 12:49–50. 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, 21:28–33. "The domain agent 156 scores the relevance of the results based on results already received, the context, the criteria, the history of the dialog, the user profile 110 and domain specific information using probabilistic or fuzzy scoring techniques. Part of the dialog history is maintained in a context stack." Ex.1001, 23:19–25.

VI. CLAIM CONSTRUCTION

44. 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

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

46. 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.

47. 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

48. 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 039 Patent.

49. 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.

50. 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.

B. Ross

51. 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 describing potential contexts 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] (disclosing "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.")

IX. CLAIMS 13-15 AND 17-18 ARE UNPATENTABLE

A. Ground 1: Maes and Ross render obvious claims 13-15 and 17-18

1. The Combination of Maes and Ross

52. It would have been obvious to include Ross's teachings of context grammars (including data/entries in such grammars), maintaining a prioritized list of context grammars in a context list, and comparing a spoken utterance[6] (or, equivalently a representation of a spoken utterance) against data in context grammars to identify matches between keywords/phrases of a context grammar and a spoken utterance, generating relevance scores/weights based on such comparisons, identifying context grammars (and associated applications/application programs) that are capable of accepting a given utterance, determining whether a given context grammar is capable of accepting a given spoken utterance, using context grammars to formulate/construct requests sent to applications (*see e.g.*, Ex.1022, Abstract, [0002], [0013], [0021], [0033]-[0038], [0052]-[0053], [0059]-[0060], and Figure 4

[6] As used herein, it will be understood that a "spoken utterance" would encompass both an audio portion (e.g., speech) and a video (e.g., non-speech) portion (such as the accompanying lip movement) of the utterance. Ex.1005, 6:43–47.

and 6) in Maes's system that receives and processes spoken utterances for the reasons explained below.

53. At the onset, the use of grammars and specifically, grammars that describe context for applications (i.e., context grammars) and used for comparing/matching decoded text from a user's spoken utterance was well-known in the art. For example, Ross explains that the grammars can be specified using the well-known "Backus Naur Form (BNF)." Ex.1022, [0013] and [0060]; *see also* Ex.1026, 49 (describing dictionary definition for "BNF"); Ex.1021, 15–16 (providing definitions of context-sensitive and context-free grammars). Further, Ross's teachings of testing/comparing a recognized utterance against data included in a context list storing context grammars for applications (Ex.1022, [0013], [0034]-[0037], [0053], and Figure 4) is strikingly similar to Maes's technique of associating the results of the recognized events included in the decoded text/script against data stored in an organized/sorted context stack. *See* Ex.1005, 7:60–8:4 and Figures 1 and 8. Further, Maes's "context stack" is similar to Ross's "context list." For example, Ross's description of maintaining a priority order in its context list in a manner such that the most-recently accessed application moves to the top of its context list (*see* Ex.1022, [0037]) is consistent with the well-known concept of a "stack," in which the most recent item added to the stack will be the first item considered from the stack in a later operation and termed in the art as "last in, first

out" (LIFO) methodology of processing a queue. *See* Ex.1026, 305 (dictionary definition for "last in, first out"); *compare with* 215 (dictionary definition for "first in, first out"). In view of the similarity of the techniques in Maes and Ross related to performing comparisons of spoken utterances against data stored in a context list/ stack, which were known in the prior art, a POSITA would have understood that combining the system of Maes and with the context grammar techniques of Ross would improve the context identification abilities of Maes's system (*see e.g.*, Ex.1005, 7:60–8:4, 8:37–42, and 37:53–55), and therefore provide at least one reason to combine these references.

54. Both Maes and Ross are analogous art to the 039 Patent for at least three reasons. *First*, both references are directed toward the same field of endeavor as the 039 Patent—computer-implemented systems interpreting user utterances. Ex.1001, Abstract, 3:17–37; Ex.1005, Abstract, 4:7–17; and Ex.1022, Abstract, [0021]. *See also* Ex.1005, 7:60–8:4 (describing an example in which Maes's system interprets the user's spoken utterance "turn it on"); Ex.1022, [0053] (describing an example in which Ross's system interprets the user's spoken utterance "print it"). *Second*, both Maes and Ross are reasonably pertinent to a problem described by the 039 Patent: an environment for reliably processing a user's language queries, which is a problem of conventional systems identified in the 911 Patent. Ex.1001, 1:55-61. Maes describes its invention as "provid[ing] an effective conversational computing

environment for one or more users" based on processing a user's multi-modal input in the form of audio input data (e.g., spoken utterance), video input data (e.g., lip movement and/or visual gesture), as well as other types of input data. Ex.1005, Abstract, 4:7–17 and 6:39–55. Ross describes a technique for determining a context associated with a user's recognized utterance that involves evaluating grammars against the user's recognized utterance. Ex.1022, [0033]-[0034] and Figure 5. *Third*, similar to the 039 Patent, both Maes and Ross teach the use of *grammars* and *vocabularies* in recognizing speech inputs. Ex.1001, 13:52–55; Ex.1005, 31:15–18, 34:61–67, 33:11–21, 39:59, 41:13–20, and 41:35–38; and Ex.1022, [0033]-[0035], Figure 4, and [0028].

55. A POSITA would have been motivated to combine the teachings of Maes and Ross as described above because Maes solves the problems specifically identified by Ross. Specifically, Ross describes the problem associated with conventional systems stating that different speech-enabled applications "may not share the same speech enabling interfaces" and "cannot make a determination which application should receive a particular speech utterance." Ex.1022, [0004].. For example, if a user "speaks a calendar command to set up an appointment that is received by a word processing application, then the user can experience an unexpected or undesirable result," which can lead to "wasted time and effort, as well as frustration." Ex.1022, [0004]. Maes solves these problems because Maes's

system includes use of "[an] I/O subsystem 12 … compris[ing] one or more microphones for capturing audio input data from the environment in which the system is deployed" and based on processing the captured audio input data, Maes's system "determines … which application(s) [*e.g.*, speech-enabled applications] should handle the user inputs." Ex.1005, 6:5–8 and 36:54–57. Thus, by employing a single I/O subsystem, for instance, "across all registered applications," Maes's system addresses Ross's concern that in the conventional practice, "individual, independent speech-enabled applications … each contend[] for control of the microphone, and each oblivious to the other." Ex.1005, 36:49–54; Ex.1022, [0005]. As such, a POSITA would have understood Maes's teachings to expand or otherwise improve the capabilities of Ross, and vice-versa.

56. A POSITA would have recognized that Ross's teachings related to "resolv[ing] … ambiguous phrase[s]" such as "print this" or "print it" in spoken utterances would be complimentary to Maes's system, which expressly seeks to provide ambiguity resolution, if needed, by "seek[ing] confirmation, disambiguation, correction, more details … until the intent is unambiguous and fully determined." Ex.1022, [0059]; Ex.1005, 36:59–63, 5:52–56 and Figure 2; *see also* 8:30–42 (describing an example of an in-vehicle system performing ambiguity resolution in connection with the potentially ambiguous spoken utterance "turn it on" because it identifies that there are likely other devices in the vehicle that could

be turned on). Furthermore, Ross describes maintaining a priority list of context grammars used in resolving ambiguities in a spoken utterance and identifying a target application for the spoken utterance. Ex.1022, [0035], [0053] and [0059]. "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 [within a prioritized list of grammars in context list 62]**." Ex.1022, [0034]-[0035]. "[T]he 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 [and then] 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]. Thus, a POSITA would have been motivated to combine Maes and Ross because Ross's teachings of priority list of context grammars when combined with Maes's system would benefit Maes's system by providing it with ambiguity resolution functionality. Ex.1022, [0059]; Ex.1005, 36:59–63, 5:52–56.

57. Maes also provides the function of determining a target application or appliance addressed by the user's spoken utterance. Ex.1005, 2:54–56. Maes explains "determin[ing] and execut[ing] 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. Accordingly, both Maes and Ross are directed at determining the target application to handle the task corresponding to the user's intention and/or react to the user activity based on processing/decoding a user's spoken utterance. Ex.1005, 2:54–56 and Ex.1022, [0013]. It would have been obvious to a POSITA to modify Maes's teachings of generalized representations of data (or, simply queries) generated as a result of merging the transcriptions (Ex.1005, 6:32–50) to additionally include Ross's teachings of context grammars so that the queries are compared against data in context grammars for that target application, per Ross's teachings. Ex.1022, [0034]-[0037].

58. Indeed, Ross expressly teaches that "a grammar is defined for each application." Ex.1022, [0035]. For example, in the combined Maes/Ross system, applications ("*domain agents*") that are considered as target application candidates would be accessed, as taught by Ross, to ascertain whether their associated context grammars are capable of accepting a recognized representation of the processed/decoded spoken utterance. Ex.1022, [0052]-[0053] and [0013]. One benefit (among others) of incorporating Ross's teachings of context grammars in Maes's system is that it would expand the vocabulary (i.e., more number of keywords and phrases) of Maes's system, which therefore would result in interpreting spoken utterances in a manner that is "closer in meaning" to the user

inputs. A POSITA would have thus recognized that Ross's teachings of context grammars (and their use in testing against decoded text/script corresponding to the user's spoken utterance) are compatible with Maes's system in many ways, and therefore would have had a reasonable expectation of success for combining them and without undue experimenation. For instance, the "applications" in Ross and "grammar[s] defined for each application" are "speech-enabled" and Maes is directed at speech processing methods and systems. Ex.1022, [0035]-[0038]; Ex.1005, 4:7–16 and Figures 1 and 4.

59. Additionally, a POSITA would have had a reasonable expectation of success in implementing Ross's teachings with Maes's system, because doing so would involve minimal changes to Maes's system architecture and the combined system uses the hardware and techniques already described in each reference, and each system would perform the same in the combined system. For instance, a POSITA would recognize that Ross's teachings related to use of context grammars for applications, e.g., maintaining a list of context grammars according to a certain priority would merely be a few additional process steps for "grammars" or "grammar database" used in Maes's system. Ex.1005, 31:15–18, 34:61–67, 33:11–21, 39:59, 41:13–20, and 41:35–38; and Ex.1022, [0033]-[0035], Figure 4, and [0028]. And, therefore, in one implementation, a POSITA would have been 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." 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.

60. 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 Maes/Ross 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) and the outcome of prior comparisons would be used to generate a recency of

access characteristic (e.g., "*relevance score*") used for maintaining the priority order of Ross's context list. Ex.1002, [0034]-[0035] and [0053]-[0054]. Such access characteristics would allow tracking ("*selecting*") recently accessed applications and obtaining relevant content from the recently accessed applications, in the Maes/Ross system.

61. A POSITA would have also understood that the Maes/Ross system would have predictably resulted in Maes's dialog manager additionally performing the step of testing the decoded text/script processed from the spoken utterance against entries in context grammars, per Ross. Ex.1022, [0013] and [0034]-[0035]. Ross explains a spoken phrase (e.g., a voice command or other speech input) from a user sent to a microphone connected to "the computer system," such as for example, Maes's system (which includes the dialog manager). Ex.1022, [0021]; Ex.1005, Figure 1. Maes already describes its dialog manager associating the results of the recognized events included in the decoded text/script (produced as a result of the transcription) against data stored on the context stack. *See* Ex.1005, 37:53–55, 7:60–8:4, 8:37–42, and Figure 1. Adding the extra step of testing the decoded text/script against entries in Maes's and/or Ross's grammar would improve the dialog manager's ability to identify contexts. A POSITA would have the ability to make the combination cited above and would have further understood these options as obvious-to-try and therefore had good reason to pursue them.

62. A POSITA would have been motivated to combine Maes and Ross because it would have involved the application of known techniques (e.g., identifying contexts) to improve a similar system in the same way. Maes already describes using a context stack "organized/sorted context corresponding to each active dialog" for storing "all the information associated with an application." Ex.1005, 37:55–56 and 37:60-61. Similar to Maes's context stack (Ex.1005, context stack 817 in Figure 8 or context stack 20 in Figure 1), Ross uses a "context list" (Ex.1022, context list 62 in Figure 4). For example, Ross teaches "maintaining an ordered list of applications" in context list 62 such that "[w]henever an application 26 gains window focus, it will move to the head of the list 62. Likewise, whenever an application 26 which is not the top priority application 26 is chosen as the target for a speech command, the application 26 indicated by the selected context 72 will move to the head of the list 62. In this way, the last application 26 that the user touched or talked to will get the first opportunity at interpreting the next user utterance, and the other applications 26 will be ordered in a most-recently-accessed way." Ex.1022, [0035]. A POSITA would have recognized that Ross's technique of maintaining a priority list of context grammars for applications in a manner that results in the most-recently accessed application to move to the top of the list would greatly benefit Maes's system because it would allow efficient identification of

context by retrieving the most-recently accessed application in the modified context stack of the combined Maes/Ross system.

63. Furthermore, a POSITA would have been motivated to combine Maes 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 (describing Maes's system comprises at least one processor and memory operatively coupled to the at least one processor), 45:50–60 (describing that "the elements illustrated in FIGS. 1 through 9C may be implemented in various forms of hardware, software, or combinations thereof"); 12:43–49 (disclosing 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.1022, [0002] (describing commercially-available speech recognition products that convert speech into text strings that can be utilized by software applications on a computer system include the ViaVoice™ product from IBM®, Armonk, N.Y., and NaturallySpeaking Professional from Dragon Systems, Newton, Mass.), [0025] (describing use of Microsoft® Active Accessibility® (MSAA) from Microsoft Corporation, Redmond, Wash).

64. A POSITA would be motivated to combine Maes with 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 or phrases included in the utterance) and yielding no more than what one would expect from such an arrangement (an improved computer-implemented system interpreting user utterances), as Maes demonstrates. *See*, Ex.1005, 2:54–67.

