Paper No. 1

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

DIALECT, LLC,
Patent Owner.

IPR2025-00655
U.S. Patent No. 7,640,160

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,640,160

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page(s)

Cases

Abbot Vascular, Inc. v. Flexstent,
IPR2019-00882, Paper 48, 28-29 (Oct. 2, 2020)..10

Advanced Display Sys., Inc. v. Kent State Univ.,
212 F.3d 1272 (Fed. Cir. 2000)..7

In re Apple Inc.,
979 F.3d 1332 (Fed. Cir. 2020)..71

Apple Inc. v. Fintiv, Inc.,
IPR2020-00019, Paper 15, 12 (May 13, 2020)...71, 73

Arbutus Biopharma Corp. v. ModernaTX, Inc.,
65 F.4th 656, 662-63 (Fed. Cir. Apr. 11, 2023)...7

Black v. CE Soir Lingerie Co.,
No. 2:06-CV-544, 2008 WL 3852722 (E.D. Tex. Aug. 15, 2008)
aff'd, 319 F. App'x 901 (Fed. Cir. 2009)..10

Comcast Cable Communs., LLC v. Promptu Sys. Corp.,
Case No. IPR2018-00342, Paper 13 (P.T.A.B. July 19, 2018)..........................71

CommScope Techs. LLC v. Dali Wireless, Inc.,
IPR2022-01242, Paper No. 23 (PTAB Feb. 27, 2023).......................................73

Dialect, LLC v Bank of America, N.A.,
No. 2:24-cv-00207-JRG (E.D. Tex.)..1

Dish Network v. Broadband iTV,
IPR2020-01280, Paper 17 (PTAB Feb. 4, 2021)..72

Fintiv,
IPR2020-00019, Paper 11 (PTAB Mar. 20, 2020)...73

Google, LLC v. Parus Holdings Inc.,
IPR2020-00847, Paper 9 (PTAB Oct. 21, 2020)..71

Statutes

Other Authorities

EXHIBIT LIST

No.	Exhibit Description
1001	U.S. Patent No. 7,640,160
1002	File History of U.S. Patent No. 7,640,160
1003	Declaration of Dr. Henry Houh
1004	CV of Dr. Henry Houh
1005	U.S. Patent No. 6,964,023 ("Maes")
1006	RESERVED
1007	Second Amended Complaint
1008	RESERVED
1009	RESERVED
1010	RESERVED
1011	RESERVED
1012	International Patent Application Publication No. WO 00/20962 ("Coffman") filed as PCT/US99/22927
1013	U.S. Patent No. 7,137,126 ("Coffman126")
1014	RESERVED
1015	RESERVED
1016	RESERVED
1017	Excerpts from Dafydd Gibbon et al., "Handbook of Multimodal and Spoken Dialogue Systems: Resources, Terminology, and Product Evaluation" (2000) ("Gibbon")
1018	U.S. Patent No. 5,799,276 ("Komissarchik")
1019	Excerpts from McGraw-Hill Dictionary of Scientific Terms (2003)
1020	U.K. Patent Application GB2162347A
1021	D. Walters "Deterministic Context-Sensitive Languages: Part I*" ("Walters"), INFORMATION AND CONTROL 17, 14-40 (1970)
1022	U.S. Patent Application Publication No. 2002/0133354 ("Ross")
1023	M. Mao et al. "Automatic training set segmentation for multi-pass speech recognition," IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005
1024	Hetherington, I. L. (2005, September). A multi-pass, dynamic-vocabulary approach to real-time, large-vocabulary speech recognition. In *INTERSPEECH* (pp. 545-548).
1025	RESERVED
1026	Excerpts from Microsoft Computer Dictionary, 5th edition (2002)

No.	Exhibit Description
1027	RESERVED
1028	RESERVED
1029	EDTX Calendar, Judge Gilstrap

TABLE OF ABBREVIATIONS AND CONVENTIONS

Abbreviation	Meaning
160 Patent	Ex.1001: U.S. Patent No. 7,640,160
IPR	*inter partes* review
Petitioner	Microsoft Corporation ("Microsoft")
Patent Owner or PO	Dialect, LLC ("Dialect")
Second Amended Complaint	Compl.: Ex.1007
xx:yy–zz	column *xx*, lines *yy* to *zz*

I. INTRODUCTION

Petitioner submits this Petition for *Inter Partes* Review of claims 12 and 13 (the "Challenged Claims") of U.S. Patent No. 7,640,160 (the "160 Patent" (Ex.1001)), assigned to Patent Owner ("PO"). Petitioner respectfully submits that the Challenged Claims of the 160 Patent are unpatentable under 35 U.S.C. §103 in view of the prior art references discussed herein. Accordingly, it is respectfully requested that the Board institute an *inter partes* review of the 160 Patent.

II. MANDATORY NOTICES

A. Real Party-in-Interest

Apart from Petitioner Microsoft Corporation, a real party-in-interest for this Petition is Bank of America, N.A.

B. Related Matters

The 160 Patent is asserted by PO in *Dialect, LLC v Bank of America, N.A.,* Eastern District of Texas No. 2:24-cv-00207-JRG.

C. Lead and Back-Up Counsel and Service Information

Lead Counsel	Back-Up Counsel
Scott M. Border (Reg. #77,744) Winston & Strawn LLP 1901 L Street, N.W. Washington, D.C. 20036 Tel: (202) 282-5054 sborder@winston.com	**Carson Swope** *(Pro Hac Vice to be submitted)* Winston & Strawn LLP 255 Shoreline Dr., Ste. 520 Redwood City, CA 94065 Tel: (650) 858-6407 cswope@winston.com

Lead Counsel	Back-Up Counsel
	T.D. Goswami (Reg. #78,054) Winston & Strawn LLP 1901 L Street, N.W. Washington, D.C. 20036 Tel: (202) 282-5309 tdgoswami@winston.com

Petitioner consents to electronic service by email at Winston-Microsoft-Dialect-IPR@winston.com and the e-mail addresses listed above.

III. PAYMENT OF FEES

Petitioner authorizes the Office to charge the filing fee and any other necessary fee to Deposit Account No. 501814.

IV. REQUIREMENTS FOR *INTER PARTES* REVIEW

A. Grounds for Standing

Petitioner certifies that the 160 Patent is available for inter partes review. Petitioner is not barred or estopped from requesting an inter partes review challenging the claims on the identified grounds herein. Petitioner has not filed a civil action challenging the validity of a claim of the 160 Patent. This petition is being filed no more than 1 year after the date on which Petitioner was served with a complaint alleging infringement of the 160 Patent.

B. Identification of Challenged Claims

Ground 1: Maes (Ex.1005) and Ross (Ex.1022) render obvious claims 12 and 13 under 35 U.S.C. § 103.

Ground 2: Maes, Coffman (Ex.1012) and Ross render obvious claims 12 and 13 under 35 U.S.C. § 103.

V. THE 160 PATENT

A. Effective Filing Date

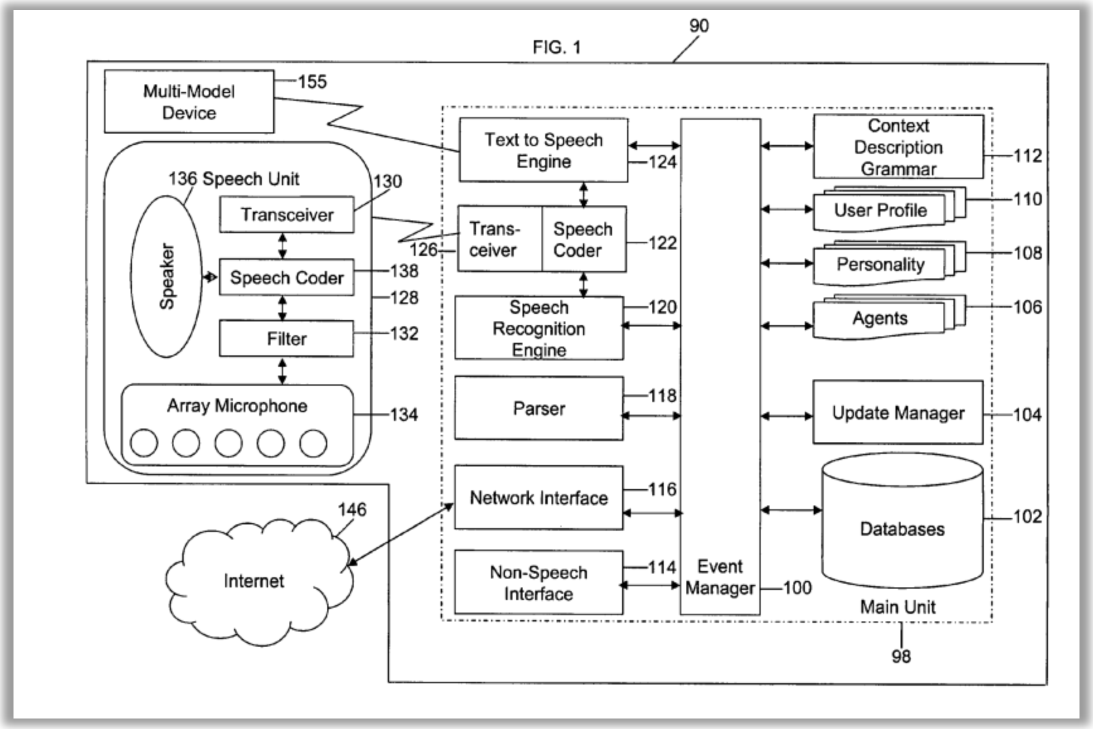
Petitioner assumes for the purposes of this Petition that August 5, 2005 is the effective filing date.

B. Person of Ordinary Skill in the Art (POSITA)

A POSITA with respect to the Asserted Patents as of the time of the invention, i.e., the earliest possible priority date of the 160 Patent, August 5, 2005, would have had a bachelor's degree in electrical engineering, computer science, computer engineering, or an equivalent, and two years of relevant experience involving computer science fundamentals, including natural language processing, speech recognition and transcription, non-speech recognition and transcription that is pertinent to the 160 Patent. Lack of professional experience can be remedied by additional education, and vice versa. Ex.1003, ¶24.

C. Overview of the 160 Patent

The 160 Patent "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 it "provides a fully integrated environment that allows users to submit natural language questions and commands via the speech interface and the non-speech interface. Information may be obtained from a wide range of disciplines, making local and network inquiries to obtain the information and presenting results in a natural manner, even in cases where the question asked or the responses received are incomplete, ambiguous or subjective." Ex.1001, 1:8–18.



Ex.1001, Figure 1.

Referencing Figure 1, the 160 Patent describes a system that includes speech unit, speech recognition engine, context description grammar module, parser, and agents. Speech unit includes a microphone to receive a spoken utterance from a user. Ex.1001, 11:55-57.

VI. CLAIM CONSTRUCTION

Claims are given their "ordinary and customary meaning" as understood by a POSITA and the prosecution history pertaining to the patent. 37 C.F.R. § 42.100(b). Because a POSITA would find the challenged claims unpatentable under any interpretation consistent with their plain and ordinary meaning in the context of the 160 Patent, the Board need not expressly construe the claim terms. *See Vivid Techs., Inc. v. Am. Sci. & Eng.'g. Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

VII. PRINCIPAL PRIOR ART

A. Summary of Maes

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. Therefore, Maes qualifies as prior under at least 35 U.S.C. § 102(e).

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 order to provide an effective conversational computing environment for … users." Ex.1005, Abstract.

Specifically, Maes explains that its system "receives multi-modal input in the form of audio input data, video input data, as well as other types of input data …, processes the multi-modal data … and performs various recognition tasks (e.g., speech recognition, speaker recognition, gesture recognition, lip reading, face recognition … in accordance with the recognition engines …) … 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 … ." Ex.1005, 4:7–22.

Moreover, Maes expressly incorporates by reference Coffman in its entirety. In particular, Maes states that "[i]t is to be appreciated the conversational virtual machine disclosed in PCT patent application identified as PCT/US99/22927 filed on Oct. 1, 1999 and entitled "Conversational Computing Via Conversational Virtual Machine," [*i.e.,* Coffman] the disclosure of which is incorporated by reference herein, may be employed to provide a framework for the I/0 manager, recognition engines, dialog manager and context stack of the invention." Ex.1005, 9:30–37. Coffman is, therefore, considered effectively part of Maes. *Arbutus Biopharma Corp. v. ModernaTX, Inc.*, 65 F.4th 656, 662-63 (Fed. Cir. Apr. 11, 2023) ("When a reference or material from various documents is incorporated, they are 'effectively part of the host document as if [they] were explicitly contained therein.'"); *see also Advanced Display Sys., Inc. v. Kent State Univ.*, 212 F.3d 1272, 1282 (Fed. Cir. 2000) ("Material not explicitly contained in the single, prior art document may still be considered for purposes of anticipation if that material is incorporated by reference into the document."); *see also Harari v. Lee*, 656 F.3d 1331, 1335 (Fed. Cir. 2011).