2. Invalidity Analysis of claims 13-15 and 17-18

a. Independent Claim 13

i. [13.0] A method of processing speech and non-speech communications, comprising:

65. Maes teaches or suggests "*[a] method of processing speech and non-speech communications, comprising*:"

66. Maes describes "[s]ystems and methods are provided for performing focus detection, referential ambiguity resolution and mood classification **in**

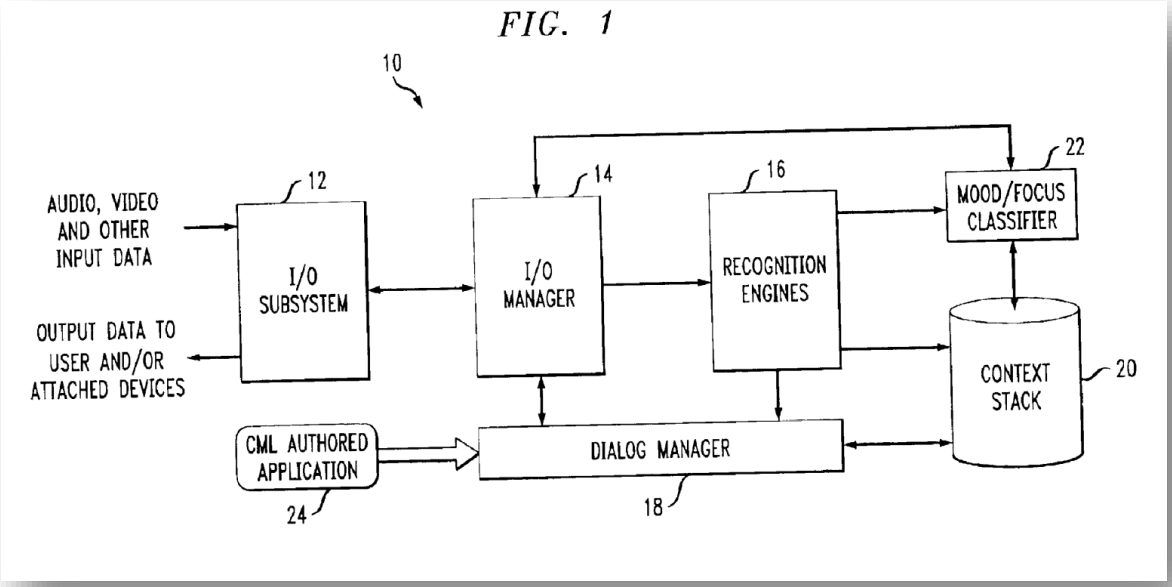
accordance with multi-modal input data" and, more particularly, to "systems and methods for performing focus detection, referential ambiguity resolution and mood classification in accordance **with multi-modal input data**. … Systems that employ such 'multi-modal' input techniques have inherent advantages over systems that use only one data input mode." Ex.1005, Abstract and 1:9–23.

67. Further, Maes's systems and methods are directed at "*processing speech and non-speech communications*." Referencing Figure 1, 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.

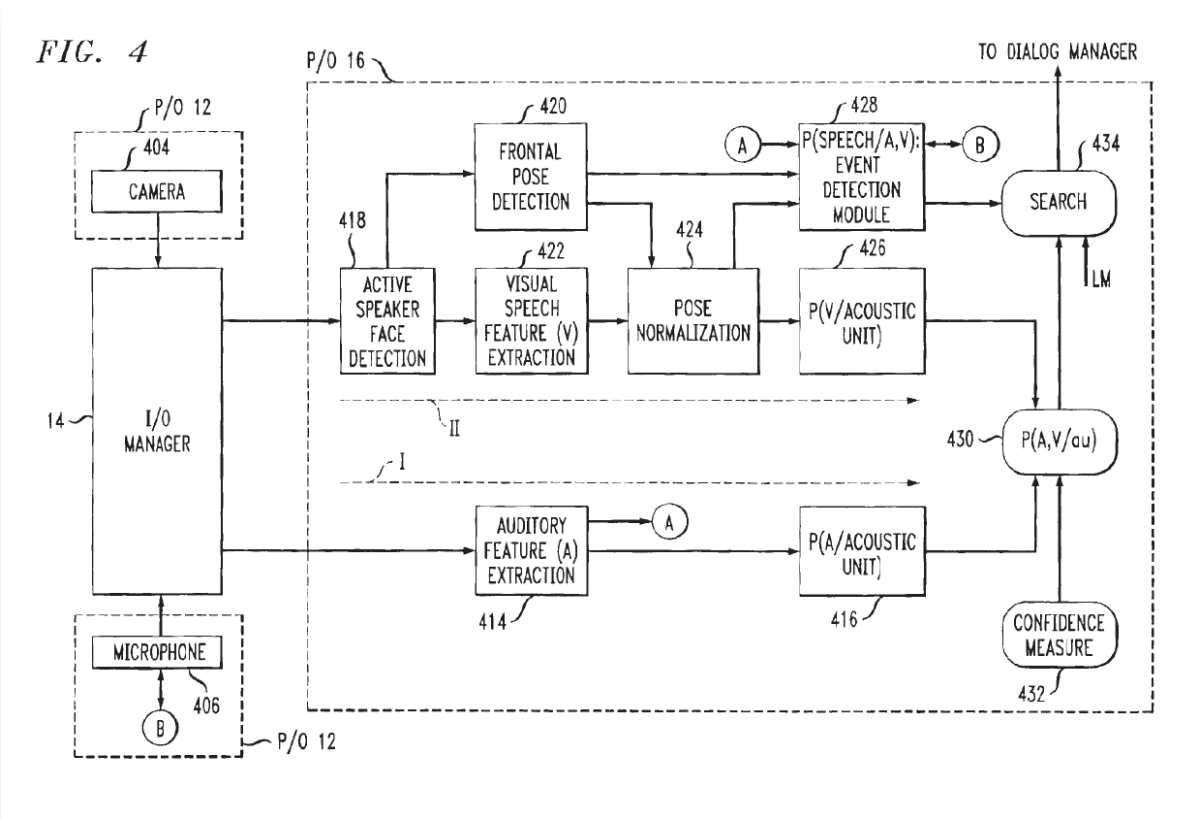


Ex.1005, Figure 1.

68. Maes describes Figure 4 as "an audio-visual speech recognition module[7] that may be employed as one of the recognition modules[8] 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) ("*speech*") and a video signal (e.g., the speaker's face including lip movement and/or background objects in the environment) ("*non-speech communications*") and processes the multi-modal input data (e.g., "*speech and non-speech communications*"). Video signals are processed by blocks/modules 418, 422, 424, 426 and audio signals are processed by blocks/modules 414, 416.

[7] As used herein, the terms "Maes's system" and "audio-visual speech recognition module" are used interchangeably.

[8] 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."



Ex.1005, Figure 4.

69. Therefore, Maes discloses "*[a] method of processing speech and non-speech communications.*"

70. Maes explains that "multi-modality" "may comprise a combination of other modalities other than voice and video. For example, multi-modality may include keyboard/pointer/mouse (or telephone keypad) and other sensors, etc. Thus, a general principle of the present invention of the combination of **modality through at least two different sensors** (and actuators for outputs) to disambiguate the input,

and guess the mood or focus, can be generalized to any such combination. Engines or classifiers for determining the mood or focus will then be specific to the sensors but the **methodology** of using them is the same as disclosed herein." Ex.1005, 3:4–11.

71. Accordingly, in Maes, "**computer software including instructions or code for performing the methodologies of the invention**, as described herein, may be stored in one or more of the associated memory devices (e.g., ROM, fixed or removable memory) and, when ready to be utilized, loaded in part or in whole (e.g., into RAM) and executed by a CPU. … [T]he 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, 45:45–60.

72. Maes describes examples in which its methods are applicable to a broadcast news system, which further confirms that its method applies to "*speech and non-speech communications*." 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.

73. As another example, Maes states that "the multi-modal conversational computing system … 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, 6:43–50, and 7:63–8:29.

74. Therefore, Maes teaches or suggests this claim element for this additional reason.

ii. [13.1] receiving the speech and non-speech communications;

75. Maes discloses "*receiving the speech and non-speech communications*."

76. For example, Maes describes Figure 4 (reproduced below) as "an audio-visual speech recognition module that may be employed as one of the recognition modules of FIG. 1 **to perform speech recognition using multi-modal input data received** in accordance with the invention." Ex.1005, 10:50–52. Specifically, Maes's Figure 4 discloses "an audio-visual speech recognition module" that

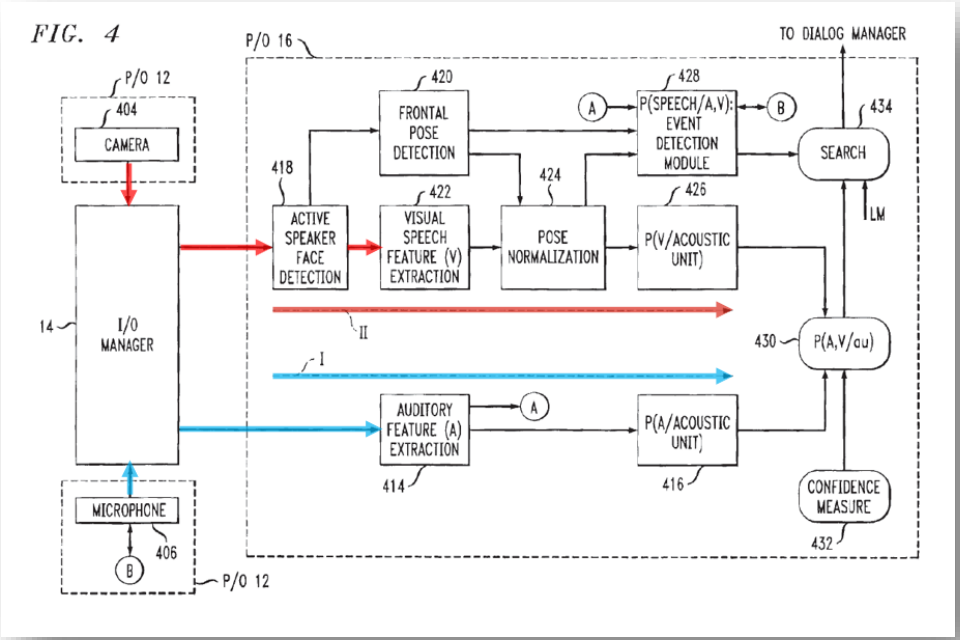
"receive[s]" ("*receiving*") multi-modal input data—such as an audio signal and a video signal—from a speaker ("*the speech and non-speech communications*") at an "input/output (I/O) subsystem … compris[ing] one or more microphones for capturing audio input data," which "one of ordinary skill in the art will realize that other user interfaces and devices … may be included **for capturing user activity**." Ex.1005, 10:47–11:23, 6:21–29, and 6:5–8.

77. Maes also discloses that the "multi-modal input data" includes "an audio signal" (e.g., speech provided by the speaker and/or background noise) ("*speech*") "*and*" "a video signal" (e.g., the speaker's face including lip movement and/or background objects in the environment) ("*non-speech communications*"). Ex.1005, 11:8–63. Maes's system "receives multi-modal input in the form of **audio input data**, **video input data,** as well as **other types of input data** …." Ex.1005, 4:7–15; Ex.1005, 4:67–5:5 ("system 10 … **taking both audio input data and image data input and processing it**"):

78. With reference to Figure 4[9], Maes depicts "[a] phantom line denoted by Roman numeral I represents the processing path the audio information signal takes

[9] There appears to be a typographical error in Figure 4—the labels "P/O" are incorrect and should have been denoted "I/O," e.g., "I/O subsystem 12." Ex.1005, 4:8–16 and Figure 1 confirm this.

within the module, while a phantom line denoted by Roman numeral II represents the processing path the video information signal takes within the module." Ex.1005, 11:55–63. Paths I and II are annotated in blue and red in Figure 4 (below).



Ex.1005, Figure 4 (annotated).

79. Maes explains "receiving real-time arbitrary content from a video camera 404 and microphone 406 via the I/O manager 14 ... While the video signals received from the camera 404 and the audio signals received from the microphone 406 are shown in FIG. 4 as not being compressed, they may be compressed and therefore need to be decompressed in accordance with the applied compression scheme." Ex.1005, 11:23–31. Specifically, the feature extractor 414 of the audio-visual speech recognition module "**receives an audio or speech signal**" and the

active speaker face detection module 418 "**receives video input [from] camera 404**" (e.g., "the visual portion (e.g., lip movement) of the utterance," (Ex.1005, 6:46-47)). Ex.1005, 11:64–12:2, 12:49–61.

80. Maes explains in detail what constitutes the "audio signal" ("*speech*") and the "video signal" ("*non-speech communications*"). For example, the "audio signal" ("*speech*") can include "silence," "speech," and "noise." Ex.1005, 13:3–15. And, furthermore, with regards to the "video signal" ("*non-speech communications*") comprising "the visual portion (e.g., lip movement) of the utterance," (Ex.1005, 6:46–47), Maes describes that:

> the video signal captured by the camera 404 can be of any particular type. As mentioned, the face and pose detection techniques may process images of any wavelength such as, e.g., visible and/or non- visible electromagnetic spectrum images. By way of example only, this may include infrared (IR) images (e.g., near, mid and far field IR video) and radio frequency (RF) images. Accordingly, the module may perform audio-visual speech detection and recognition techniques in poor lighting conditions, changing lighting conditions, or in environments without light. For example, the system may be installed in an automobile or some other form of vehicle and capable of capturing IR images so that improved speech recognition may be performed. Because video information (i.e., including visible and/or non-visible electromagnetic spectrum images) is used in the speech recognition process, the system is less susceptible to recognition errors due to noisy conditions, which significantly hamper conventional recognition systems that use only audio information. In addition, due to the methodologies for processing the visual information described herein, the

> module provides the capability to perform accurate LVCSR (large vocabulary continuous speech recognition).

Ex.1005, 11:32–54.

81. Further, Maes explains that, in accordance with its invention, "**multi-modality** … **may comprise a combination of other modalities other than voice and video**. For example, multi-modality may include keyboard/pointer/mouse (or telephone keypad) and other sensors, etc. Thus, a general principle of the present invention of the combination of modality through at least two different sensors (and actuators for outputs) to disambiguate the input, and guess the mood or focus, can be generalized to any such combination. Engines or classifiers for determining the mood or focus will then be specific to the sensors but the methodology of using them is the same as disclosed herein." Ex.1005, 3:1–12. Maes's system therefore complements conventional interfaces and user input/output rather than replacing them and as Maes describes "[t]his is the notion of "multi-modality" whereby speech, and video … may be used in parallel with a mouse, keyboard, and other input devices such as a pen." Ex.1005, 31:61–66.

82. Additionally, with reference to the Figure 1[10], Maes discloses 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"). Ex.1005, 4:7–15. "The system 10 therefore performs referential ambiguity resolution with respect to multiple users **by taking both audio input data and image data input and processing it** to make a user resolution determination. This may include detecting speech activity and/or the identity of the user based on both audio and image cues." Ex.1005, 4:67–5:5.

83. Maes explains how the multi-modal input comprising the audio signal ("*speech*") and video signal ("*non-speech communications*") are received.

> As mentioned above, **the data input portion of the subsystem 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**. The cameras/sensors may be capable of capturing not only visible image data (images in the visible electromagnetic spectrum), but also IR (near,

[10] Maes identifies Figure 4 as "a preferred embodiment of an audio-visual speech recognition module" that may be employed as one of the recognition modules of the Figure 1 embodiment. Ex.1005, 10:47–52.

> mid and/or far field IR video) and/or RF image data. Of course, in systems with more than one camera, different mixes of cameras/sensors may be employed, e.g., system having one or more video cameras, one or more IR sensors and/or one or more RF sensors.
>
> In addition to the one or more cameras, **the I/O subsystem 12 may comprise one or more microphones for capturing audio input data from the environment in which the system is deployed**. Further, the I/O subsystem may also include an analog-to-digital converter which converts the electrical signal generated by a microphone into a digital signal representative of speech uttered or other sounds that are captured. Further, the subsystem may sample the speech signal and partition the signal into overlapping frames so that each frame is discretely processed by the remainder of the system.

Ex.1005, 5:60–6:15.

84. Maes describes an example of how the disclosed system works. For example, Maes discloses the "*receiving* ..." step by providing an example in which 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 …. 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." 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.

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

iii. [13.2] transcribing the speech and non-speech communications to create a speech-based textual message and a non-speech-based textual message;

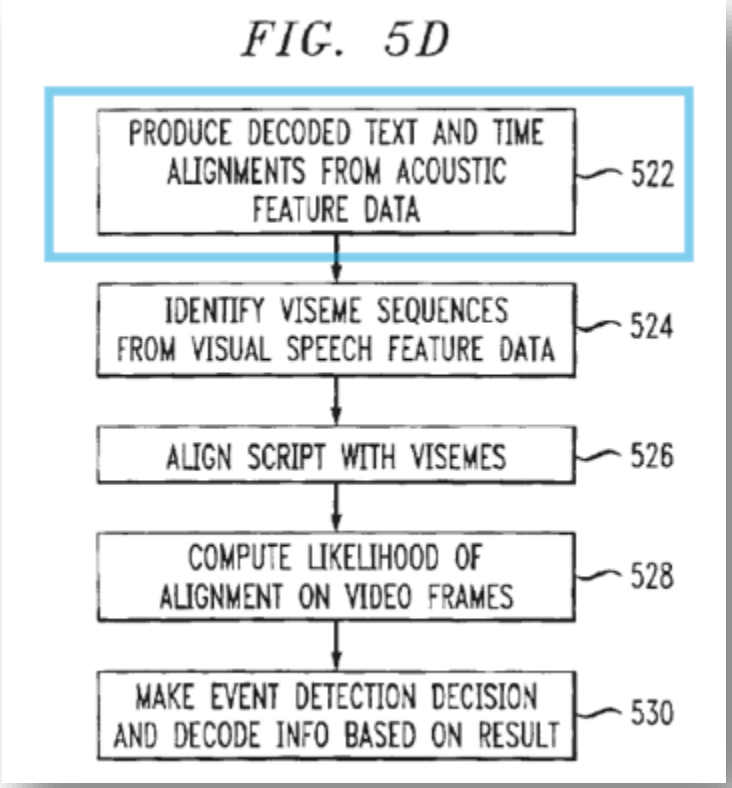
86. Maes teaches or suggests this claim element.

(A) Transcribing the speech communications

87. Maes teaches "*transcribing the speech … communications to create a speech-based textual message.*"

88. With reference to Figure 5D (reproduced below), Maes discloses classical speech recognition techniques processing the audio signal including the spoken utterance ("*transcribing the speech … communications*") "*to*" "produce"

("*create*") "decoded text" (or, script) ("*a speech-based textual message*") "using the feature data from the acoustic feature extractor 414." *See generally* Ex.1005, 21:43–47.



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 understand that the "decoded script" described in Maes is essentially the tangible

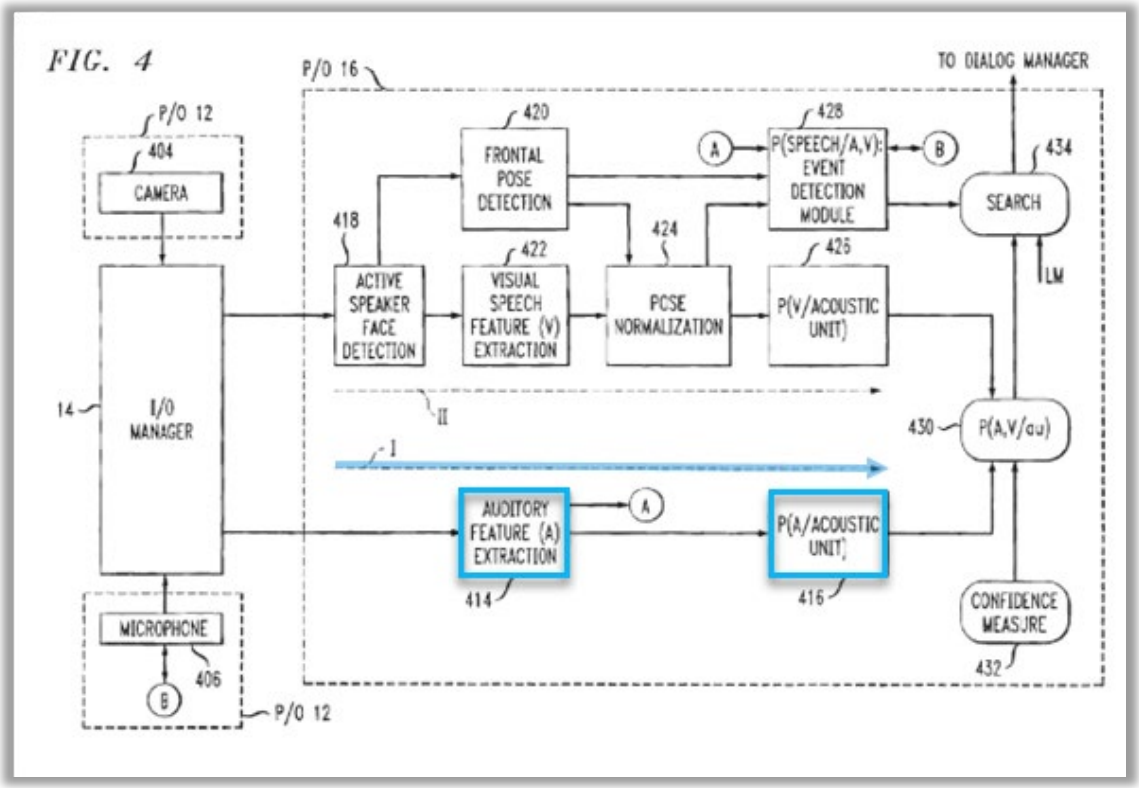
result of a speech recognition system's interpretation of spoken language and provides a text representation ("*a speech-based textual message*") of the multi-modal content including the audio signal further comprising the spoken utterance / speech. E.g., Ex.1005, 20:61-64 ("If the buffered data is tagged 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, so as to yield a **decoded output**."); *see also* 20:23-33.

89. With reference to Figures 1 and 4 (reproduced below), Maes provides additional details explaining the "*transcribing ...*" step: Maes describes Figure 4 as "an audio-visual speech recognition module that may be employed as **one of the recognition modules[11] of FIG. 1 to perform speech recognition using [received] multi-modal input data**." Ex.1005, 10:50–52.

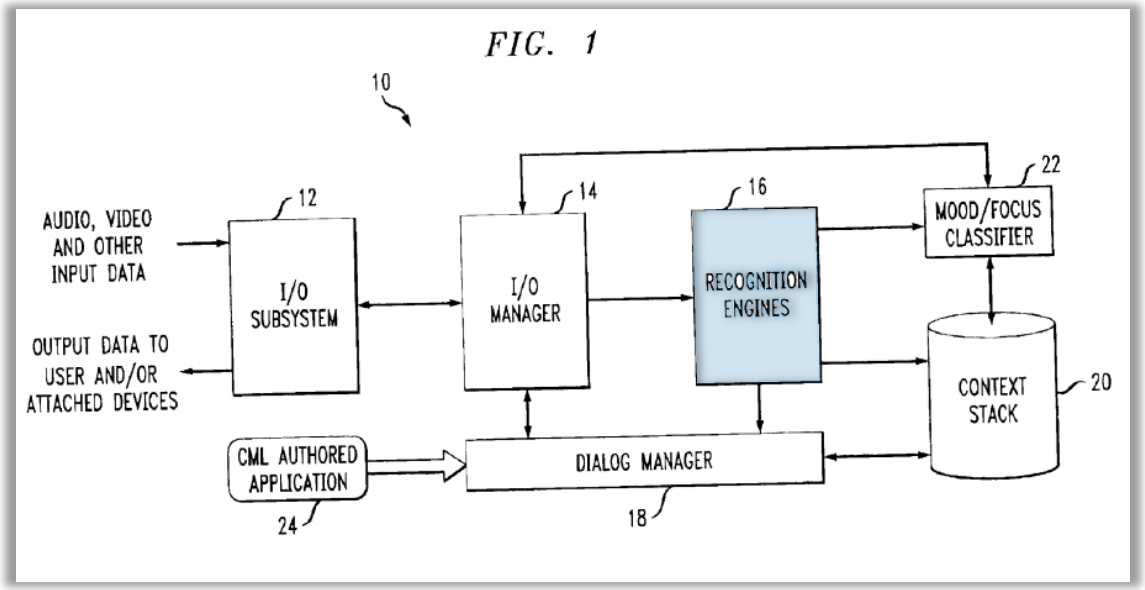
90. Specifically, Figure 4 shows that the audio-visual speech recognition module receives multi-modal input data including an audio signal (e.g., speech provided by the speaker and/or background noise) and a video signal (e.g., the speaker's face including lip movement and/or background objects in the environment). For example, the auditory feature extractor 414 "receives an audio or

[11] 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."

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 in Figure 4, the acoustic feature vectors are 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.



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



Ex.1005, Figure 1 (annotated).

91. 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.

92. 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.

93. 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.

94. 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." Ex.1005, 12:43–49. One such "acoustic information recognition system" that provides additional details on how speech is processed in Maes is described with reference to Figure 9A, explained below.

95. Referencing Figures 1 and 9A, Maes provides details of the "*transcribing* ..." step explaining 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," including "a dialog management unit 902 for conduct[ing] a conversation with the user," that provides functionality including natural language understanding (NLU), natural language generation (NLG), finite state grammar (FSG), and/or text-to-speech Syntheses (TTS) for machine-prompting the user…." Ex.1005, 39:22–29, 41:13–20, and Figure 9A.

96. Maes continues: "Apparatus 900 … includes a processing module 910" further including "**a speech recognizer 926** which … include[s] **a speech recognition module 928**," and "a speech prototype, language model and grammar database". Ex.1005, 39:59, 41:35–38. "Apparatus 900 can further include **a post processor 938** … **configured to transcribe user utterances and … perform keyword spotting thereon**. … Post processor 938 can employ speech recognizor 926" and "can also include a semantic module (not shown) to interpret meaning of phrases. The semantic module could be used by speech recognizor 926 to indicate that some decoding candidates in a list are meaningless and should be

discarded/replaced with meaningful candidates." Ex.1005, 41:48-64. Ex.1005, 41:48–64.

97. Maes explains that "the processing performed in blocks 414 and 416" in Figure 4 "produces" ("*creat[es]*"), for instance, "**decoded text**" ("*a speech-based textual message*") (Ex.1005, 21:43–47), and further, referencing Figure 9, Maes explains that the "semantic module" functions to "indicate that some **decoding candidates** in a list are meaningless and should be discarded/replaced with meaningful candidates" (Ex.1005, 41:61–64). Thus, Maes explains that the "**decoded text**" produced by the processing performed in blocks 414 and 416 is the same "decoding candidates" Maes describes in connection with Figure 4, further confirming Figure 9A and its corresponding disclosure provide additional details to, among other things, the system described in Figure 4. For example, in Figure 9A, Maes identifies "semantic module," "speech recognizor 926," "post processor 938," "speech recognition module 928" and other components/functionalities.