As the entire disclosure of Coffman is directed to "Conversational Computing Via Conversational Virtual Machine," Dr. Houh explains that a POSITA would understand Maes to be incorporating with detailed particularity the entire disclosure of Coffman. At a minimum, the disclosures related to the "conversational virtual machine" that "provide a framework for the I/0 manager, recognition engines, dialog manager and context stack," are expressly incorporated with particularly, which is the material relied upon below. Ex.1003, ¶51.

B. Summary of Ross

U.S. Pub. No. 2002/0133354 to Ross et al. (Ex.1022) was filed on August 16, 2001 and published on September 19, 2002. Therefore, Ross qualifies as prior under at least as 35 U.S.C. § 102(b).

As the title suggests, Ross entitled "System and Method for Determining Utterance Context in a Multi-Context Speech Application." describes techniques for determining a context associated with a user's spoken command or question to determine an application to invoke to process the command or question. Ex.1022, [0010] and [0013]. In Ross, determining a context involves evaluating the user's recognized spoken command against grammars for applications in which the grammars describe potential contexts (e.g., keywords/phrases) related to the utterance. Ex.1022, [0033]–[0034]. Ross teaches "test[ing]…against the active grammars" and finding a successful "match" involves matching text combinations in a user's transcribed utterance to grammar expressions in a grammar. Ross explains that "a grammar is defined for each application" 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 … hears a phrase such as 'print the first message' or 'print the first appointment,' the context manager

… can readily figure out the intended target application … for the uttered sentence. Only one grammar will accept the phrase, which thus indicates the selected context … for that phrase and that associated application … is the one that should be targeted to receive the corresponding command." Ex.1022, [0052]; see also [0053] ("The context manager … tests the utterance against these grammars (indicated by the contexts … in the context list …) in priority order, and passes the commands on to the first application … that has a grammar that will accept the phrase.")

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

To the extent one asserts that Coffman discloses a different embodiment from that of Maes, a POSITA would have been motivate to combine Coffman's teachings related to searching a context stack to find entries matching transcribed user input and merging/combining speech and non-speech transcriptions (Ex.1012, 41:2–5, 42:1–3, and 38:1–19) with Maes's teachings[1] that receives and processes multi-modal inputs (e.g., by producing decoded text or script) as described below. Ex.1003, ¶54.

[1] *See In re Stephan*, 868 F.3d 1342, 1346 n.1 (Fed. Cir. 2017) ("Whether a rejection is based on . . . combining multiple embodiments from a single reference . . . there must be a motivation to make the combination and a reasonable expectation that such a combination would be successful.")

As explained above, Maes expressly incorporates by reference Coffman in its entirety, names Stephane Maes as a common inventor, and has IBM as the original assignee. Ex.1005, Abstract, 9:30–37 and 19:43–49 and Ex.1012, Cover, Abstract. Accordingly, a POSITA would have been motivated to combine the teachings of Coffman with those of Maes for at least these reasons. *Black v. CE Soir Lingerie Co*., No. 2:06-CV-544, 2008 WL 3852722, at *14 (E.D. Tex. Aug. 15, 2008) aff'd, 319 F. App'x 901 (Fed. Cir. 2009). ("strong motivation to combine" prior art "patented by the same inventor" and that "reference each other."); *Abbot Vascular, Inc. v. Flexstent,* IPR2019-00882, Paper 48, 28-29 (Oct. 2, 2020) (finding that a POSITA would be motivated to combine references by the same author and from the same company). Ex.1003, ¶55.

Additionally, it would have been obvious to combine Ross, another IBM reference (Ex.1022, Cover), with Maes (and Coffman) because Ross and Coffman each provide motivation to combine their teachings with those of Maes. For example, Ross's teachings of maintaining a prioritized list of context grammars associated with applications/application programs, comparing a spoken utterance[2] to identify matches with context grammars, and selecting applications capable of

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

handling the user's spoken utterance combined with Maes would enhance its ability to determine target applications. Ex.1022, Abstract, [0002], [0013], [0021], [0033]–[0038], [0052]–[0053], [0059]–[0060], and Figure 4 and 6; Ex.1005, 36:54–66. Ex.1003, ¶56.

Combining Coffman's teachings of organizing/updating its context stack such that the most-recent context is at the top of the context (Ex.1012, 21, 23–28, 41:1–5, 40:15–19, and 42:7–9) with the teachings of Maes would benefit Maes's system by providing efficient identification and retrieval of the most-recent context, allowing for faster executions for matching/decoding of keywords and improving the responsiveness of Maes's system aligning with its goal of faster executions (Ex.1005, 11:64–12:30, 5:29–34), and would be inline with Ross's goal of providing quick or faster support in connection with determining "target applications" (Ex.1022, [0031] and [0035]-[0036]). A POSITA would have understood that by keeping the context stack updated, e.g., with the most-recent context associated with a user's dialog, would allow an application ("*domain agent*") to use the most-recent context, which is likely to be relevant to the user's utterance, in determining the user's intent. Ex.1005, 36:64–37:3; Ex.1003, ¶57.

Dr. Houh further explains that a POSITA would be motivated to combine Maes (and Coffman) and Ross because they teach strikingly similar techniques. For example, Ross's teachings of testing/comparing a recognized utterance against data included in a context list (Ex.1022, [0013], [0034]–[0037], [0053], and Figure 4) is similar to Maes's technique of associating recognized events against data in a context stack (Ex.1005, 7:60–8:4 and Figures 1 and 8), and Coffman's technique of searching a context stack to find entries matching transcribed user input (Ex.1012, 42:1-9). Notably, Maes's context stack disclosures are almost verbatim as Coffman's. Ex.1012, 21, 23–28 and Ex.1005, 37:51–62; *compare also* Ex.1005, Figure 8B and Ex.1012, Figure 2. Ex.1003, ¶58.

Dr. Houh further explains that a POSITA would have understood that Maes's and Coffman's "context stack," which is similar to Ross's "context list" involves the application of known techniques (e.g., identifying contexts by performing comparisons of spoken utterances against data stored in a context list/ stack) to improve a similar system (e.g., Maes's system) in the same way. Dr. Houh explains that Ross's description of priority ordering of its context list (Ex.1022, [0037]) is consistent with the well-known concept of a "stack." Ex.1003, ¶59 (citing Ex.1026, 305, Ex.1026, 215).

Maes, Coffman, and Ross are analogous art as the 160 Patent for at least three reasons. *First*, all three references are directed toward the same field of endeavor as the 160 Patent—computer-implemented systems interpreting user utterances or otherwise natural language processing systems. Ex.1001, Abstract, 1:51–61, 3:25–45; Ex.1005, Abstract, 4:7–17; Ex.1012, 1–4 and 29:1–4; and Ex.1022, Abstract, [0021]. *See also* Ex.1005, 7:60–8:4; Ex.1022, [0053]. Ex.1003, ¶60.

Second, Maes, Coffman, and Ross are reasonably pertinent to a problem described by the 160 Patent: an environment for **reliably processing** a user's language queries, which is a problem of conventional systems identified in the 160 Patent. Ex.1001, 1:35-44; Ex.1005, Abstract, 4:7–17 and 6:39–55; Ex.1012, 4:4–9; Ex.1022, [0033]-[0034] and Figure 5. *Third*, similar to the 160 Patent, Maes, Coffman, and Ross teach the use of **grammars** and **vocabularies** in recognizing speech inputs, which indicate compatibility of their teachings and therefore a POSITA would have had a reasonable expectation of success combining them. Ex.1001, 13:35–38; Ex.1005, 31:15–18, 34:61–67, 33:11–21, 39:59, 41:13–20, and 41:35–38; Ex.1012, 29:21–32, 25:6–9; Ex.1022, [0028], [0033]-[0035], and Figure 4. Ex.1003, ¶61.

A POSITA would be motivated to combine Maes (and Coffman), and Ross because to do so would have been the arrangement of old elements (a speech transcription and recognition system comprising modules for speech transcription and recognition, speech-enabled applications handling a user's spoken utterance input, context grammars comprising keywords/phrases pertaining to applications, context stack storing current and historical data related to user interactions) with each performing the same function it has been known to perform (e.g., recognizing spoken utterances, converting spoken utterances into computer-readable format based on decoding the utterances, matching the decoded text of the utterance against grammars describing potential contexts such as keywords/phrases included in the utterance, searching a context stack to find matches between data stored on the context stack and transcribed user input) and yielding no more than what one would expect from such an arrangement (an improved computer-implemented system interpreting user utterances), as Maes seeks to implement. Ex.1005, 11:62–63, 2:54–67; Ex.1003, ¶62.

Furthermore, a POSITA would have been motivated to combine Maes (and Coffman), and Ross because the references make clear that their systems require no specialized hardware or software. In fact, these references teach using conventional, commercially-available systems. Ex.1005, 2:38–53 and 45:50–60, 12:43–49 ("processing … may be accomplished via any **conventional acoustic information recognition system**") (emphasis added[3]); Ex.1022, [0002] and [0025] (identifying examples of commercially-available speech recognition products); Ex.1012, Abstract ("The conversational computing system may be built on top of a

[3] Emphasis added throughout unless otherwise specified.

conventional operating system and API's … and conventional device hardware.") Ex.1003, ¶63.

Dr. Houh explains that a POSITA would have had a reasonable expectation of success in combining the teachings of these references because to do so would be a simple combination of analogous teachings that would be well within the skill of a POSITA. For example, modifying Maes's system producing decoded text to include Coffman's teachings related to searching a context stack to find entries that match transcribed user input and merging/combining speech and non-speech transcriptions and to further include Ross's teachings of maintaining a priority list of context grammars for applications would enhance the functionality of Maes's system and further, as Dr. Houh explains, such modifications would have been simple implementations predictably resulting in a recognition system with improved accuracy Maes seeks to implement. In one implementation, a POSITA would been motivated to modify Maes's "grammar" / "grammar database" to additionally include Ross's teachings of context grammars. Ex.1005, 31:15–18, 34:61–67, and 33:11-21; Ex.1022, [0028] and [0033]-[0035]; Ex.1003, ¶64.

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. Ex.1003, ¶65.

Moreover, in yet another example implementation, a POSITA would have understood to incorporate Ross's teachings related to context grammars for applications as an additional stand-alone database/module within Maes's system. A POSITA would have further understood that any of the above implementations would have yielded a system in which the results of the recognized input/output (I/O) events produced as a result of processing the user's spoken utterance using Maes's system would be compared, per Ross, against data (e.g., keywords/phrases used in applications) included in context grammars (for instance, stored within Maes's modified grammar database, or within Maes's modified context stack). Ex.1002, [0034]-[0035] and [0053]-[0054]. A POSITA would recognize that grammars (such as Backus Naur Form (BNF)) describing context associated with applications was well-known and utilized in the art. Ex.1022, [0013] and [0060]; Ex.1026, 49). Ex.1003, ¶66-67.

VIII. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1: Maes and Ross Render Obvious claims 12 and 13

1. Claim 12

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

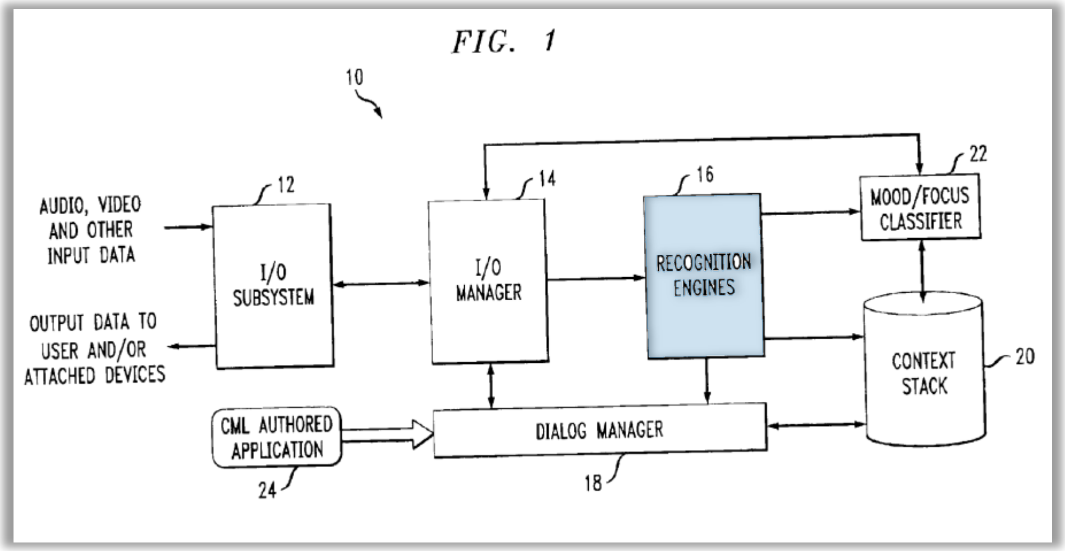
Initially, the 160 Patent describes "*knowledge-enhanced speech recognition*" as follows:

> [I]f a match is not found, or only a partial match is found, between the text message and active grammars, then a knowledge-enhanced speech recognition system may be used to semantically broaden the search. The knowledge-enhanced speech recognition system may be used to determine the intent of the request and/or to correct false recognitions. The knowledge-enhanced speech recognition may access a set of expected contexts that are stored in a context stack to determine a most likely context."