98. Therefore, Maes discloses the "*transcribing ...*" step.

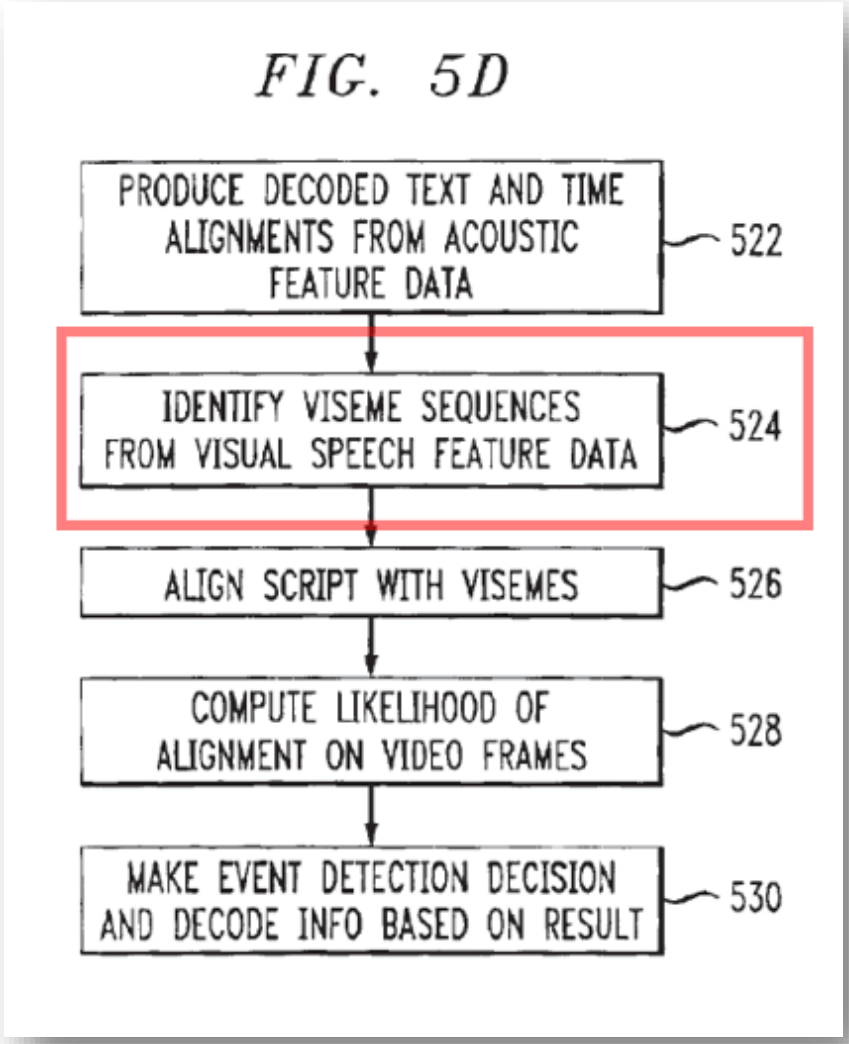
99. Referencing the in-vehicle system example, Maes explains the "*transcribing ...*" step as follows. The raw multi-modal input data relating to an I/O event (e.g., representative of the user's spoken utterance and lip movement accompanying the spoken utterance) received by the multi-modal conversational computing system is abstracted into a recognizable form. For example, "[t]he 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** ... 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**." Ex.1005, 6:21–38. Thus, as a result of processing by several modules in Maes's system, the system produces decoded text/script corresponding to the user's spoken utterance, for example, of saying "turn it on."

(B) Transcribing the non-speech communications

100. Maes teaches "*transcribing the ... non-speech ... communications to create ... a non-speech-based textual message.*"

101. For example, Maes depicts an audio-visual speech recognition module in Figure 4 comprising a visual speech feature extractor 422 that "extracts visual speech feature vectors (e.g., mouth or lip-related parameters) by processing the video signal ("*transcribing the ... non-speech communications*"), e.g., the face detected in the video frame including the user's lip movement "*to*" "produce" ("*create*") "a visual phonemes (visemes) sequence" ("*a non-speech-based textual message*"). Ex.1005, 17:51–55, 21:47–50. Maes discloses the "*transcribing ...*" step as a processing step in block 524 in Figure 5D.



Ex.1005, Figure 5D (annotated).

102. Maes further explains modules 422, 424 and 426 in Figure 4 extracting, normalizing, and labeling ("*transcribing*") to create visemes, e.g., in a visemes sequence ("*a non-speech-based textual message*") using the extracted visual speech vectors.

> The extracted visual speech feature vectors [produced by visual speech feature extractor 422] **are then normalized in block 424 with respect to the frontal pose estimates generated by the detection module 420**. **The normalized visual speech feature vectors are then provided to a probability module 426.** 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.** … **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, 18:42–63.

103. "As is known, visemes, or visual phonemes, are generally canonical mouth shapes that accompany speech utterances which are categorized and pre-stored similar to acoustic phonemes." Ex.1005, 29:27–30. A POSITA would have understood that visemes sequence ("*a non-speech-based textual message*") generated in Maes is essentially a string of visemes that is the tangible result of a visual recognition system's interpretation of a user's lip movement accompanying a spoken utterance and provides a visual representation of the non-speech content. For example, Shdaifat describes visemes as composed of features. Ex.1028, 1 ("The different features of the visemes are stored in a template").

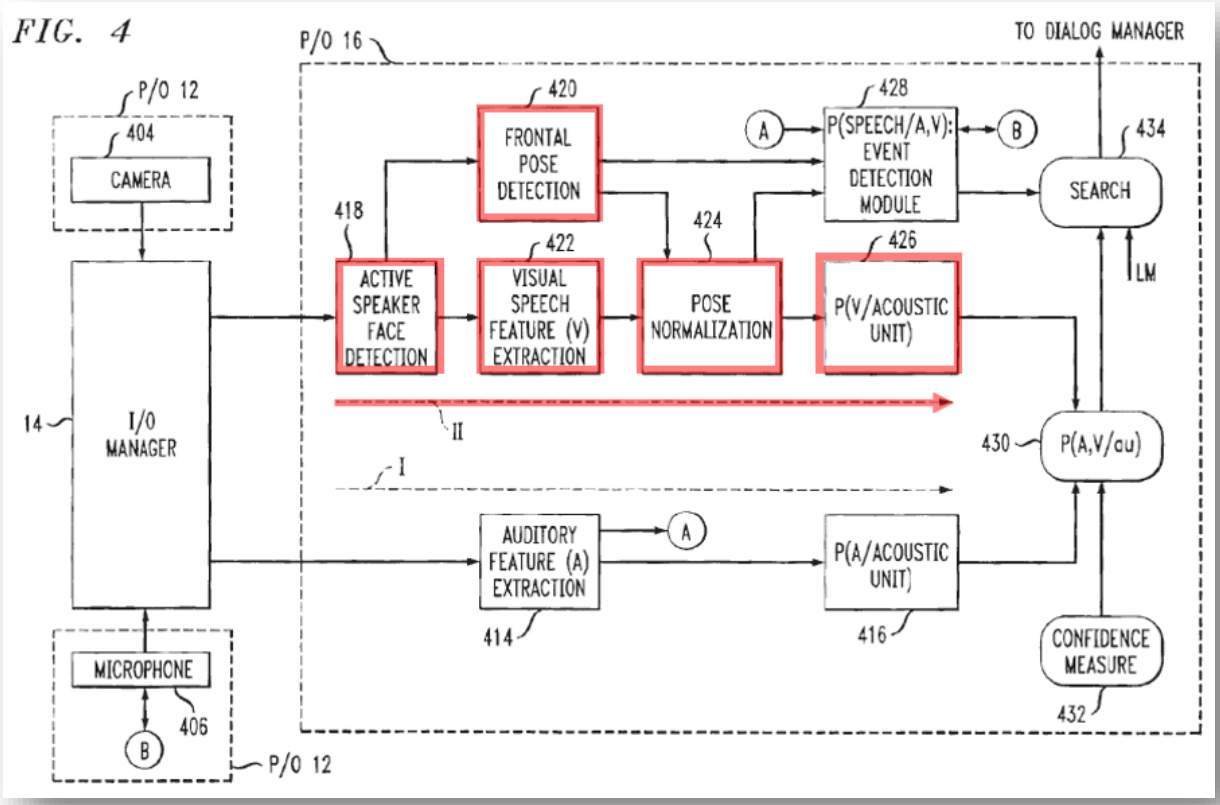
Figure 5: Features of the one viseme.

Ex.1028, Figure 5. Shdaifat further describes that visemes can be represented by images:

Figure 1: An example of images of five viseme classes.

Ex.1028, Figure 1. Referencing Figure 1, Shdaifat explains that "[f]or each viseme class the features of the lips are extracted and stored in templates. Each template contains the two images of the left and right corners of the mouth, the edge image of the viseme image, line segments of 10 pixel length fitted to the edges." Ex.1028, 2.

104. In Figure 4, the visual speech feature vectors are denoted by the letter "V" and "a phantom line denoted by Roman numeral II represents the processing path the video information signal takes within the module." Ex.1005, 23:1–5.



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

105. Maes provides several examples of visual speech features. "Examples of visual speech features that may be extracted are grey scale parameters of the mouth region; geometric/model based parameters such as area, height, width of mouth region; lip contours arrived at by curve fitting, spline parameters of

inner/outer contour; and motion parameters obtained by three dimensional tracking. Still another feature set that may be extracted via module 422 takes into account the above factors." Ex.1005, 17:55–63.

106. In addition, Maes describes multiple ways of extracting visual speech feature vectors.

> [W]hile the visual speech feature extractor 422 may implement one or more known visual feature extraction techniques, in one embodiment, the extractor extracts grey scale parameters associated with the mouth region of the image. Given the location of the lip comers, after normalization of scale and rotation, a rectangular region containing the lip region at the center of the rectangle is extracted from the original decompressed video frame. Principal Component Analysis (PCA), as is known, may be used to extract a vector of smaller dimension from this vector of grey-scale values."

Ex.1005, 18:1–11.

> Another method of extracting visual feature vectors that may be implemented in module 422 may include extracting geometric features. This entails extracting the phonetic/visemic information from the geometry of the lip contour and its time dynamics.

Ex.1005, 18:12–20. And as explained earlier, Maes describes that the visual speech feature vectors from the visual feature extractor 422 (along with other modules depicted in Figure 4) are used to "produce a visual phonemes (visemes) sequence" ("*create … a non-speech-based textual message*"). Ex.1005, 21:43–50.

107. Therefore, Maes discloses "*transcribing the … non-speech … communications to create … a non-speech-based textual message*."

108. Referring to the example in which Maes's multi-modal conversational computing system is employed within a vehicle, Maes explains the "*transcribing* ..." step in that example as follows. The raw multi-modal input data relating to an I/O event representative of the user's spoken utterance and lip movement accompanying the spoken utterance received by the multi-modal conversational computing system is abstracted into a recognizable form. For example, "[t]he 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. 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**." Ex.1005, 6:21–38. Thus, as a result of processing by several modules in Maes's system, the system produces a visemes sequence corresponding to the lip movement accompanying the spoken utterance of the user. *See also* Ex.1005, 6:43–50.

109. Accordingly, Maes teaches or suggests this element.

iv. [13.3] merging the speech-based textual message and the non-speech-based textual message to generate a query;

110. Maes teaches or suggests this claim element.

(A) Merging the speech-based and non-speech-based textual messages

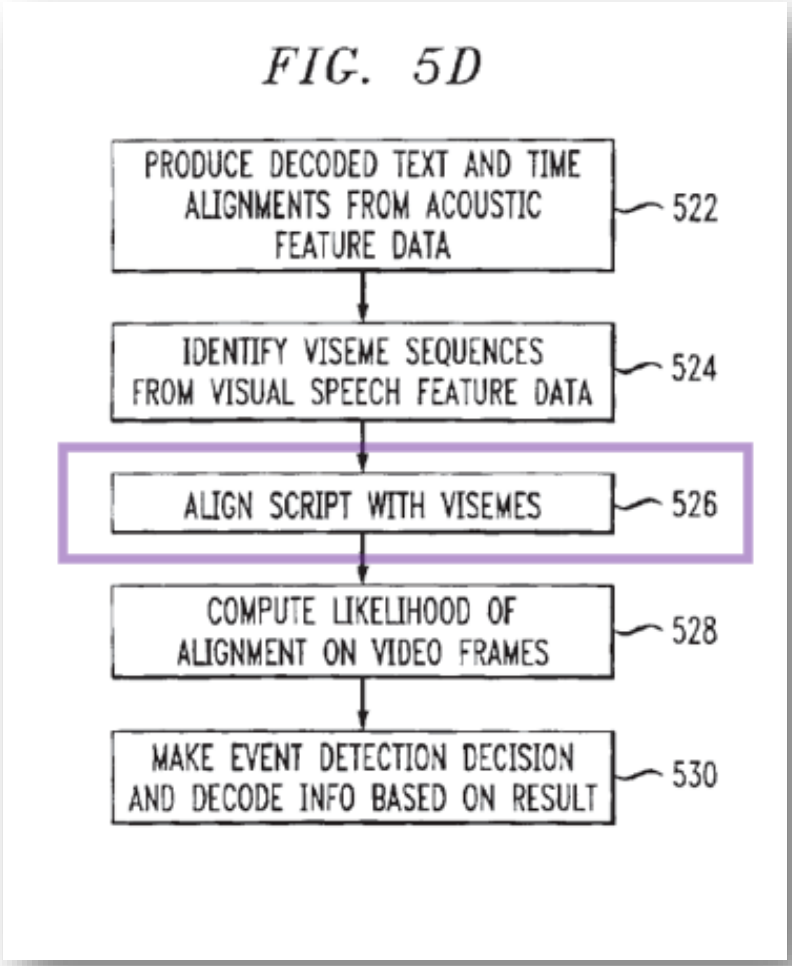
111. First, Maes teaches or suggests "*merging the speech-based textual message and the non-speech-based textual message*."

112. As explained earlier in connection with element [13.2], Maes discloses "the visual phonemes (visemes) sequence" ("*the non-speech-based textual message*") and "decoded text" (or, script) ("*the speech-based textual message*").

113. Referencing step 526 of Figure 5D, Maes discloses:

> the script is aligned with the visemes. A rapid (or other) alignment may be performed in a conventional manner in order to attempt to **synchronize the two information streams [i.e, the audio and video information streams]**.

Ex.1005, 21:51–54.



Ex.1005, Figure 5D (block 526).

114. Maes explains the purpose of aligning/synchronizing the audio and video information streams. "A goal associated with utterance verification is to make **a determination that the speech used to verify the speaker in the audio path I and the visual cues used to verify the speaker in the video path II correlate or align. This allows the system to be confident that the speech data that is being**

used to recognize the speaker is actually what the speaker uttered." Ex.1005, 29:30–36.

115. Further, Maes explains that aligning/synchronizing the audio and video information streams has "many advantages." "For example, from the utterance verification, it can be determined whether the user is lip synching to a pre-recorded tape playback to attempt to fool the system. Also, from utterance verification, errors in the audio decoding path may be detected. Depending on the number of errors, a confidence measure may be produced and used by the system." Ex.1005, 29:37–43.

116. Therefore, by teaching synchronization of the two information streams (i.e, the audio and video information streams) or, more specifically, alignment of "the decoded text" (or, script) ("*the speech-based textual message*") with "the visual phonemes (visemes) sequence" ("*the non-speech-based textual message*"), Maes discloses "*merging the speech-based textual message and the non-speech-based textual message*."

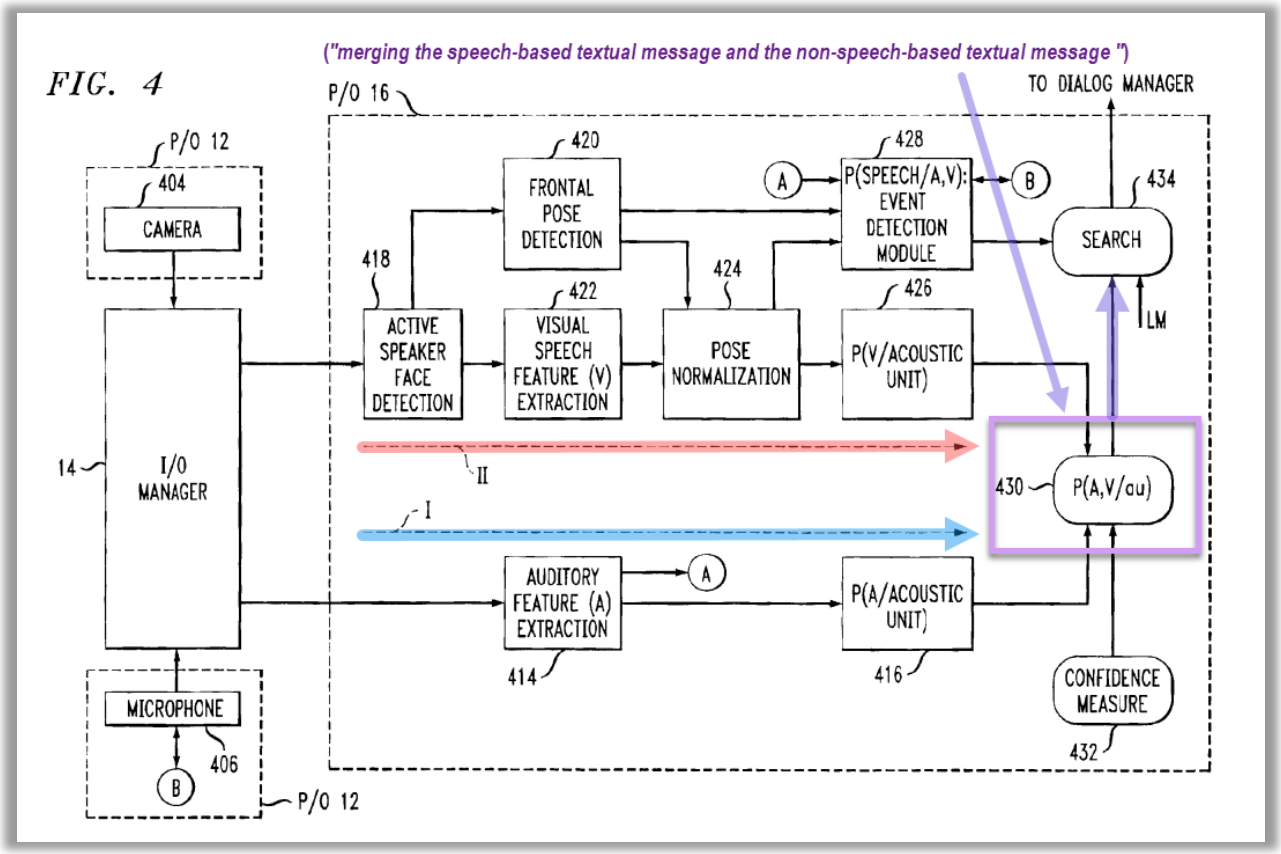
117. Referencing the example of an in-vehicle system, Maes discloses:

> 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.

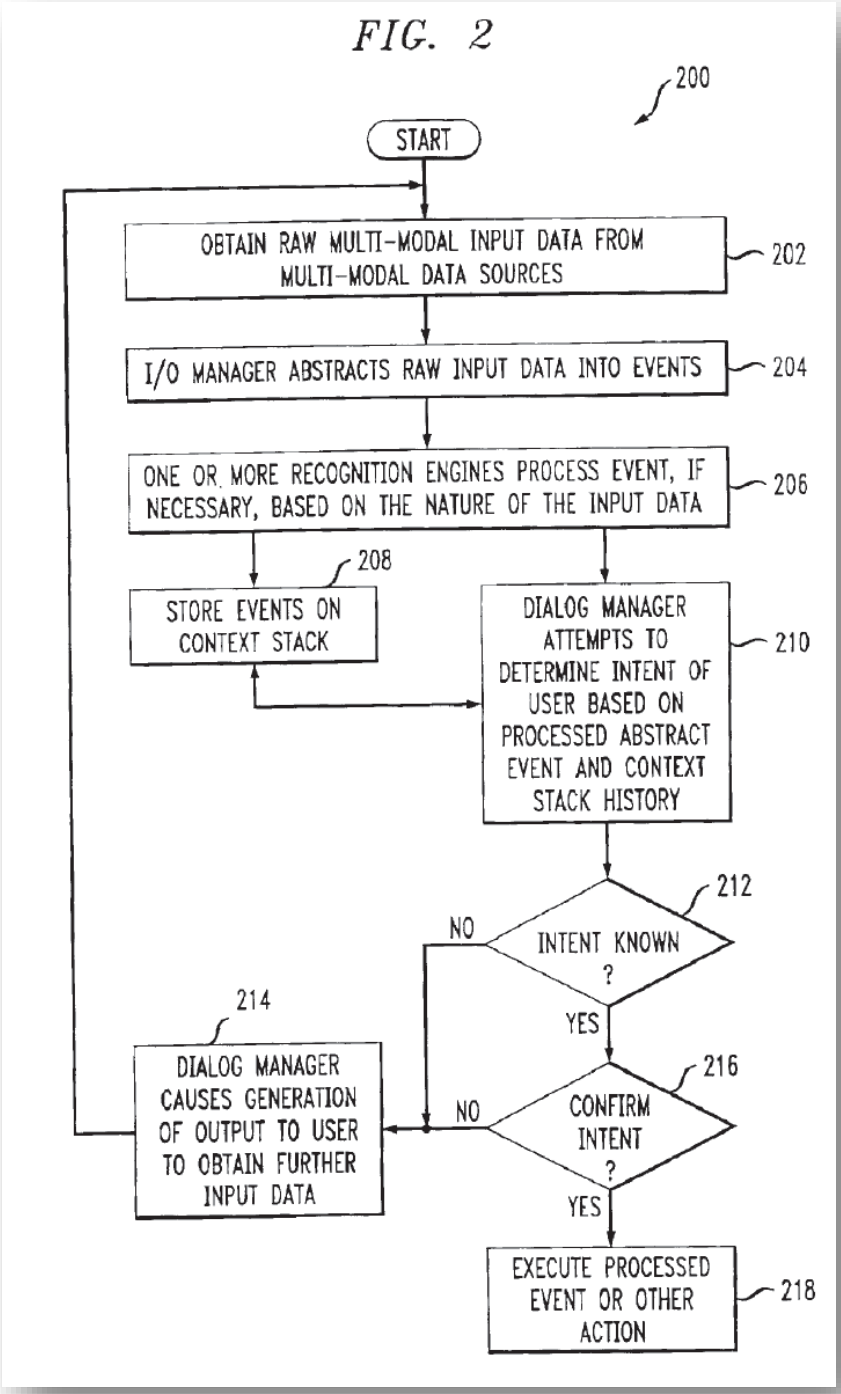
Ex.1005, 7:65–8:4, 8:37–42. In this example, the aligned decoded script and visemes sequence created from merging the transcriptions includes the results of the recognized events associated with the spoken utterance "turn it on" (with the accompanying lip movement), therefore, constitutes the outcome of the "*merging* ..." step.

118. Indeed, with regards to module 430 in Figure 4, Maes identifies "the **first joint use of the visual information and audio informatio**n in module 430" and "an explanation of how the two types of information [i.e, the audio information included in the decoded text/script and video information included in the viseme sequence] are combined to provide improved recognition accuracy." Ex.1005, 20:5–7, 11:55–63. In view of these teachings, a POSITA would have understood that module 430 provides the "*merging* ..." step and the outcome of this step produces the aligned decoded script and visemes sequence (i.e., a merged transcription), e.g., associated with the spoken utterance "turn it on" with the accompanying lip movement.



Ex.1005, Figure 4 (annotated).

119. Maes provides detailed explanation of how the audio signal and video signal are synchronized in the "*merging* …" step. 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.



Ex.1005, Figure 2. Referencing Figure 2, Maes explains that in step 202, raw multi-modal input (e.g., comprising audio and video input data) is obtained from multi-

modal data sources associated with the system. This data is then abstracted into a form that represents one or more events, such as "a spoken utterance." *See generally* Ex.1005, 5:57–6:8 and 6:28–35. In step 206, "the abstracted data or event is then sent by the I/O manager … to one or more recognition engines … to have the event recognized … . That is, depending on the nature of the event, one or more recognition engines may be used to recognize the event. For example, … **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, 6:38–50.

120. In Maes, one of the ways in which the audio-visual speech recognition engines processes or otherwise, recognizes event data is by generating an outcome in which the transcribed audio information such as the decoded text/script is aligned or synchronized with the transcribed video information such as the visemes sequence (*i.e.,* the outcome of the "*merging …*" step). *See also* Ex.1005, 21:50–54 ("**[T]he script is aligned with the visemes**. A rapid (or other) alignment may be performed in a conventional manner in order to attempt to **synchronize the two information streams**."), 21:59–61 ("a likelihood on the alignment is computed to determine how well **the script aligns to the visual data**"), 7:22–32 (**two, more or even all of the input modes described herein may be synchronized"** via the techniques disclosed in U.S. App. No. 09/507,526).

121. Therefore, Maes teaches or suggests "*merging the speech-based textual message and the nonspeech-based textual message*."

(B) Merging generates a query

122. Maes teaches or suggests that the "*merging ...*" is "*to generate a query*."

123. For example, Maes teaches:

> [t]he I/O manager receives the raw multi-modal data [in the form of audio input data, video input data, as well as other types of input data] and abstracts the data into a form that represents one or more events, e.g., a spoken utterance, a visual gesture. … **[A] data abstraction operation [e.g., comprising the "*merging* …"] may involve generalizing details associated with all or portions of the input data … to yield ["*generate*"] a more generalized representation of the data ["*a query*"]** for use in further operations.

Ex.1005, 6:32–38; *see also* Ex.1005, 4:6–22. In Maes, therefore, the generalized representation of the input data (including the spoken utterance and accompanying lip movement) is a computerized representation of the user's query that is generated as the tangible outcome of an abstraction operation (e.g., comprising the "*merging ...*" step). *Id*. When an abstract event occurs, Maes's system determines the target of the event, such as by performing the "*merging* …" step and then "**launches the action associated to the user's query**." Ex.1005, 36:59–37:3, *see also* 7:60–67.

124. Indeed Maes's system "*generate[s] a query*" because it:

> provides the capability to: (i) determine an object, application or appliance addressed by the user; … (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 ….

Ex.1005, 2:54–67. Therefore, Maes discloses that the "*merging* ..." step is "*to generate a query*."

125. Therefore, Maes teaches or suggests this element.

v. [13.4] searching the query for text combinations;

126. For example, Maes teaches:

> [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 [e.g., "*the query*"] for use in further operations [e.g., *searching*]**.