Ex.1001, 13:60–14:2. Therefore, as described in the 160 Patent, the functionality of a knowledge-enhanced speech recognition system is to determine the intent of the request and/or to correct false recognitions, which is analogously disclosed in Maes as ambiguity resolution or disambiguating the user's intent, as explained below. Ex.1003, ¶70.

Maes describes "[s]ystems and methods … provided for performing focus detection, **referential ambiguity resolution and mood classification in accordance with multi-modal input data [in the form of audio input data, video input data**, as well as other types of input data]." Ex.1005, Abstract, 4:10–11, 1:9-23. Ex.1003, ¶71.

Further, Maes is directed to "*interpreting natural language utterances using … speech recognition engine.*" Referencing Figure 1 (below), Maes describes its system including "**one or more recognition engines**" and that its system "receives multi-modal input in the form of audio input data, video input data, as well as other types of input data …, processes the multi-modal data …, and **performs various recognition tasks (e.g., speech recognition,** … in accordance with the recognition engines)." Ex.1005, 4:1–17; *see generally* 3:66–4:17:



Ex.1005, Figure 1 (annotated); Ex.1003, ¶73.

Referencing Figures 1 and 4 (below), Maes explains that its system comprises "*speech recognition engine*." Figure 4 is "an audio-visual speech recognition module[4] that may be employed as **one of the recognition modules[5] of FIG. 1 to perform speech recognition using multi-modal input data received** in accordance with the invention." Ex.1005, 10:50–52. 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) ("*the natural language utterance*") and a video signal (e.g., the speaker's face including lip

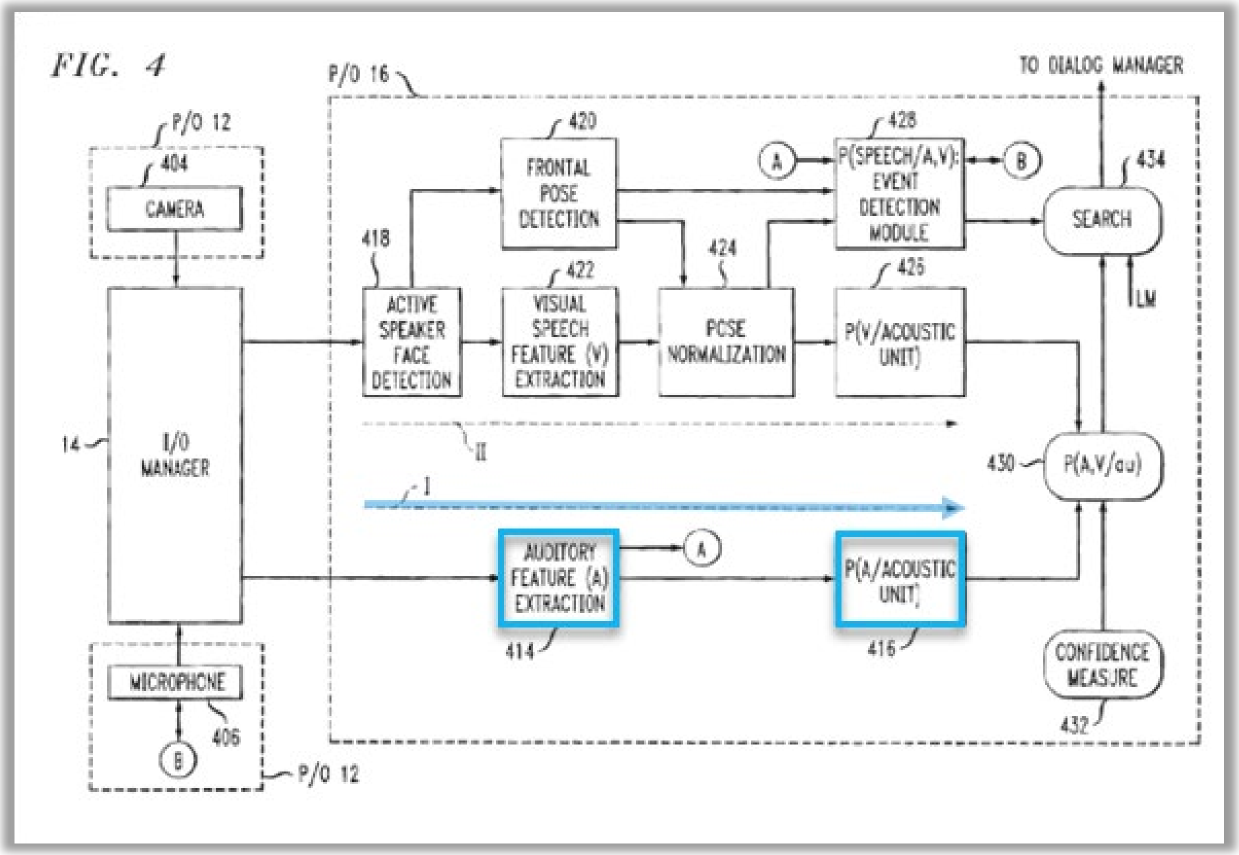
[4] As used herein, the terms "Maes's system" and "audio-visual speech recognition module" are used interchangeably.

[5] 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."

movement and/or background objects in the environment) and processes the multi-modal input data by passing through several processing modules (e.g., the video signal is processed by blocks/modules 418, 422, 424, 426 and the audio signal is processed by blocks/modules 414, 416). Ex.1003, ¶74.

Maes notes that "**the processing … in blocks 414 and 416 may be accomplished via any conventional acoustic information recognition system** ["*speech recognition engine*"] capable of extracting and labeling acoustic feature vectors, e.g., Lawrence Rabiner, Biing-Hwang Juang, "Fundamentals of Speech Recognition," Prentice Hall, 1993." Ex.1005, 12:43–49; Ex.1003, ¶74.

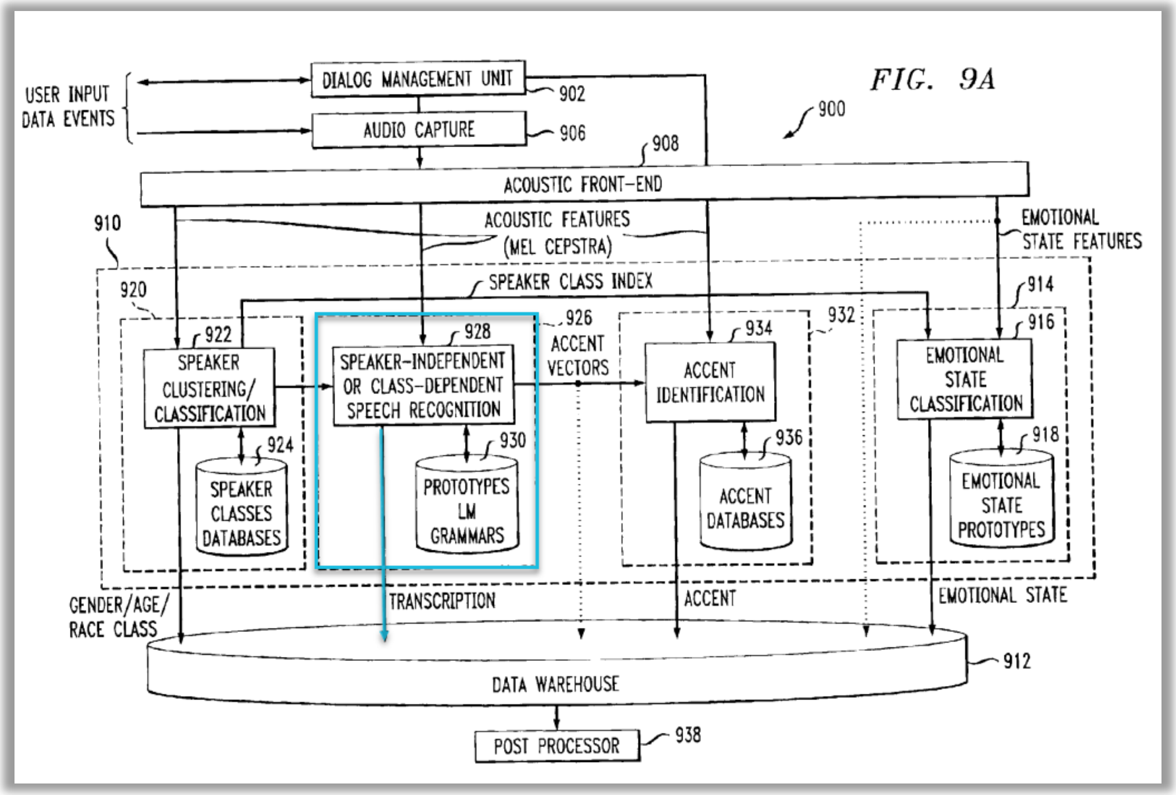
FIG. 4



Ex.1005, Figure 4 (showing audio information processing path (Path I) annotated in blue); Ex.1003, ¶74.

Referencing Figures 1 and 9A (below), Maes provides details of "*speech recognition*" functionality 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. Ex.1003, ¶75.

Maes continues: "Apparatus 900 … includes a processing module 910" which "can further include a speech recognizer 926 which … include[s] a speech recognition module 928," e.g., providing "*speech recognition*" *functionality* and "a speech prototype, language model and grammar database." Ex.1005, 39:59, 41:35–38; Ex.1003, ¶76.



Ex.1005, Figure 9A (annotated). "Apparatus 900 can … include **a post processor 938 … configured to transcribe user utterances and … perform keyword spotting thereon**" and which can employ speech recognizor 926. Ex.1005, 41:48–51 and 41:58–59. Therefore, Maes's system comprises functionality for "*speech recognition.*" Ex.1003, ¶76.

Maes further teaches or suggests "*a knowledge-enhanced speech recognition engine.*" Ex.1003, ¶77.

In addition to speech recognition module 928 which provides "*speech recognition*" functionality, Maes's system (e.g., embodied in apparatus 900) includes a semantic module. Specifically, Maes describes that apparatus 900 which receives "user-provided input data events" "via the I/O manager 14 of FIG. 1 includes a "**semantic module … to interpret meaning of phrases**" (*e.g.*, "*knowledge-enhanced*") such that it is "**used by speech recognizer 926 to indicate that some decoding candidates in a list are meaningless and should be discarded/replaced with meaningful candidates**." Ex.1005, 41:59–64. Thus, by indicating certain decoding candidates are meaningless and should be discarded/replaced with meaningful candidates, Maes's semantic module provides ambiguity resolution function or otherwise disambiguates the user's intent. Ex.1005, Abstract, 4:10–11, and 36:59–63. Dr. Houh notes that contemporaneous art such as the "Handbook of Multimodal and Spoken Dialogue Systems : Resources, Terminology, and Product Evaluation" (2000) by Daffyd Gibbon et al. ("Gibbon") describes "knowledge-based speech recognition system" as it applies to "knowledge-enhanced systems" as a system that "specifies explicit acoustic-phonetic rules that are robust enough to allow recognition of linguistically meaningful units and that ignore irrelevant variation in these units." Ex.1017, 429. For at least the reason above, the semantic module in Maes is "*knowledge-enhanced.*" And, as explained earlier, Maes's speech recognition module 928 provides "*speech recognition*" functionality and, therefore taken together, Maes's speech recognition module 928 and the semantic module, both comprised in Maes's system, jointly provide functionality that make Maes's system a "*knowledge-enhanced speech recognition engine*." Ex.1003, ¶77.

Indeed, Maes's system is "*knowledge-enhanced*" because Maes's system provides "**functions needed to adapt … to** the capabilities and constraints of the device, application and/or **user preferences**," uses "a-priori knowledge, … information about the user," and Maes teaches training methods based on "**speaker profile features (accent, dialect, gender, age, speaking rate** …" which are consistent with Patent Owner's description of "*knowledge-enhanced*" in its Complaint[6]. Ex.1005, 36:44–48, 37:22–28, and 19:7–19; Ex.1003, ¶78.

For the additional reasons below, Maes's semantic module is "*knowledge-enhanced*" which therefore makes Maes's system (which, as explained above, comprises speech recognition functionality) a "*knowledge-enhanced speech recognition engine*" system. Ex.1003, ¶79.