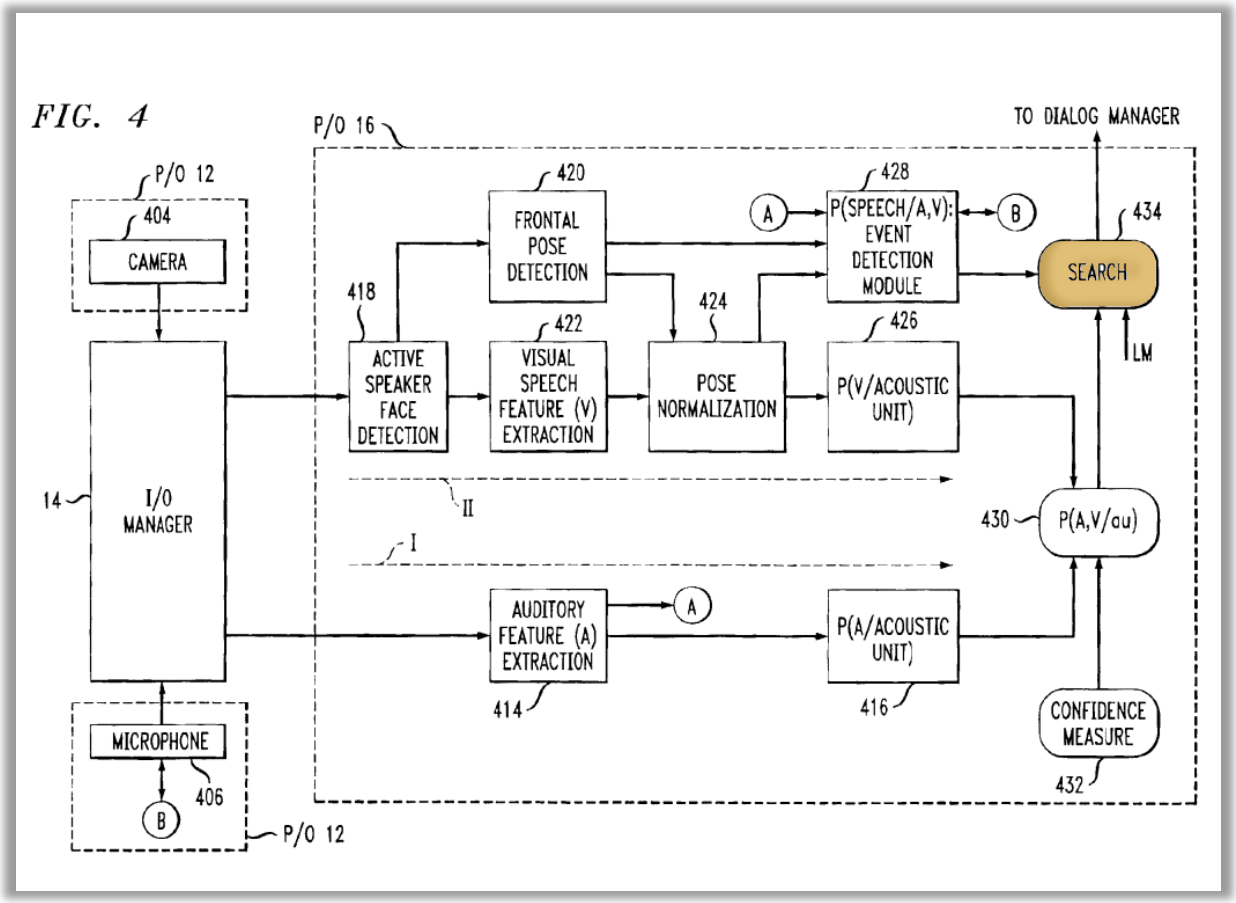
Ex.1005, 6:35–38.

127. Maes provides additional details of the "*searching* ..." step. Referencing Figure 4, Maes discloses "**a search is performed in search module**[12]

[12] Maes uses the terms "search module 434" and "search engine 434" interchangeably. Ex.1005, 20:11 and 20:16–17.

434 [“*searching*”] with language models (LM) … [and] the acoustic units … **representing what was uttered … are put together to form words**[, which] are output by the search engine 434 as the decoded system output. A conventional search engine may be employed. This output is provided to the dialog manager 18 of FIG. 1 for use in disambiguating the user’s intent.” Ex.1005, 20:11–20.

128. In Maes, therefore, the search engine 434 operates on the generalized representation of the data (“*the query*”), which represents what was uttered “*for*” providing as output, words put together from acoustic units (“*text combinations*”). A POSITA would have understood that a conventional search engine (such as search engine 434) operates on queries, or otherwise generalized representations of data, to provide outputs. *Id*.



Ex.1005, Figure 4 (annotated).

129. Therefore, Maes teaches or suggests "*searching the query for text combinations*."

vi. [13.5] comparing the text combinations to entries in a context description grammar;

130. Maes in combination with Ross renders obvious "*comparing the text combinations to entries in a context description grammar*."

131. Maes generally describes use of "grammar" and "grammar database" in its system.

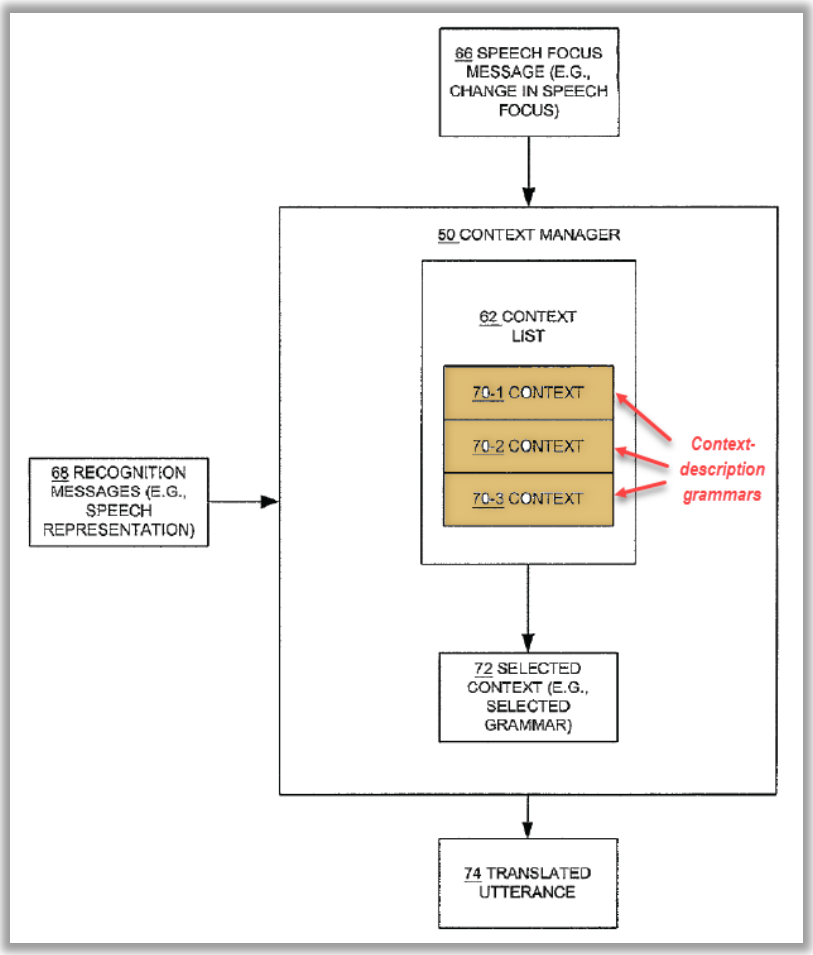
132. Maes explains that "**data needed by any recognition engine (e.g., grammar**, …)" is present on the conversational virtual machine (CVM), which Maes describes is a "component for implementing conversational computing … with respect to the present invention." Ex.1005, 31:15–18 and 34:61–67. The CVM handles input/output issues with conversational subsystems which "**us[e] the appropriate data files … (e.g., contexts, finite state grammars, vocabularies** …)." Ex.1005, 33:11-21.

133. Referencing Figure 9A, 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," and that dialog management unit 902 can "include … **finite state grammar (FSG)** … for machine-prompting the user." Ex.1005, 41:13–20, 39:22–29. Therefore, Maes discloses use of "grammar."

134. Maes describes using a "grammar database" stating: "Apparatus 900 … includes a processing module 910" which can further include "a speech recognizor … which can … include … **grammar database 930**." Ex.1005, 39:59 and 41:35–38.

135. Maes does not expressly disclose "*comparing the text combinations to entries in a context description grammar*," but, Ross does, as explained below.

136. First, Ross teaches "*entries in [a] context description grammar.*" For example, 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]. 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).

137. Ross teaches or suggests examples demonstrating "*entries*" included in *a* "*context description grammar.*" In Ross, "a grammar is defined for each application 26" and 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 "*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 "*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 "*entries.*"

138. 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 "match[ed]" ("*compar[ed]*") against entries in the grammar for the electronic mail application ("*entries in a context description grammar*"). The grammar for the calendar application allows "match[ing]" ("*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].

139. Ross describes the "*comparing ...*" step:

> 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 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].

140. Ross explains that one goal of matching ("*comparing*") a recognized utterance (e.g., "*the text combinations*") against data included in context grammars ("*entries in a 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, Ross "**uses the grammar 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].

141. In view of Ross's afore-mentioned teachings, it would have been obvious to a POSITA to modify Maes's teachings of generalized representations of data (or, simply queries/user queries) generated as a result of merging the transcriptions (Ex.1005, 6:32–50) to additionally include Ross's teachings such that queries generated using Maes's system would be compared against data (e.g., keywords/phrases used in applications) included in context grammars, per Ross. *See* §IX.A.1.

142. Therefore, Maes and Ross render obvious this claim element.

vii. [13.6] accessing a plurality of domain agents that are associated with the context description grammar;

143. Maes in combination with Ross renders this element obvious.

144. Initially, the 039 Patent describes "*domain agents*" as used for organizing "generic and domain specific behavior and information …. The domain agents provide complete, convenient and re-distributable packages or modules for each application area." Ex.1001, 14:40–44. Thus, "*domain agents*" in the 039 Patent broadly refer to software modules that are specific to each application area.

145. Ross teaches multiple speech-enabled applications, such as word processing application, mail application, spreadsheet application, calendar

application, and stock monitoring application running on a computer system in a multi-context speech enabled environment, which serve as examples of "*domain agents.*" Ex.1022, [0004], [0007]–[0008], and [00010].

146. Ross describes an example of selection between two applications: an electronic mail application and a calendar application which are candidate applications "targeted to receive" the user's spoken utterance "print it."

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

Ex.1022, [0053]; *see also* [0045] and [0051] (confirming t146146he "print it" phrase in grammars for both applications).

147. Therefore, Ross "uses a grammar to identify which speech enabled application is to receive the representation of the spoken utterance" and specifically, if such an utterance "**is acceptable to (can be processed by) a particular speech-**

enabled application." Ex.1022, [0013]; *see also* [0052] ("If **both of these grammars were loaded** into the context manager 50, the speech center system 20 is listening for any of the phrases accepted by either grammar. … 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**"). Therefore, Ross discloses this claim element.

148. Therefore, Maes and Ross render this element obvious.

viii. [13.7] generating a relevance score based on results from comparing the text combinations to entries in the context description grammar;

149. Maes and Ross render this claim element obvious.

150. For example, Ross describes its system using an "access characteristic" ("*generating a relevance score*") "*based on*" "recency of relevant access to the context [by] determin[ing] the context 70 for a speech enabled application 26 as indicated by the speech focus message 66" ("*results from comparing the text combinations to entries in the context description grammar*"). Ex.1022, [0012] and [0036].

151. Ross explains how its recency of access characteristic (e.g., "*relevance score*") is used. For example, Ross states that the priority order of context grammars in its context list (such as context list 62) is maintained based on the recency of access characteristic (e.g., "*relevance score*") of context grammars in a context list.

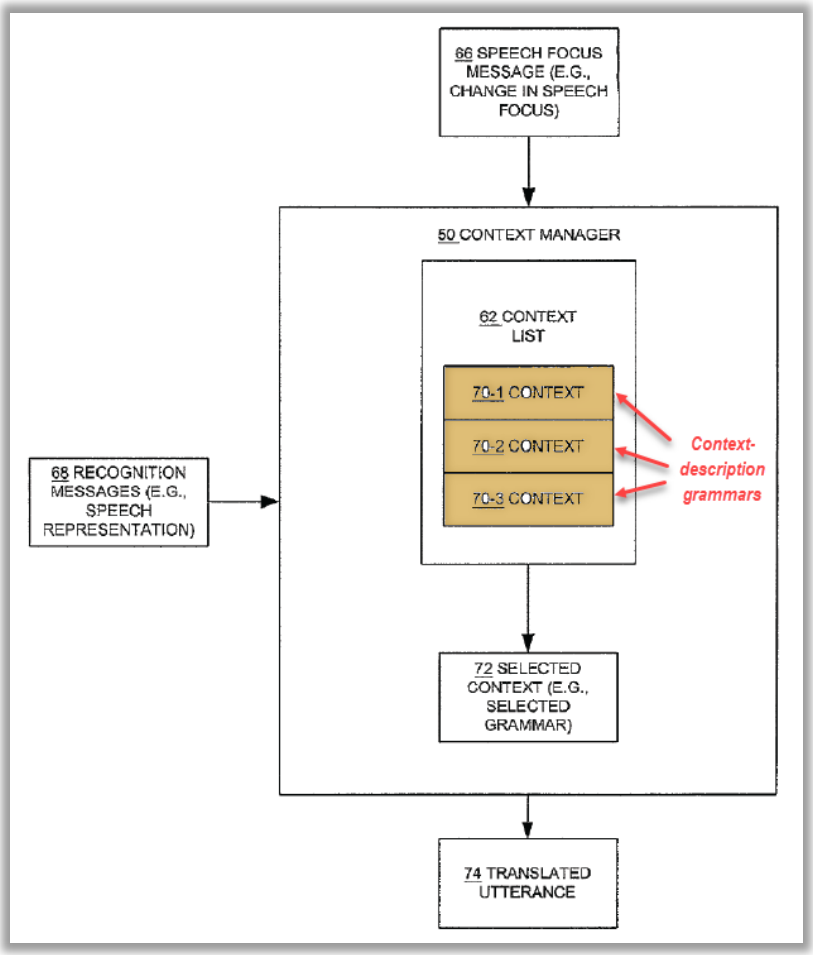
> The context manager 50 maintains the priority and state of the various grammars in the context list 62 in the system.

Ex.1022, [0034].

> 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.
>
> **The context list 62 of applications is reordered every time a recognition is directed to an application 26 other than the highest priority one, or whenever another application 26 gets windows focus (for example, because it was clicked on) …. The result is that the system 20 is biased to send the commands to the most recently accessed application 26**.

Ex.1022, [0053]-[0054]. Thus, the application that gets window focus is considered to be highly relevant in Ross and accordingly given highest priority.