Maes explains that "[a]ll data stored in the data warehouse 912 [shown in Figure 9A as coupled to postprocessor 938 that includes the semantic module] can be stored in a format to facilitate subsequent data mining thereon" and as part of the data mining, "dialect, socioeconomic classification, … can be estimated based on

[6] Ex.1007, ¶83 ("*Erica* is also knowledge-enhanced in that it identifies the user and uses knowledge and information relating to the user in interpreting natural language utterances.")

vocabulary and word series used by the user. **Appropriate key words, sentences, or grammatical mistakes to detect can be compiled via expert linguistic knowledge**." Ex.1005, 43:16–20, 4:1–17, 43:38–41; *see also id.* 39:63–67, 41:48–51 ("a post processor 938 [that includes a semantic model] … coupled to the data warehouse 912 and … configured to … perform keyword spotting thereon"), 42:60–61 ("**a keyword spotter to detect insults**."). As Dr. Houh explains, using "expert linguistic knowledge" and detecting/spotting appropriate key words, sentences, insults, and/or grammatical mistakes fall within the purview of a semantic model/module. Ex.1003, ¶79 (citing Ex.1017, 468 (defining "*semantic[s]*") and Ex.1019, 1157 defining ("knowledge-based")). Therefore, Maes's semantic module is "*knowledge-enhanced.*" Ex.1003, ¶79.

In addition, Maes teaches "modify[ing] behavior of [its] system … based on …[a] user attribute … determined …[from an] acoustic feature extracted from the user's speech" and further that the "[f]eatures can be extracted and **decisions dynamically returned by the models**"—which indicate that the behavior of Maes's system is modified on-the-fly based on the knowledge (e.g., decisions) incorporated by its models (such as the semantic model), further confirming that Maes's semantic model is "*knowledge-enhanced.*" Ex.1005, 43:53-54, 44:7–11; Ex.1003, ¶81.

Moreover, as Dr. Houh explains, at the time of the 160 Patent, "*knowledge-enhanced speech recognition engine*" in the context of speech recognition systems was a well-known concept. For example, Komissarchik, reflecting the POSITA's knowledge as it applies to "*knowledge-enhanced*" systems, defined "knowledge-based systems" as "involv[ing] the application of acoustic, phonetic and natural language processing ('NLP') information theory to the speech recognition process" and proposed "knowledge-based speech recognition system and methods … which provide real-time analysis and transcription of spoken utterances." Ex.1003, ¶82 (citing Ex.1018, 2:39-43, 8:4–14, 6:45–47); *see also* Ex.1018, 2:60–3:62.

Therefore, in view of the above-described teachings, Maes teaches or suggests that its "*method …*" is "*using knowledge-enhanced speech recognition engine*. Ex.1003, ¶83.

Additionally, Maes teaches or suggests that its method is "*configured to determine an intent and correct false recognitions of the natural language utterances*." Ex.1003, ¶84.

Maes discloses: "When an abstract [input/output (I/O)] event occurs, … **[the system] … seeks confirmation, disambiguation, correction, more details, … until the intent is unambiguous and fully determined**," which serve as "*determin[ing] an intent and correct[ing] false recognitions of the natural language* *utterances.*" Ex.1005, 36:59–63. For example, and as Dr. Houh explains, by seeking disambiguation and correction, Maes's system "*correct[s] false recognitions.*" Further, Maes describes its system includes functionality to "prompt or display for missing, ambiguous or confusing information, asks for confirmation or launches the execution of an action associated to a fully understood multi-modal request from the user) of that application based on the multi-modal input." Ex.1005, 8:12–16. Therefore, Maes's system is capable of handling multi-modal I/O events to "**(a) understand the intent of the user; (b) follow up with a dialog to disambiguate, complete, correct or confirm the understanding [of the intent of the user]; (c) or, dispatch a task resulting from fall [sic:full] understanding of the intent of the user**." Ex.1005, 36:64–37:3; Ex.1003, ¶85.

The above disclosures in Maes are analogous to the 160 Patent's description of "knowledge-enhanced speech recognition system … to determine the intent of the request and/or to correct false recognitions." Ex.1001, 13:64–66. Ex.1003, ¶86.

Maes describes examples of spoken utterance inputs which further confirm that its method applies to "*natural language utterances.*" For example, Maes's system, which, among others, provides functions of "natural language understanding (NLU), natural language generation (NLG)" and "NL parsing" can be used 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, 41:14–16, 35:58, 10:63–11:15) and may be employed within a vehicle such that if a user says the spoken utterance "turn it on," Maes's system processes inputs relating to an I/O event representative of the user's spoken utterance. Ex.1005, 4:30–32, 6:43–50, and 7:63–8:29; Ex.1003, ¶¶87–92.

Therefore, if limiting, Maes teaches or suggests the preamble. Ex.1003, ¶93.

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

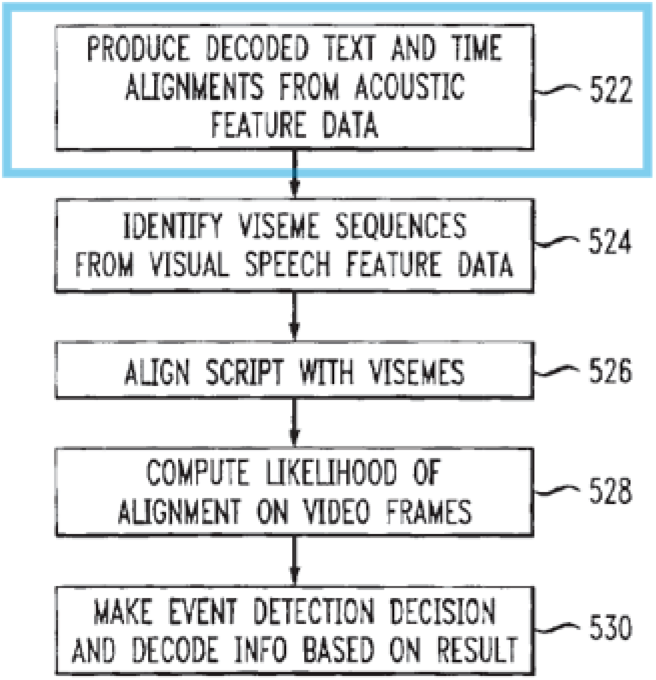
Because "*receiving a transcription of a natural language utterance*" requires "*a transcription of a natural language utterance*" before the transcription is received, the step of creating "*a transcription of a natural language utterance*" is first described below. Ex.1003, ¶95.

As explained in [12.0], Maes's system corresponds to a "*knowledge-enhanced speech recognition engine*." Ex.1003, ¶96.

And further, Figure 5D (below), Maes discloses classical speech recognition techniques producing decoded text (or, script) ("*a transcription of a natural language utterance*") using the feature data from the acoustic feature extractor 414 included in Maes's audio-visual recognition module ("*at a computer comprising the knowledge-enhanced speech recognition engine*") based on processing the spoken utterance. *See generally* Ex.1005, 21:43–47, Figure 4, and 10:47–48 (identifying

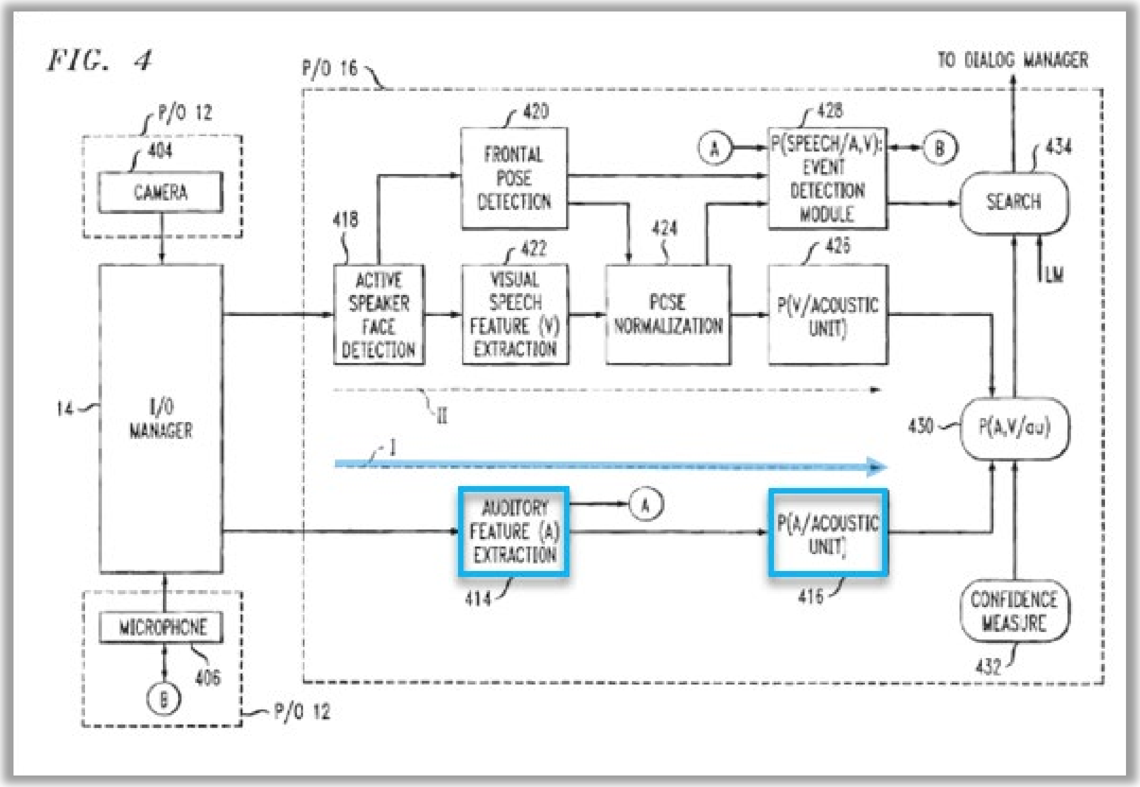
Figure 4 as an embodiment of an audio-visual speech recognition module that may be employed as one of the recognition modules of Figure 1 to perform speech recognition). Ex.1003, ¶97.

FIG. 5D


PRODUCE DECODED TEXT AND TIME ALIGNMENTS FROM ACOUSTIC FEATURE DATA
522
IDENTIFY VISEME SEQUENCES FROM VISUAL SPEECH FEATURE DATA
524
ALIGN SCRIPT WITH VISEMES
526
COMPUTE LIKELIHOOD OF ALIGNMENT ON VIDEO FRAMES
528
MAKE EVENT DETECTION DECISION AND DECODE INFO BASED ON RESULT
530

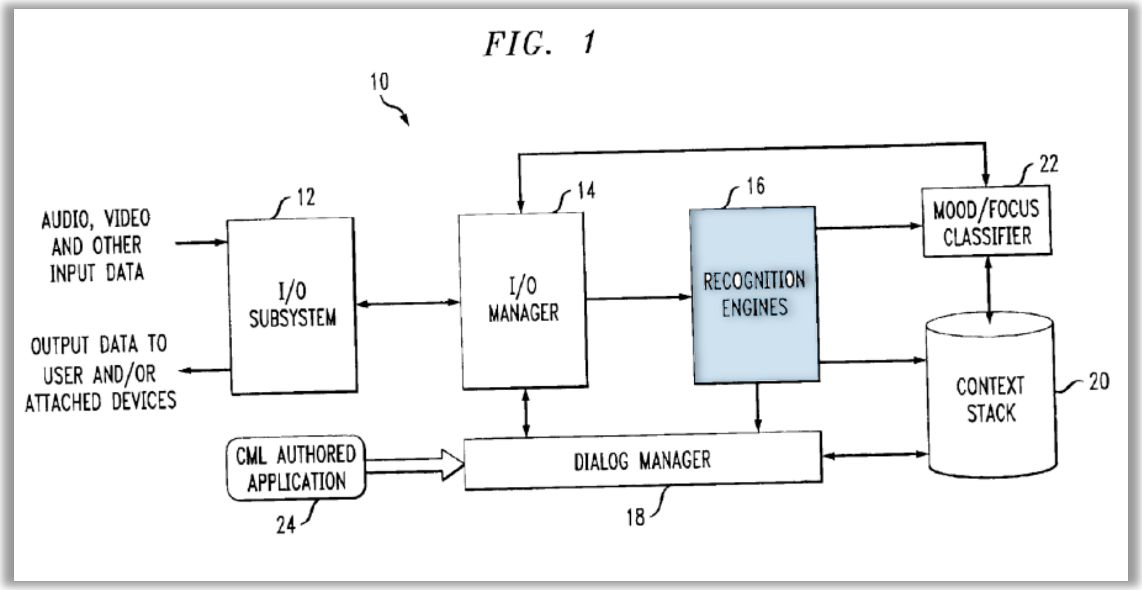
Ex.1005, Figure 5D (block 522). "[I]n step 522, the uttered speech to be verified **may be decoded by classical speech recognition techniques so that a decoded script** and associated time alignments **are available**. This is accomplished using the feature data from the acoustic feature extractor 414." Ex.1005, 21:43–47. A POSITA would have understood that the "decoded script" in Maes is essentially the tangible result of a speech recognition system's interpretation of spoken language and provides a text representation of the multi-modal content. Ex.1005, 20:61-64 ("If the buffered data is tagged as speech, in step 518, the buffered data is sent on through the acoustic path so that the buffered data may be recognized, in step 520, … to yield a **decoded output**."); *see also* 20:23-33.

Referencing Figures 1 and 4 (below), Maes provides details explaining the "*transcription ...*": Figure 4 shows that the audio-visual speech recognition module receives multi-modal input data including an audio signal (e.g., speech provided by the speaker and/or background noise) and a video signal (e.g., the speaker's face including lip movement and/or background objects in the environment). For example, the auditory feature extractor 414 "receives an audio or speech signal and … extracts spectral features from the signal at regular intervals. The spectral features are in the form of acoustic feature vectors (signals) … ." Ex.1005, 11:64–12:2. 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 of the audio information signal. Ex.1005, 11:55–64.



Ex.1005, Figure 4 (showing acoustic feature vectors denoted by the letter "A" and audio information processing path (Path I) annotated in blue); *see also* 23:1–5; Ex.1003, ¶.

FIG. 1



Ex.1005, Figure 1 (annotated).

Maes explains how acoustic features are extracted:

> 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; Ex.1003, ¶99.

Maes describes sampling the signal prior to extracting the acoustic feature vectors. "Before acoustic vectors are extracted, the speech signal may be sampled at a rate of 16 kilohertz (kHz)." Ex.1003, ¶100 (citing Ex.1005, 12:2–12). Indeed, Maes notes 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.1003, ¶100.

As Dr. Houh explains, Maes provides several examples of acoustic feature vectors (or alternatively cepstral vectors) that may be extracted: e.g., [linear prediction coefficients] LPC cepstra, [Perceptual Linear Prediction] PLP, MEL cepstra, and that the invention is not limited to any particular type. Ex.1003, ¶101 (citing Ex.1005, 23:40–43, 39:55–58).

Furthermore, Maes's system corresponds to "*a computer comprising the knowledge-enhanced speech recognition engine*." *See* [12.0]. Maes's "**system … comprises at least one processor, operatively coupled to the user interface subsystem, … configured to receive at least a portion of the multi-modal input data from the user interface subsystem**. … [and] further configured to then make a determination of at least one of an intent, a focus and a mood of … users based on at least a portion of the received multi-modal input data. The processor is still further configured to … cause execution of one or more actions to occur in the environment based on at least one of the determined intent, the determined focus and the determined mood. The system further comprises a memory, operatively coupled to the at least one processor, which stores at least a portion of results associated with the intent, focus and mood determinations made by the processor for possible use in a subsequent determination or action." Ex.1005, 2:38–54; Ex.1003, ¶102.

Maes explains that "the elements … in FIGS. 1 through 9C may be implemented in … 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 …." Ex.1005, 45:50–60. Maes, therefore, makes clear that the "*transcription* ..." results in computer-recognizable data, and consequently, it can be "*receiv[ed]*" by a computer or sent to another computer. Ex.1005, 11:55–64; Ex.1003, ¶103.

Referring to the in-vehicle example (Ex.1005, 4:30–32, 6:43–50, 10:63–11:15, and 7:63–8:29), Maes explains the "*transcription* ..." process: Maes's system receives the raw multi-modal data relating to an I/O event (e.g., representative of the user's spoken utterance and lip movement accompanying the spoken utterance) and “**abstracts the data into a form that represents … a spoken utterance** ... **[by] generalizing details associated with all or portions of the input data … to yield a more generalized representation of the data for use in further operations**.” Ex.1005, 6:29–38; Ex.1003, ¶105.

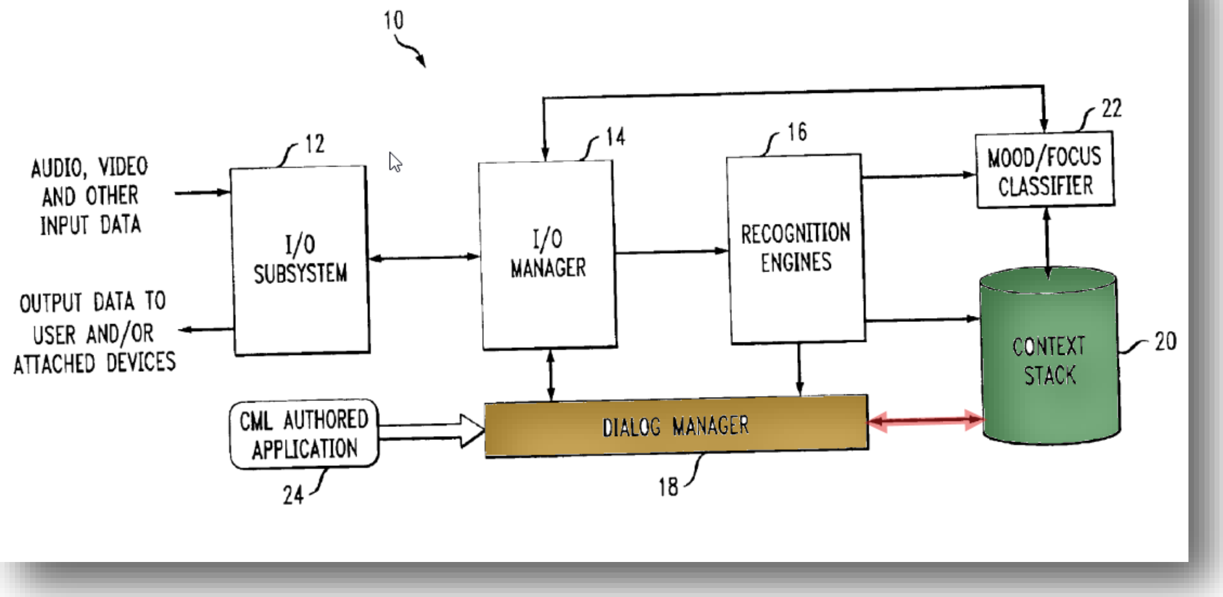
c. [12.2.0] identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription,

As explained in [12.1], in Maes, processing the spoken utterance generates decoded text (or, script) (“*a transcription of a natural language utterance*”). Ex.1003, ¶107.

Maes teaches storing “*one or more contexts*” stored in a context stack (such as context stack 817 in Figure 8 or context stack 20 in 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” (*see generally* Ex.1005, 3:66–4:6) storing “historical information (e.g., past events),” such as past input/output (I/O) events (“*one or more contexts*”) generated previously in the context of a dialog. Ex.1005, 7:62–63, 37:60–61; Figure 1 (below); *see also* Ex.1005, 7:40–45 (“the system … determine[s] the user intent based on the current event and the **historical interaction information stored in the context stack**”); 5:16–19 (“**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'").

FIG. 1



Ex.1005, Fig. 1 (annotated); Ex.1003, ¶¶108–109.

As explained below, Maes teaches "*identifying one or more contexts*." Referring to 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 and gesture could be associated**.

Ex.1005, 7:65–8:4.

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

Ex.1005, 8:37–42. In the above example, "this event" corresponds to the current event of the user pointing to the radio, which occurred after "the previous spoken utterance event" of the user saying "turn it on. Data for both events ["*one or more contexts*"] are stored on the context stack because Maes discloses "**all the variable, states, input, output and queries to the backend that are performed in the context of the dialog and any extraneous event that occurs during the dialog**" are stored on the context stack. Ex.1005, 37:55–62. Maes clarifies that "the dialog" includes "**conversational dialog comprising speech** and other multi-modal I/O such as GUI keyboard, pointer, mouse, as well as video input, etc." Ex.1005, 36:49–54. Maes teaches associating the results of the recognized events ("*the one or more text combinations*[7]) included ("*contained*") "*in*" the decoded text/script ("*the transcription*") with an organized/sorted context in a context stack such as context

[7] Patent Owner's Complaint appears to describe "*text combinations*" as "user-requested-associated keywords." Ex.1007, ¶88.

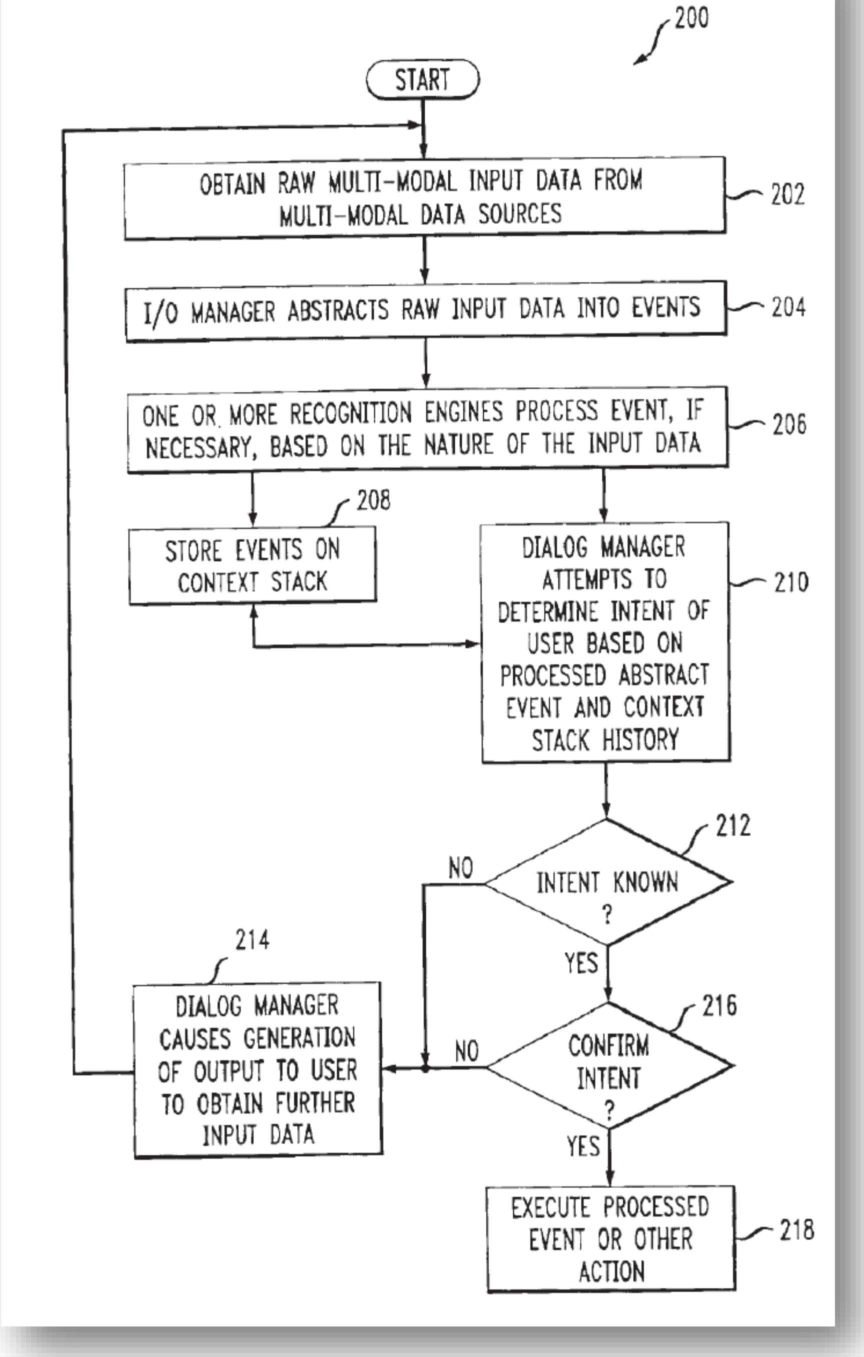
stack 817 in Figure 8 or context stack 20 in Figure 1 corresponding to in an active dialog, which, therefore, serves as the "*identifying* ..." step. And, further, the associated context(s) stored in the context stack corresponding to the user's intent in the active dialog constitutes the "*one or more contexts that completely or partially match*" the results of the recognized events ("*the one or more text combinations*") included in the decoded text/script ("*contained in the transcription*") from processing the spoken utterance. *See* Ex.1005, 7:65–8:4 and 37:53–55; Ex.1003, ¶110.

Further, as Dr. Houh explains, the "*identifying* ..." step would generate one or more candidate contexts selected from the context stack that potentially match (i.e., *completely or partially match*") the results of the recognized events ("*one or more text combinations*") included in the decoded text/script ("*contained in the transcription*"). Ex.1003, ¶¶110–111.

Therefore, Maes teaches or suggests this claim element. Ex.1003, ¶112.