152. In Figure 4, (reproduced below) Ross illustrates the priority order of the context grammars in the context list:



Ex.1022, Figure 4 (annotated). For example, within context list 62, in priority order, 70-1 (appearing at the top of the context list) is the most-recently accessed grammar having the highest recency of access characteristic (highest "*relevance score*"); 70-2 (appearing in the middle of the context list) is the next most-recently accessed grammar having a medium recency of access characteristic (medium "*relevance score*"); and 70-3 (appearing at the bottom of the context list) is the least accessed grammar having the lowest recency of access characteristic (lowest "*relevance score*").

153. Ross further explains its use of recency of access characteristic (e.g., "*a relevance score*") of a context grammar is "*based on results from*," for instance, on "the last application 26 that the user touched or talked to," which is determined upon matching ("*comparing*") *processed spoken utterances* ("*the text combinations*") "*to*" data (e.g., the phrases, keywords, and operators) in an application's grammar ("*entries in the context description grammar*"):

> [W]hen an utterance is recognized, it will be tested against each application's grammar to see if the grammar will accept it. The order of testing is based on a dynamic speech focus priority. The first application's grammar which will accept the utterance is then used for translation, and the command dispatched to the corresponding application 26. The speech focus priority is established by maintaining an ordered list of applications (e.g., context list 62). Whenever an application 26 gains window focus, it will move to the head of the list 62. Likewise, whenever an application 26 which is not the top priority application 26 is chosen as the target for a speech command, the application 26 indicated by the selected context 72 will move to the head of the list 62. In this way, **the last application 26 that the user touched or talked to** will get the first opportunity at interpreting the next user utterance, and the other applications 26 will be ordered in a most-recently-accessed way.

Ex.1022, [0035]. *See also* [13.5] describing the "*comparing ...*" step.

154. Therefore, Maes and Ross render this element obvious.

ix. [13.8] selecting one or more domain agents based on results from the relevance score;

155. Ross teaches or suggests this claim element.

156. Ross "maintains a context list of speech enabled applications" that the user has accessed and teaches "deciding the target application" ("*selecting one or more domain agents*") for a particular utterance provided by the user, e.g., "*based on*" "a recency of access characteristic" ("*the relevance score*"). Ex.1022, [0005] and Abstract; see also [13.7] (explaining the use of recency of access characteristic as "*relevance score*").

157. Ross provides additional details of the "*selecting ...*" using the recency of access characteristic ("*the relevance score*"):

> Whenever an application 26 gains window focus, it will move to the head of the list 62. Likewise, whenever an application 26 which is not the top priority application 26 is chosen as the target for a speech command, the application 26 indicated by the selected context 72 will move to the head of the list 62. In this way, **the last application 26 ["*domain agent*"] that the user touched or talked to will get the first opportunity at interpreting the next user utterance**, and the other applications 26 will be ordered in a most-recently-accessed way.

Ex.1022, [0035]. Identifying the last application that the user accessed (e.g., touched or talked to) would result in having that application having a higher recency of access characteristic, and therefore, Ross teaches or suggests "*selecting one ... domain agent[] based on*" recency of access ("*results from the relevance score*")..

158. Furthermore, Ross describes its system "biased to send the commands to the most recently accessed application 26" (e.g., "*based on results from the relevance score*") (Ex.1022, [0054]) and provides an example of "*selecting*" a calendar application ("*one or more domain agents*") "targeted to receive" the user's spoken utterance "print it" because, as Ross describes, "**the context … for the calendar application … supersedes the context … for the electronic mail application … in the context list 62**" "*based on*" the recency of access characteristic ("*results from the relevance score*"). Ex.1022, [0058].

159. In this example, "[t]he user … us[ing] the mouse to click on the fourth appointment displayed in a window for the calendar application," causes the calendar application to "gain[] window focus" or otherwise have a higher recency of access characteristic (e.g., higher "*relevance score*") to accept the spoken utterance "print it." This example further confirms Ross describing the "*selecting ...*" step. Ex.1022, [0058] and [0035].

160. Ross describes its system "tracking window focus" for applications (e.g., in "*selecting one or more domain agents*") using Microsoft Active

Accessibility (MSAA) from Microsoft Corporation, Redmond, Wash. Ex.1022, [0025] and [0035]. "**Changes in window focus, such as dialogs popping up and being dismissed, and applications 26 launching and exiting, must all be monitored in order to interpret the meaning of voice commands**. A preferred embodiment uses Microsoft(R) Active Accessibility(R) (MSAA) from Microsoft Corporation, Redmond, Wash." Ex.1022, [0025]. Therefore, Ross teaches or suggests this claim element.

161. Therefore, Maes and Ross render this element obvious.

x. [13.9] obtaining content that is gathered by the selected domain agents; and

162. At the outset, the 039 Patent suffers from antecedent basis issues. First: there is no mention of "***the*** *selected domain agents*." While there is recitation of "*selecting one or more domain agents*" in a previous claim element, there is simply no reference to "*selected domain agents*." For purposes of my analysis, I have interpreted "***the*** *selected domain agents*" in this claim element to mean "***the*** *selected one or more domain agents.*" Notwithstanding these antecedent basis issues, as explained in the analysis below, Ross teaches or suggests this claim element.

163. As explained in [13.8], Ross teaches or suggests the step of "*selecting one or more domain agents*" with an illustrative example in which the calendar application ("*the selected domain agent[]*") is "targeted to receive" user's spoken

utterance "print it." Ex.1022, [0058]. Because Ross teaches that "a grammar is defined for each application," consequently, an application "targeted to receive" a user's spoken utterance has "a grammar that will accept the phrase." Ex.1022, [0035] and [0053]. Further, Ross teaches applications "load[ing]" "grammars," e.g., represented by contexts. Ex.1022, [0028] (disclosing the system "allows applications … to load … grammars"), [0033] ("contexts 70 (e.g., 70-1, 70-1, 70-3) for speech enabled applications …, which represent the grammars for the applications …"), and [0052] ("grammars … loaded into the context manager"). Thus, in the calendar application example above, Ross's system receives ("*obtain[s]*") "the context … for the calendar application" ("*content*"), e.g., including the data/entries for the calendar application (see Ex.1022, [0046]) "load[ed]" ("*gathered by*") the calendar application ("*the selected domain agent[]*").

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

Ex.1022, [0046] (showing exemplary "*content*").

164. Therefore, Ross teaches or suggests this claim element.

xi. [13.10] generating a response from the content, wherein the content is arranged in a selected order based on results from the relevance score.

165. As explained in [13.9], in Ross, "the context … for the calendar application" constitutes an example of "*the content.*" Furthermore, by teaching that "the context … for the calendar application" "supersedes the context … for the electronic mail application in the context list," Ross describes that "*the content is arranged in a selected order.*" Ex.1022, [0058].

166. Furthermore, Ross discloses its system "determines whether the context 70 ["*the content*"]" for that application 26 "is **at the top** of the context list 62" ["*arranged in a selected order*"] and further that the system "has to make a choice by referring to the context list 62 of applications" "in order of recency of access" ("*based on results from the relevance score*"). Ex.1022, [0036] and [0053]. Thus, with respect to the calendar application example in [13.9], the outcome of determining whether the context for the calendar application is at the top "*generat[es] a response*" identifying that "the context … for the calendar application" ("*the content*") "supersedes the context … for the electronic mail application in the context list." Ex.1022, [0058].

167. For the additional reasons below, Ross teaches or suggests that "*the content is arranged in a selected order based on results from the relevance score*," as required in this claim element.

168. For example, Ross explains "grammars (indicated by the contexts 70 [e.g., including the grammar for the calendar application] in the context list 62) in **priority order**." Ex.1022, [0053]; *see also* [0034] ("active grammars in the context list 62 in priority order" and "maintain[ing] the priority and state of the various grammars in the context list 62 in the system.").

169. Ross explains the priority order of the various grammars (or, equivalently contexts represented by the grammars) is "*based on*" the most-recently accessed application, such that grammar of the last application that the user touched or talked to ("*results from the relevance score*") has highest priority:

> **The speech focus priority is established by maintaining an ordered list of applications (e.g., context list 62).** Whenever an application 26 gains window focus, it will move to the head of the list 62. Likewise, whenever an application 26 which is not the top priority application 26 is chosen as the target for a speech command, the application 26 indicated by the selected context 72 will move to the head of the list 62. In this way, **the last application 26 that the user touched or talked to** will get the first opportunity at interpreting the next user utterance, and the other applications 26 will be ordered in a most-recently-accessed way.

Ex.1022, [0035]. Therefore, the priority order (or, equivalently the speech focus priority) of Ross's context list (comprising context grammars) in which the grammar for the most-recently accessed application (e.g., the calendar application) is at the top of the context list constitutes "*the content ... arranged in a selected order.*"

170. Ross further explains that its system

> **prioritize[s] the contexts [included in the context list] based on the access characteristic** [which depends] on recency of relevant access to the context.

Ex.1022, [0012]; *see also* [0053] ("**context list 62 of applications [arranged] in order of recency of access**").

171. Ross further explains that "[t]he context list 62 of applications is reordered every time a recognition is directed to an application 26 other than the highest priority one, or whenever another application 26 gets windows focus (for example, because it was clicked on)". Ex.1022, [0054].

172. Therefore, Ross discloses this claim element.

b. Dependent Claim 14

i. [14.0] The method according to claim 13, further comprising generating an aggregate response that includes the content that is gathered by the selected domain agents.

173. Ross provides examples of a grammar for a calendar application and a grammar for an electronic mail application, each of which includes the "**print it**"

phrase. Ex.1022, [0045] and [0051]. Ross explains: "[i]f the sentence is 'print it,'" the system determines "**both grammars are capable of accepting the utterance**" and thus the system "tests the utterance against these grammars (indicated by the contexts 70 in the context list 62) in priority order." Ex.1022, [0053]; *see also* Ex.1022, [0059] (describing that the system "resolves the ambiguous phrase (e.g., "print this" or "print it") by looking in the context list 62 to determine the top priority context 70"). As the above-mentioned disclosures demonstrate, the outcome of identifying that both grammars are capable of accepting the "print it" utterance and testing the processed utterance against these grammars, therefore, serves as "*generating an aggregate response.*" And, furthermore, the "*aggregate response*" concerns a determination involving both grammars.

174. Furthermore, the "*aggregate response*" includes the "the context … for the calendar application" ("*the content*") "load[ed]" ("*gathered by*") the calendar application ("*the selected domain agent[]*") because, Ross teaches "[o]nly 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]. Indeed, Ross teaches that "the context … for the calendar application" ("*the content*") "supersedes the context … for the electronic mail application in the context list." Ex.1022, [0058]. *See also*

[13.9] and [13.10] explaining "*content … gathered by the selected domain agents*" and "*a response from the content*."

c. Dependent Claim 15

i. [15.0] The method according to claim 13, further comprising: receiving a follow-up speech and non-speech communications;

ii. [15.1] transcribing the follow-up speech and non-speech communications to create a follow-up speech-based textual message and a follow-up non-speech-based textual message; and

iii. [15.2] merging the follow-up speech-based textual message and the follow-up non-speech-based textual message to generate a follow-up query.

(A) [15.0]

175. Referencing Figure 2, Maes explains that when the system does not have enough original input to determine the user intent: "in step 214," the system "generat[es] … an output to the user **requesting further input data** ["*a follow-up speech and non-speech communications*"] so that the user's intent can be disambiguated," "the system … **obtains ["receiv[es]"] the raw input data**, again in step 202, and the process … iterates based on the new data. Such iteration can continue as long as necessary … to determine the user's intent." Ex.1005, 8:47–50, 8:58–65. The system may "seek confirmation in step 216 from the user in the same manner as the request for more information (step 214)." Ex.1005, 8:66–9:1.

176. Based on the iterative Figure 2 process, Maes's system, which "*receiv[es] the speech and non-speech communications*" (*see* [13.1]), therefore, applies the same receiving technique of the "further input data" ("*a follow-up speech and non-speech communications*") as the originally-received/raw input data.

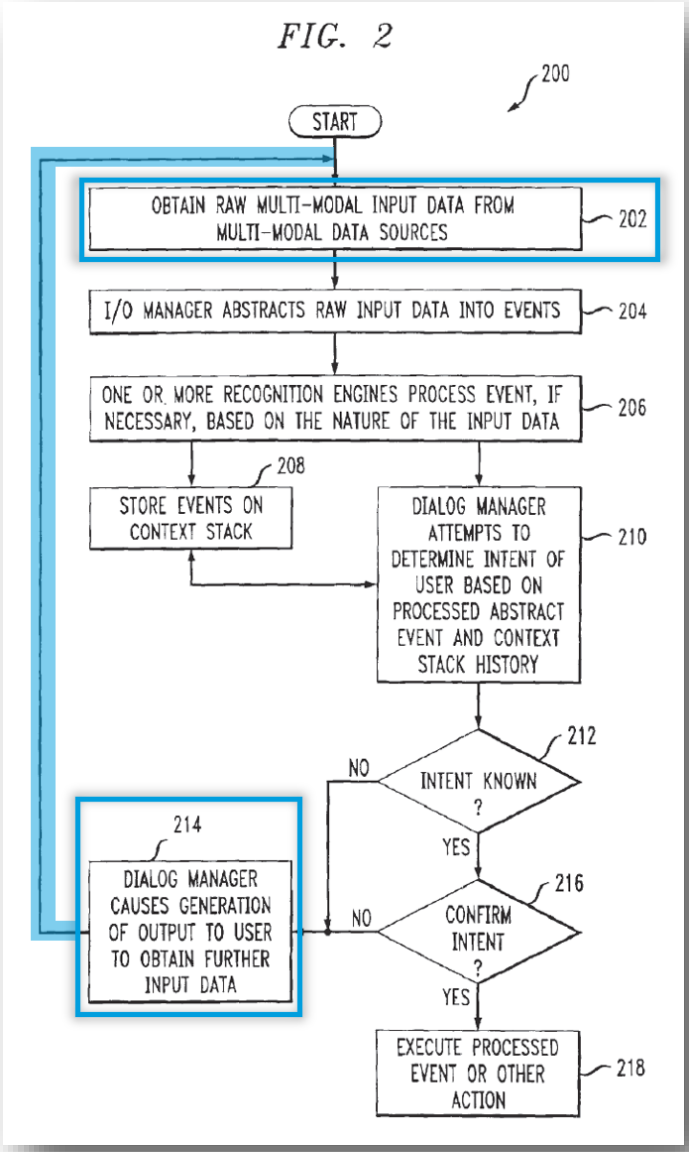
177. With reference to the in-vehicle example in which the system does not have enough original input to determine the user intent, referencing Figure 2, Maes discloses:

> Consider the case where the user says "turn it on," but makes no gesture and provides no other utterance. In this case, assume that the dialog manager does not have enough input to determine the user intent (step 212 in FIG. 2) and thus implement the command. The dialog manager, in step 214, then causes the generation of an output to the user **requesting further input data** so that the user's intent can be disambiguated. This may be accomplished by the dialog manager instructing the I/O manager to have the I/O subsystem output a request for clarification. … The dialog manager then generates a predetermined question such as "what device do you want to have turned on?" … The user, hearing the query, could then point to the radio or say **"the radio"** thereby providing the dialog manager with the additional input data to disambiguate his request. That is, with reference to FIG. 2, the system 10

> obtains the raw input data, again in step 202, and the process 200 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, 8:43–65.