Moreover, for the additional reasons below, Maes teaches or suggests this claim element. For example, Maes describes Figure 2 (below) as "a flow diagram illustrating a referential ambiguity resolution methodology performed by a multi-modal conversational computing system." Ex.1005, 3:24–28.

FIG. 2


200
START
OBTAIN RAW MULTI-MODAL INPUT DATA FROM MULTI-MODAL DATA SOURCES
202
I/O MANAGER ABSTRACTS RAW INPUT DATA INTO EVENTS
204
ONE OR MORE RECOGNITION ENGINES PROCESS EVENT, IF NECESSARY, BASED ON THE NATURE OF THE INPUT DATA
206
208
STORE EVENTS ON CONTEXT STACK
DIALOG MANAGER ATTEMPTS TO DETERMINE INTENT OF USER BASED ON PROCESSED ABSTRACT EVENT AND CONTEXT STACK HISTORY
210
212
NO
INTENT KNOWN ?
YES
214
DIALOG MANAGER CAUSES GENERATION OF OUTPUT TO USER TO OBTAIN FURTHER INPUT DATA
216
NO
CONFIRM INTENT ?
YES
EXECUTE PROCESSED EVENT OR OTHER ACTION
218

Maes, Figure 2. Referencing Figure 2, Maes explains that "[i]n step 208, the recognized events … are stored in a storage unit referred to as the context stack 20 [which] … is used to create a history of interaction between the user and the system **… to assist the dialog manager 18 in making referential ambiguity resolution determinations** when determining the user's intent. Next, in step 210, the system … attempts to **determine the user intent based on the current event** and **the historical interaction information stored in the context stack** ….” which serve as “*identifying one or more contexts that completely or partially match one or more text combinations contained in the transcription*”. Ex.1005, 7:33–45; Ex.1003, ¶113.

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

To the extent Patent Owner contends that Maes does not teach or suggest the “*identifying* ... ” step, Coffman (incorporated by reference in Maes[8]) discloses this element. Ex.1005, 9:30–37 (incorporating by reference specific features of Coffman identified as PCT/US99/22927); Ex.1003, ¶114.

Coffman explains that context stack entries are searched to identify a context matching one or more words transcribed/determined from the user's spoken utterance. “**Completion/search for the active context is performed from context to context down the stack. That is, new queries or arguments are compared by**

[8] *See* §VII.C.

the dialog engine by going down the stack until an acceptable match is obtained … ." Ex.1012, 42:1–3. Coffman further explains that searching for the context includes comparing words of the user's spoken utterance against information stored in the context stack:

> Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with **speech inputs being transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active query and complete it**.

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

> **Completion/search for the active context is performed from context to context down the stack**. That is, **new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained** ….

Ex.1012, 42:1–3. Thus, Coffman's teachings of searching the context stack until an acceptable match is found between one or more words transcribed/determined from the user's spoken utterance and data stored on the context stack serve as the "*identifying ...*" functionality. Indeed, Coffman teaches updating the context stack. (Ex.1012, 42:7–9) which further confirm Coffman provides the "*identifying ...*" functionality. Accordingly, based on Maes and its incorporation of Coffman[9], decoded text (or, script) ("*the transcription*") comprising "*one or more text combinations*" and produced as a result of processing the user's spoken utterance, as Maes teaches, would be compared against data stored on the context stack, per Coffman's teachings, to "*identify[] one or more contexts that completely or partially match one or more text combinations contained in the transcription.*" Ex.1003, ¶115.

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

Petitioner interprets "*the text combinations*" to mean "*the one or more text combinations*" and "*the* grammar expression entries in *the* context description grammar" to mean "grammar expression entries in a context description grammar." The combination of Maes and Ross renders obvious this claim element. Ex.1003, ¶117.

[9] *See* §VII.C

Maes teaches or suggests that the "*identifying* ..." includes "*comparing the text combinations ... against one or more expected contexts stored in a context stack.*" Ex.1003, ¶118.

Maes teaches or suggests "*one or more expected contexts stored in a context stack.*" As explained in connection with [12.2.0], Maes discloses "*one or more ... contexts stored in a context stack*" and, further, the contexts are "*expected contexts*" because Maes teaches such contexts that "could be associated" with transcribed user input, thereby signaling a possibility or expectation of finding such contexts:

> The dialog manager would … receive the results of the recognized events associated with the spoken utterance "turn it on" …. **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 … could be associated**.

Ex.1005, 7:65–8:4. Maes continues explaining that upon searching the context stack, the associated context(s) stored in the context stack corresponding to the user's intent in the active dialog ("*one or more expected contexts stored in a context stack*") is provided by the dialog manager when such context "**could be associated**" which serves as teaching or suggesting that contexts stored on the context stack are "*expected contexts.*" Ex.1003, ¶¶119–120.

Maes discloses information stored in the context stack relate to "context[s] corresponding to each active dialog," Ex.1005, 37:60–62. *See also* Ex.1005, 7:60–

8:6 ("[T]he dialog manager must … determine the user's intent based on the current event and … the historical information (e.g., past events) stored in the context stack. … The dialog manager would … receive the results of the recognized events …. Based on these events, the dialog manager … determines the appropriate [context]."). Therefore, Maes teaches or suggests "*one or more expected contexts stored in a context stack.*" Ex.1003, ¶121.

Moreover, for the reasons below, Maes discloses that the "*identifying…*" includes "*comparing the text combinations ... against one or more expected contexts stored in a context stack.*" For example, Maes teaches or suggests upon searching the context stack, the associated context stored in the context stack corresponding to the active dialog ("*one or more expected contexts stored in a context stack*") is provided by the dialog manager when such context "could be associated," which therefore constitutes "*comparing*" the results of the recognized events ("*the text combinations*") included in the decoded text/script (i.e., [*contained in the transcription*]") against data stored in the context stack (e.g., *one or more expected contexts stored in a context stack*). Ex.1005, 7:65–8:4; Ex.1003, ¶122.

Referencing context stack 817 depicted in Figure 8, Maes discloses

> context stack 817 comprises **all the information** associated with an application. Such information includes **all the variable, states, input, output and queries to the backend that are performed in the context of the dialog**

> and any extraneous event that occurs during the dialog. The context stack is associated with the organized/sorted context corresponding to each active dialog.

Ex.1005, 37:55–62. Because Maes teaches "**all** the information" and "**any** extraneous event" that occurs during the dialog is stored on the context stack, the context stack would includes "*one or more expected*"— contexts, therefore Maes teaches or suggests "*identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack*." Ex.1005, 37:55–62; *see also* 7:60–8:4 and 8:37–42; Ex.1003, ¶122.

Referring to the in-vehicle example, Maes explains the "*comparing* ..." as follows:

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

Ex.1005, 8:30–42. In the above example, "this event" corresponds to the current event of the user pointing to the radio, which occurred few seconds after "the previous spoken utterance event" of the user saying "turn it on. Data for both events [“*one or more contexts*”] are stored on the context stack because Maes discloses "all the variable, states, input, output and queries to the backend that are performed in the context of the dialog and any extraneous event that occurs during the dialog" are stored on the context stack. Ex.1005, 37:55–62. Therefore, in making a determination of the user's intent based on data for both events stored on the context, Maes teaches or suggests "*comparing the text combinations ... against one or more expected contexts stored in a context stack*" and the outcome of the "*comparing*" results in identification of "the user['s] inten[t] to have the radio turned on" ("*the matching context*"). Therefore, Maes discloses "*identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*" Ex.1003, ¶123.

To the extent Patent Owner contends that Maes does not teach or suggest "*identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack*," Coffman (incorporated in Maes) additionally discloses this step as explained below:

> Based on the modality (pointer, keyboard, file, speech), the task dispatcher 402 redirects the stream to the appropriate conversational subsystems or conventional subsystem with **speech inputs being**

> **transcribed/understood. The output of these subsystems is run down the context stack 405 to extract the active query and complete it.**

Ex.1012, 41:2–5. Thus, in Coffman, the outputs of conversational subsystems or conventional subsystems which constitute one or more words transcribed/determined from the user's spoken utterance, are compared against data stored in the context stack until a match is found. Thus, Coffman's teachings of searching the context stack until an acceptable match is found between one or more words transcribed/determined from the user's spoken utterance and data stored on the context stack serve as the "*identifying the matching contexts includes comparing the text combinations ... against one or more expected contexts stored in a context stack.*" Ex.1003, ¶124.

Coffman explains that identifying context entries stored on the context stack includes "*comparing*" data stored on the context stack to one or more words transcribed/determined from the user's spoken utterance. "**Completion/search for the active context is performed from context to context down the stack. That is, new queries or arguments are compared by the dialog engine by going down the stack until an acceptable match is obtained** … ." Ex.1012, 42:1–3. Therefore, Maes (based on incorporating Coffman) teaches or suggests "*identifying the* *matching contexts includes comparing the text combinations … against one or more expected contexts stored in a context stack.*" Ex.1003, ¶125.

Maes generally describes using "grammar" and "grammar database" in its system. Ex.1003, ¶126.

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. 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." Ex.1003, ¶127.

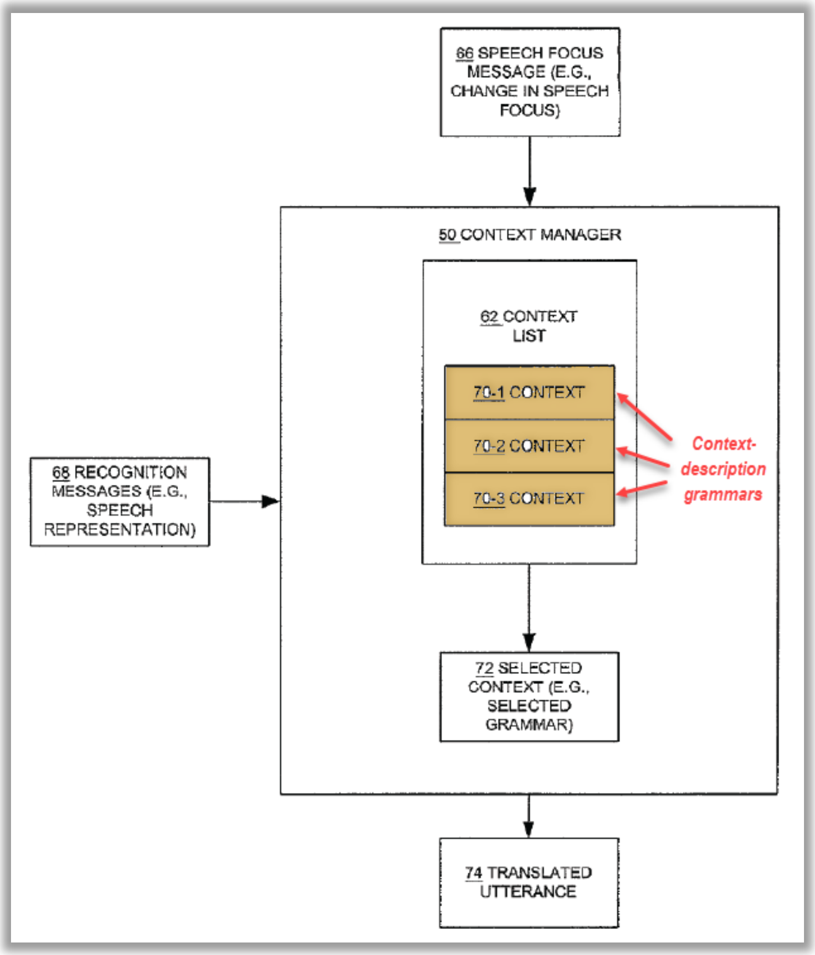
Maes describes using a "grammar database" stating: "Apparatus 900 … includes a processing module 910" which can further include "a speech recognizor

… which can … include … **grammar database 930**." Ex.1005, 39:59 and 41:35–38. Therefore, Maes discloses use of "grammar database." Ex.1003, ¶128.

Maes does not expressly disclose "*identifying the matching contexts includes comparing the text combinations against the grammar expression entries in the context description grammar*," but Ross teaches or suggests this limitation[10], as explained below. *See* Ex.1022, [0034]-[0035], [0041]-[0051]; Ex.1003, ¶129.

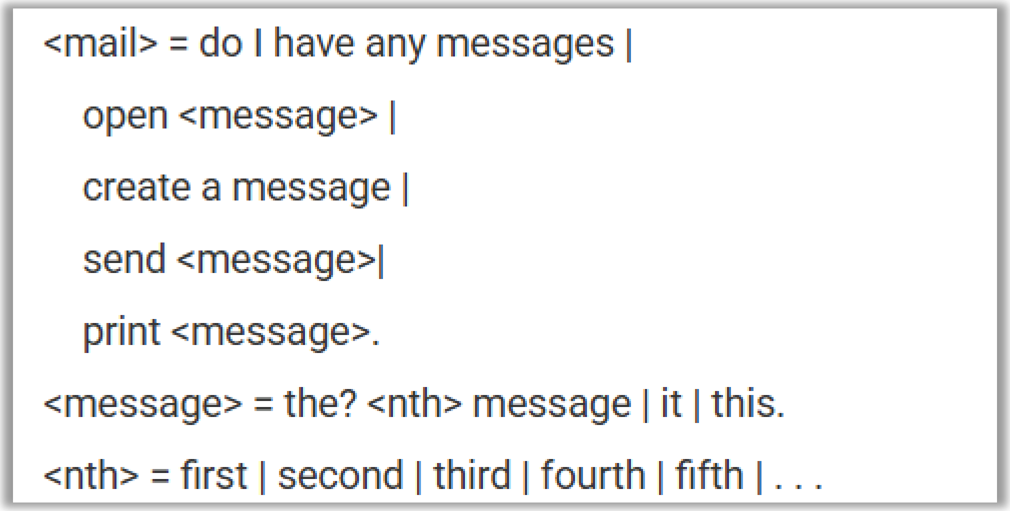
First, Ross teaches a "*grammar expression entries in [a] context description grammar.*" For example, Ross teaches "a grammar is defined for each application 26" and referencing Figure 4 (below), Ross discloses "**context list includes contexts 70 (e.g., 70-1, 70-1, 70-3, etc.) for speech-enabled applications 26, which represent the grammars for the applications 26**." Ex.1022, [0033] and [0035]. Because Ross describes that contexts 70-1, 70-1, 70-3 represent the grammars for the applications, Ross's contexts representing grammars (such as 70-1, 70-1, 70-3) constitute "*context description grammar[s]*" describing contexts for three speech-enabled applications. For instance, context 70-1 is a grammar for a first speech-enabled application, context 70-1 is a grammar for a second speech-enabled application, and context 70-3 is a grammar for a third speech-enabled application.

[10] In co-pending IPR2024-00753 re 160 Patent, the Board agreed on the record as to Ross disclosing the "*comparing …*" limitation. *See* Institution Decision, 9.


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

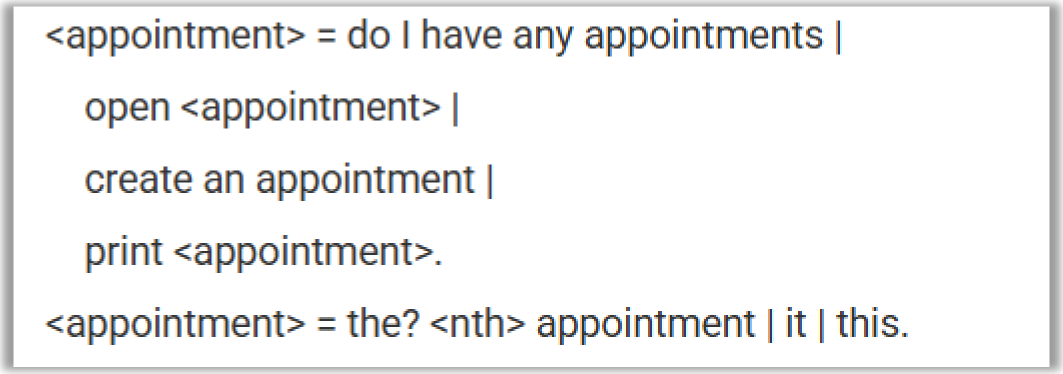
Ex.1022, Figure 4 (annotated); Ex.1003, ¶130.

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



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

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



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

(Ex.1022, [0046]-[0047]). The above-mentioned grammars comprise "*grammar expression entries*." Specifically, in the grammar for the electronic mail application shown above, the line "<message> = the? <nth> message | it | this," defines a rule named "message" which includes phrases (e.g., "the," "message"), a reference to another rule ("<nth>") and a grammar operator ("?"), which serve as examples of "*grammar expression entries*." Similarly, in the grammar for the calendar application shown above, phrases (e.g., "the," "appointment"), a reference to another rule ("<nth>") and a grammar operator ("?") serve as examples of "*grammar expression entries.*" Ex.1003, ¶131.

Additionally, Ross provides examples explaining how the entries (e.g., the phrases, keywords, and operators) included in a grammar are used. With respect to the grammar for the electronic mail application, Ross describes allowing a user's spoken phrases (e.g., processed to generate "*the text combinations*") such as "open the first message," "create a message," "send this," and "print it" to be matched ("*compar[ed]*") against entries in the grammar for the electronic mail application ("*the grammar expression entries in the context description grammar*"). The grammar for the calendar application allows matching ("*comparing*") of spoken phrases such as "open the first appointment," "create an appointment," "print the fourth appointment," and "print it" against entries in the grammar for the calendar application. *See generally* Ex.1022, [0041]-[0051]; Ex.1003, ¶132.

Ross also discloses that "*identifying the matching contexts includes*" 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 also explains that one goal of testing ("*comparing*") a recognized utterance (e.g., "*the text combinations*") against data included in context grammars ("*the grammar expression entries in the context description grammar*") is to find matches between the utterance and such grammars. "**When a successful match is found** [based on testing the recognition messages against the active grammars], the corresponding translation 74 is dispatched to the script engine 38 for execution, and the priority of the matching grammar (i.e., selected context 72) is raised." Ex.1022, [0034]. Therefore, Ross "uses a grammar to identify which speech enabled application is to receive the representation of the spoken utterance" and specifically "**to determine if a representation of a spoken utterance from a user is acceptable to (can be processed by) a particular speech-enabled application**." Ex.1022, [0013]. Ross further explains that "prior to evaluating the contexts," (the "*comparing*" step), its system "create[s] the contexts for the speech enabled applications in the speech enabled environment." Ex.1022, [0010]; Ex.1003, ¶133.

The disclosures in Ross related to contexts representing grammars (Ex.1022, [0033]) are analogous to the 160 Patent's description of "*context description grammar*" which describes it as including one or more grammar expression entries and "use[d] to process requests in respective contexts." Ex.1001, Cl. 1; Ex.1003, ¶134.

In view of Ross's above-mentioned teachings, Dr. Houh explains that it would have been obvious to a POSITA to modify Maes's system to additionally include Ross's teachings of context grammars for applications such that "*the text combinations*" produced as a result of processing the user's spoken utterance using Maes's system would be compared against data (e.g., keywords/phrases used in applications) included in context grammars, per Ross. *See* §VII.C (explaining in detail the combination of Maes and Ross). Ex.1003, ¶135.

Therefore, Maes in combination with Ross renders this element obvious. Ex.1003, ¶136.

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

First, as explained in connection with [12.2.0], in Maes, the associated context(s) stored in the context stack corresponding to the user's intent in the active dialog constitutes "*the identified matching contexts*" which equivalently are one or more candidate contexts selected from the context stack that potentially (e.g., completely or partially) match the results of the recognized events (e.g., included in the decoded text/script produced as a result of the transcription). And, furthermore, for the reasons below, Maes teaches or suggests this claim element. Ex.1003, ¶138.

Maes teaches scoring one or more candidate contexts such as phonemes, which Maes describes as "sub-phonetic or acoustic units of speech" (Ex.1005, 12:29–33) or equivalently as "portions [of dialogs]" (Ex.1005, 7:60–8:4). Specifically, Maes discloses generating, for each considered phoneme in a given context, likelihood scores (indicating the likelihood that it was that particular phoneme that was spoken) based, for example, on the received audio information (and/or video information). Ex.1003, ¶139.

Referencing Figure 4, Maes discloses "[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. Maes explains this in greater detail:

> **After the acoustic feature vectors … are extracted, the probability module labels the extracted vectors with one or more previously stored phonemes which … are sub-phonetic or acoustic units of speech. The module may also work with lefemes, which are portions of phones [sic: phonemes] in a given context. Each phoneme associated with one or more feature vectors**

> **has a probability associated therewith indicating the likelihood that it was that particular acoustic unit that was spoken.**
>
> 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, 12:29–43. Maes further describes that

> 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). …

See generally Ex.1005, 18:46–60. Calculation of likelihood scores for each considered phoneme in a given context in the form of probabilities, therefore, serves as "*scoring each of the identified matching contexts*." Ex.1003, ¶139

Therefore, Maes teaches or suggests this element. Ex.1003, ¶140.

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

As explained in connection with [12.3], Maes teaches or suggests generating likelihood scores for each considered phoneme in a given context ("*scoring each of the identified matching contexts*"). Specifically, and as explained in [12.3], Maes discloses generating, for each considered phoneme in a given context, likelihood scores (indicating the likelihood that it was that particular phoneme that was spoken) based, for example, on the audio information and/or the video information received as system input. Ex.1003, ¶141.

As Dr. Houh explains, after the likelihood scores are generated, Maes describes using such scores used in identifying ("*selecting*") "**the acoustic units** [i.e., "*the matching context*" / "*a most likely context*" identified in the context stack as explained in [12.2.0] and [12.2.1]]**… as having the highest probabilities [i.e., "a highest score"] of representing what was uttered**" ("*for the utterance*").. Ex.1005, 18:64–65 and 20:13–15. Thus, the candidate context (such as a phoneme or an acoustic unit) that has the highest probability of representing what was uttered corresponds to "the matching context" / "*a most likely context for the utterance.*" *See* Ex.1005, 18:64–65 and 20:13–15; Ex.1003, ¶142.

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

First, Maes teaches or suggests "*communicating a request to a domain agent to process requests in the most likely context for the utterance*." Ex.1003, ¶¶144–145.

Initially, the 160 Patent describes a "*domain agent*" as follows: "system 90 may include different types of agents 106. For example, generic and domain specific behavior and information may be organized into domain agents. …. The domain agents provide complete, convenient and re-distributable packages or modules for each application area." Ex.1001, 14:21–27. Thus, "*domain agent[s]*" in the 160 Patent broadly refer to software modules that are specific to each application area. Ex.1003, ¶146.

Maes teaches or suggests "*a domain agent configured to process requests in the most likely context for the utterance*:" Maes states "**determin[ing] and execut[ing] one or more application programs ["a *domain agent*"] that effectuate the user's intention and/or react to the user activity**" and that the application depends on the environment that the system is deployed in. Ex.1005, 7:40–46; Ex.1003, ¶147.

As such, Maes's teachings of application programs (effectuating the user's intention and depending on the environment that the system is deployed) are analogous to the 160 Patent's description of "*domain agent*," and therefore "application programs" or "applications" in Maes serve as "*domain agent[s]*." *See also* Ex.1005, 30:67 ("**run applications**"), 8:1–2 ("**search of the existing applications**"). Ex.1003, ¶148.

Maes describes that its system "**determines … which application(s) should handle the user inputs**" and specifically that the DMA "handles multi-modal I/O events to … **determine the target application or dialog (or portion of it)**." *See generally* Ex.1005, 36:49–37:3. Maes explains that "[w]hen an abstract event occurs, the DMA determines the target of the event and … launches the action associated to the user's query" which serve as "*process[ing] requests in the most likely context for the utterance.*" Ex.1005, 36:49–37:3. For example,

> [i]f the dialog manager determines … that the application it selects is the one which will effectuate the users desire, the dialog manager … launches the execution of an action associated to a fully understood multi-modal request from the user) of that application based on the multi-modal input. That is, the dialog manager selects the appropriate device (e.g., radio) activation routine and instructs the I/O manager to output a command to activate the radio.

Ex.1005, 8:8–8:19. For example, in Maes, a matching context corresponding to a fully understood request based on the user's input serves as "*the most likely context for the utterance.*" By teaching that Maes's dialog manager (1) determines which application(s) should handle the user inputs (such as a spoken utterance), (2) determines the target application or dialog (or portion of it), and (3) launches the execution of an action associated to a fully understood request from the user of that application based on the user's input, Maes teaches or suggests "*a domain agent configured to process requests in the most likely context for the utterance.*" Ex.1003, ¶148.

Referring to the in-vehicle example, Maes explains the "*communicating* ..." step: "the dialog manager selects the appropriate device (e.g., radio) activation routine and instructs the I/O manager to output a command to activate the radio." Ex.1005, 7:63–64, 8:15–19. Therefore, in this example, "the application" ("*the domain agent*") such as a "radio activation routine" handles the activation task "*configured to process requests*" "resulting from fall [sic:full] understanding of the intent of the user" ("*in the most likely context*") "*for*" the user's spoken utterance ("*the utterance*"). Ex.1005, 37:2–3. The user's query/command for a particular application corresponds to "*a request to the domain agent*" and launching the execution of an action (e.g., selecting an application such as a radio activation routine) associated with the user's query (e.g., comprising a request for launching an application such as a radio activation routine) serves as "*communicating a request to a domain agent configured to process requests in the most likely context for the utterance.*" Ex.1005, 7:60–8:29; *see also* 2:54–64; Ex.1003, ¶¶149–150.

Therefore, Maes teaches or suggests the "*communicating ...*" step. Ex.1003, ¶151.