178. In the above example, the user's feedback saying the spoken utterance "the radio" which includes the accompanying lip movement constitutes "*a follow-up speech and non-speech communications*." The iterative nature of the process in Figure 2 confirms that Maes's system processes the further or additional input data (*see* step 214) by applying the same technique and proceeds in the same manner as the originally-received input data.



Ex.1005, Figure 2 (annotated). Ex.1005, 8:43–65, 36:59–37:2 ("follow[ing] up with a dialog," "confirmation, disambiguation, correction, more details" are needed).

iv. [15.1]

179. As explained in [15.0], Maes teaches or suggests receiving "further input data" ("*a follow-up speech and non-speech communications*").

180. Based on the iterative Figure 2 process, Maes's system, which "*transcrib[es] the speech and non-speech communications to create a speech-based textual message and a non-speech-based textual message*" (*see* [13.2]), involves applying the same technique and proceeds in the same manner for the "further input data" ("*the follow-up speech and non-speech communications*") as the originally-received/raw input data until the user's intent is understood, iteratively. *See generally* Ex.1005, 8:66–9:6 ("The dialog manager … may … seek confirmation in step 216 from the user in the same manner as the request for more information (step 214)").

181. Accordingly, for the "further input data" ("*the follow-up speech and non-speech communications*"), a follow-up decoded text or script ("*a follow-up speech-based textual message*) and a follow-up visual phonemes (visemes) sequence ("*a follow-up non-speech-based textual message*") is created. Ex.1005, 6:43–50, *see also* [13.2].

v. [15.2]

182. Maes teaches or suggests element [15.2].

183. Based on the iterative Figure 2 process, Maes's system, which "*merg[es] the speech-based textual message and the nonspeech-based textual message to generate a query*" (*see* [13.3]), involves applying the same technique and proceeds in the same manner for the "further input data" ("*the follow-up speech and non-speech communications*") as the originally-received/raw input data until the

user's intent is understood, iteratively. *See generally* 8:58–9:6. Specifically, Maes teaches or suggests synchronizing/aligning the follow-up decoded text information and the follow-up visual phonemes information for the reasons explained in [13.3], for the "further input data" ("*the follow-up speech and non-speech communications*").

184. As also explained in [13.2], the outcome of the "*merging ...*" for the originally-received/raw input data is "*to generate a query*." A POSITA would have understood that the outcome of the "*merging ...*" for the "further input data" ("*the follow-up speech and non-speech communications*") is "*to generate a follow-up query*" because Maes teaches applying the same technique and proceeds in the same manner for the "further input data" ("*the follow-up speech and non-speech communications*") as the originally-received/raw input data until the user's intent is understood, iteratively. *See generally* Ex.1005, 8:66–9:6.

185. Maes provides an example of "*generat[ing] a follow-up query*." Referencing Maes's in-vehicle system example (Ex.1005, 8:43–65) and the iterative Figure 2 process, Maes teaches the system "generat[ing] ("*generat[ing]*") a predetermined question "what device do you want to have turned on?" ("*a follow-up query*"). Ex.1005, 8:54–58.

186. Indeed Maes teaches "*generat[ing] a follow-up query*" because Maes disclose its system "seeks confirmation, disambiguation, correction, more details,

etc., until the intent is unambiguous and fully determined" by "follow[ing] up with a dialog to disambiguate, complete, correct or confirm the understanding" Ex.1005, 36:61-63 and 36:67–37:2.

d. Dependent Claim 17

i. [17.0] The method according to claim 13, further comprising generating a context stack that includes one or more contexts that are selected based on the query.

(A) Generating a context stack including contexts

187. Maes describes "the context stack 817 may be **implemented** …" ("*generating a context stack*") as part of the context stack 20 of Figure 1. Ex.1005, 37:53–55. Referencing Figure 1 (below), Maes describes "the multi-modal conversational computing system 10 comprises … a context stack 20." Ex.1005, 3:66–4:6.

188. Maes further describes its "*context stack*" as storing "historical information" e.g., past input/output (I/O) events or "*one or more contexts*."

> [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.

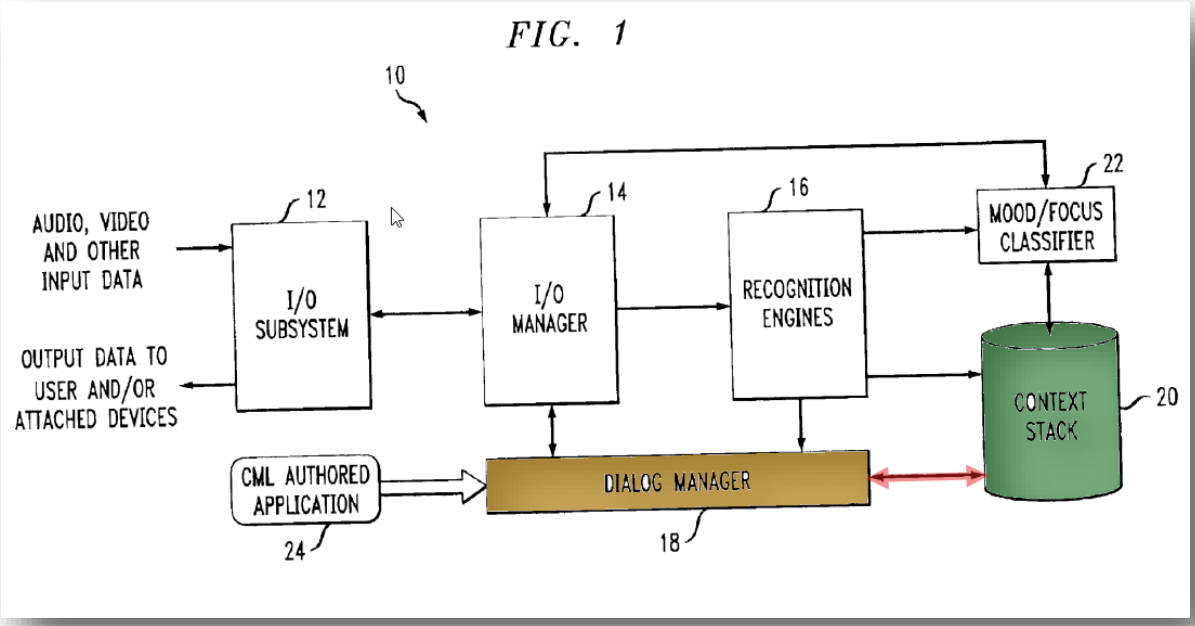


Figure 1 (annotated); *see also* Ex.1005, 7:40–45 (disclosing "the system … attempts to determine the **user intent based on the current event** and the **historical interaction information stored in the context stack**."); 5:16–19 (disclosing "**an I/O (input/output) event generated previously and stored in a context manager/history stack** (e.g., if a beeper rang and the user asked 'turn it off'").

(B) Contexts selected based on the query

189. Maes teaches the "*one or more contexts … [are] based on the query*" because, among other data, Maes's context stack comprises "queries to the backend":

> context stack 817 comprises all the information associated with an application … [which] 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.

190. Maes teaches that the "*one or more contexts … are selected*" or simply, "[*selecting*] *one or more contexts.*" Referencing the in-vehicle example in which a user may say "turn it on," 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 [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.

191. In this 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 ("*the one or more contexts*") are stored on the context stack because Maes discloses "**all** the variable, states, input, output and queries to the backend … 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." Ex.1005, 36:49–54.

192. Therefore, Maes teaches the dialog manager associating the results of the recognized events (e.g., produced as a result of merging the transcriptions) 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 an active dialog, which serves as describing the step of "[*selecting*] *one or more contexts*" from the context stack. And as explained above, Maes teaches "*generating a context stack that includes one or*

more contexts … based on the query." Therefore, taken together, Maes teaches or suggests "*generating a context stack that includes one or more contexts that are selected based on the query*."

193. Therefore, Maes teaches or suggests this claim.

e. Dependent Claim 18

i. [18.0] The method according to claim 17, wherein the one or more contexts are generated based on applying prior probabilities or fuzzy possibilities to (i) keyword matching, (ii) user profiles, (iii) a dialog history, or any combination of (i) to (iii).

194. Maes teaches or suggests this claim.

195. First, Maes discloses maintaining a "*dialog history*." For example, Maes discloses its context stack storing "historical interaction information stored in the context stack," which, for example, "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 [e.g., collectively "*a dialog history*"]. The context stack is associated with the organized/sorted context corresponding to each active dialog." Ex.1005, 7:41–42 and 37:55–61.

196. Further, Maes teaches or suggests that "*the one or more contexts are generated*" "*based on applying … fuzzy possibilities to … a dialog history*" as required in this claim.

197. Maes describes phonemes as examples of "*the one or more contexts*" stored on the context stack. According to Maes, phonemes are "sub-phonetic or acoustic units of speech" (Ex.1005, 12:29–33) or, equivalently "portions [of dialogs]" (Ex.1005, 7:60–8:4). *See also* Ex.1005, 8:37–42; *see also* 37:60–61 ("The context stack is associated with the organized/sorted context corresponding to each active dialog.")

198. Further, Maes teaches a phoneme having an associated probability indicating the likelihood ("*based on … [a] fuzzy possibilit[y]*") that it was that particular phoneme/acoustic unit that was spoken ("*[applied] … to dialog history*").

199. For example, 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, as is known in the art, are sub-phonetic or acoustic units of speech. … 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.

> [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|acoustic unit)**."

Ex.1005, 12:38–43.

> … 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|acoustic unit)**.

Ex.1005, 18:46–60. Accordingly, in Maes, a phoneme ("*the one or more contexts ... generated*") having an associated probability indicating the likelihood that it was that particular phoneme/acoustic unit that was spoken, therefore, serves as teaching or suggesting that "*the one or more contexts are generated*" "*based on applying ... fuzzy possibilities to ... a dialog history*"

200. Moreover, at the time of the claimed invention, using likelihoods (e.g., computed according to Maes's teachings described above) to calculate fuzzy probabilities was well-known. For example, Cattaneo published a paper entitled "Fuzzy Probabilities Based on the Likelihood Function" (Ex.1027) in which it proposed "a probabilistic-possibilistic hierarchical model based on the likelihood

function." Which, Cattaneo describes as offering "an ideal basis for inference and decision making." Ex.1027, 43.

201. Therefore, Maes teaches or suggests "*wherein the one or more contexts are generated based on applying … fuzzy possibilities to … a dialog history*."

X. OBJECTIVE INDICIA OF NON-OBVIOUSNESS

202. I am not aware of any objective indicia of non-obviousness that would change my opinion that the challenged claims of the 039 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

203. 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.

Henry H. Houh

Dated: March 21, 2025

CLAIM LISTING

Claim 13	
[13.0]	A method of processing speech and non-speech communications, comprising:
[13.1]	receiving the speech and non-speech communications;
[13.2]	transcribing the speech and non-speech communications to create a speech-based textual message and a non-speech-based textual message;
[13.3]	merging the speech-based textual message and the non-speech-based textual message to generate a query;
[13.4]	searching the query for text combinations;
[13.5]	comparing the text combinations to entries in a context description grammar;
[13.6]	accessing a plurality of domain agents that are associated with the context description grammar;
[13.7]	generating a relevance score based on results from comparing the text combinations to entries in the context description grammar;
[13.8]	selecting one or more domain agents based on results from the relevance score;
[13.9]	obtaining content that is gathered by the selected domain agents; and
[13.10]	generating a response from the content, wherein the content is arranged in a selected order based on results from the relevance score.
Claim 14	
[14.0]	The method according to claim 13, further comprising generating an aggregate response that includes the content that is gathered by the selected domain agents.
Claim 15	
[15.0]	The method according to claim 13, further comprising: receiving a follow-up speech and non-speech communications;
[15.1]	transcribing the follow-up speech and non-speech communications to create a follow-up speech-based textual message and a follow-up non-speech-based textual message; and
[15.2]	merging the follow-up speech-based textual message and the follow-up non-speech-based textual message to generate a follow-up query.
Claim 17	
[17.0]	The method according to claim 13, further comprising generating a context stack that includes one or more contexts that are selected based on the query.

Claim 18	
[18.0]	The method according to claim 17, wherein the one or more contexts are generated based on applying prior probabilities or fuzzy possibilities to (i) keyword matching, (ii) user profiles, (iii) a dialog history, or any combination of (i) to (iii).