Although Maes does not expressly disclose "*the request formulated using at least one grammar expression entry in the context description grammar*," Maes generally describes use of "grammar" and "grammar database" in its system (*see* [12.2.1]), and further, Ross discloses this feature. As explained in [12.2.1], Ross provides an example in which a user's query/command (e.g., "*[a] request*") utilizes spoken phrases ("*at least one grammar expression entry*") "*in*" a context representing the grammar ("*the context description grammar*") for a speech-enabled application. Ross describes that that the grammar for an electronic mail application allows "phrases to be spoken by a user" of the electronic mail application and the grammar for a calendar application allows "phrases to be spoken by a user." Ex.1022, [0041]-[0051]; Ex.1003, ¶152.

Ross describes "*using at least one grammar expression entry in the context description grammar*" to "*formulate[] [a] request.*" Ross explains that "a grammar is defined for each application." Ex.1022, [0035]. "When a successful match is found [based on testing the recognition messages against the active grammars for applications], **the corresponding translation 74 is dispatched to the script engine 38 for execution**." Ex.1022, [0034]-[0035]. Thus, the selected context 72 (comprising keywords and phrases) constitutes the specific grammar (e.g., "*the context description grammar*") that will accept the user's spoken utterance. Ex.1022, Figure 5 and [0037] (the context manager 50 **directs the speech representation to be translated according to the selected context 72**[,] … directs the translation to the script engine 38[,] … [and] 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."

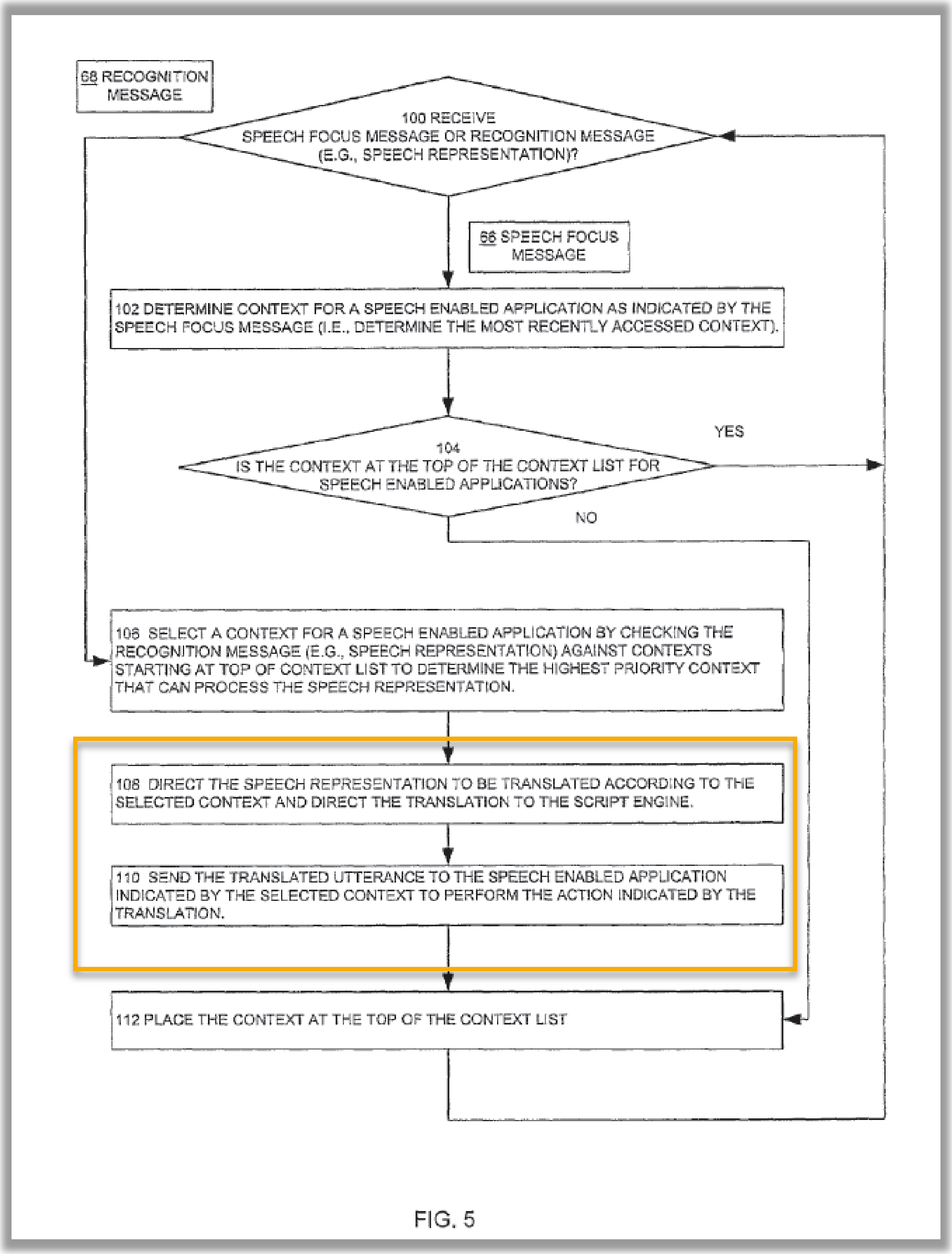

68 RECOGNITION MESSAGE
100 RECEIVE SPEECH FOCUS MESSAGE OR RECOGNITION MESSAGE (E.G., SPEECH REPRESENTATION)?
66 SPEECH FOCUS MESSAGE
102 DETERMINE CONTEXT FOR A SPEECH ENABLED APPLICATION AS INDICATED BY THE SPEECH FOCUS MESSAGE (I.E., DETERMINE THE MOST RECENTLY ACCESSED CONTEXT).
104 IS THE CONTEXT AT THE TOP OF THE CONTEXT LIST FOR SPEECH ENABLED APPLICATIONS?
YES
NO
106 SELECT A CONTEXT FOR A SPEECH ENABLED APPLICATION BY CHECKING THE RECOGNITION MESSAGE (E.G., SPEECH REPRESENTATION) AGAINST CONTEXTS STARTING AT TOP OF CONTEXT LIST TO DETERMINE THE HIGHEST PRIORITY CONTEXT THAT CAN PROCESS THE SPEECH REPRESENTATION.
108 DIRECT THE SPEECH REPRESENTATION TO BE TRANSLATED ACCORDING TO THE SELECTED CONTEXT AND DIRECT THE TRANSLATION TO THE SCRIPT ENGINE.
110 SEND THE TRANSLATED UTTERANCE TO THE SPEECH ENABLED APPLICATION INDICATED BY THE SELECTED CONTEXT TO PERFORM THE ACTION INDICATED BY THE TRANSLATION.
112 PLACE THE CONTEXT AT THE TOP OF THE CONTEXT LIST

FIG. 5

Ex.1022, Figure 5 (annotated).

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

Ex.1022, [0036]. The act of using the selected context grammar 72 for translating the speech representation (corresponding to the user's spoken utterance) constitutes "*formulat[ing]*" the speech representation (corresponding to the user's spoken utterance) ("*a request*"), for instance, "*using*" keywords/phrases ("*at least one grammar expression entry*") "*in*" the selected context ("*the context description grammar*"). In view of Ross's teachings, Dr. Houh explains that it would have been obvious to a POSITA to modify requests sent/communicated to the target application in Maes's system to additionally include Ross's teachings of context grammars for applications so that a request to the target application is constructed utilizing the context grammar for the target application, per Ross's teachings. *See* §VII.C (explaining in detail the combination of Maes and Ross); Ex.1003, ¶153.

Therefore, Maes in combination with Ross renders this element obvious. Ex.1003, ¶154.

2. Claim 13

a. [13.0] A method for processing natural language utterances, comprising:

See [12.0].

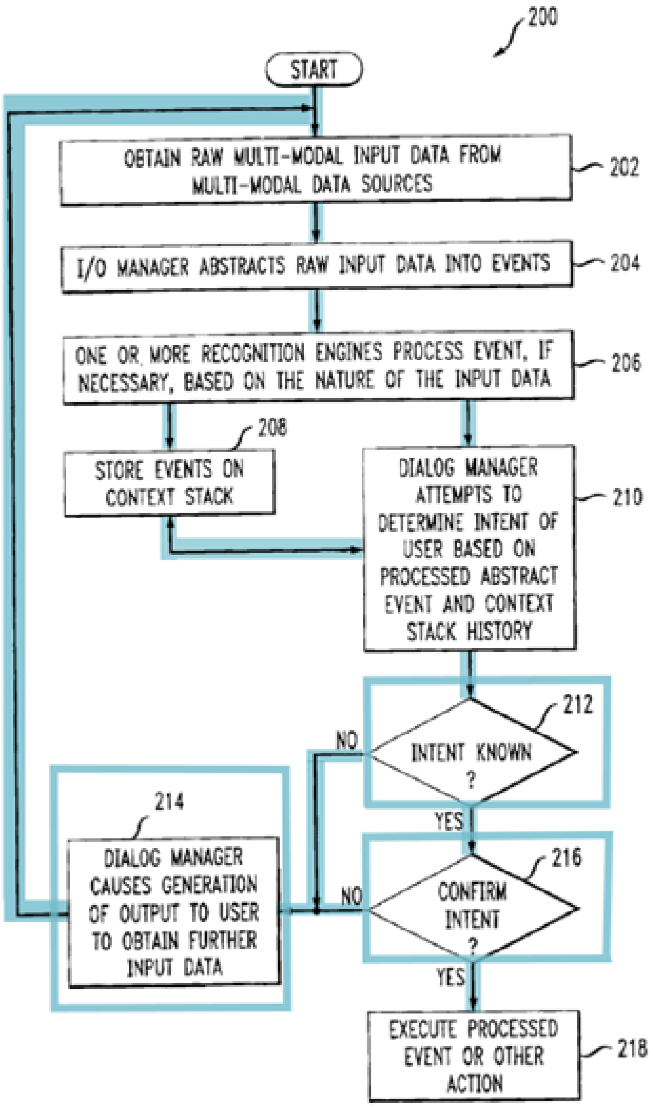
b. [13.1] receiving a natural language utterance at a computer comprising a multi-pass speech recognition module;

Other than replacing "*speech recognition **engine***" in [12.0] with "*speech recognition **module***" in this claim element, this element is substantively same as [12.0] above, which provides analysis related to Maes's disclosure of "*receiving a natural language utterance*" and [12.1] "*a computer comprising the ... speech recognition module*." Ex.1003, ¶156.

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

determined." Maes's system ("*speech recognition module*") implementing a process that performs multiple iterations ("*multi-pass*") to decode a natural language utterance until the user's intent is unambiguous and fully determined therefore constitutes a "*multi-pass speech recognition module.*" Ex.1003, ¶157 (citing Ex.1023, Abstract and Ex.1024, 545, each of which describe "*multi-pass speech recognition*" techniques like those described in Maes).

FIG. 2


200
START
OBTAIN RAW MULTI-MODAL INPUT DATA FROM MULTI-MODAL DATA SOURCES
202
I/O MANAGER ABSTRACTS RAW INPUT DATA INTO EVENTS
204
ONE OR MORE RECOGNITION ENGINES PROCESS EVENT, IF NECESSARY, BASED ON THE NATURE OF THE INPUT DATA
206
208
STORE EVENTS ON CONTEXT STACK
DIALOG MANAGER ATTEMPTS TO DETERMINE INTENT OF USER BASED ON PROCESSED ABSTRACT EVENT AND CONTEXT STACK HISTORY
210
212
NO
INTENT KNOWN ?
YES
214
DIALOG MANAGER CAUSES GENERATION OF OUTPUT TO USER TO OBTAIN FURTHER INPUT DATA
216
NO
CONFIRM INTENT ?
YES
EXECUTE PROCESSED EVENT OR OTHER ACTION
218

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

Therefore, Maes teaches or suggests a "*multi-pass speech recognition module*" at least for this reason. Ex.1003, ¶157.

To the extent one asserts that Maes does not expressly or implicitly disclose "*multi-pass speech recognition*" techniques, Dr. Houh states that those were well-known in the art, and it would have been obvious to implement such techniques in Maes's system based on such knowledge. For example, the paper "Automatic training set segmentation for multi-pass speech recognition" by Mao teaches "common approach to automatic speech recognition uses two recognition passes to decode an utterance: the first pass limits the search to a smaller set of likely hypotheses; and the second pass rescores the limited set using more detailed acoustic models which may target gender or specific channels." Ex.1023, Abstract. As another example, in 2005, a paper published by Hetherington" entitled "A multi-pass, dynamic-vocabulary approach to real-time, large-vocabulary speech recognition," describes "a multi-pass approach to certain types of large-vocabulary recognition tasks by refining the vocabulary between passes, while operating in real time." Ex.1024, 545. Hetherington states that "[a] multi-pass approach locating the state name in the first pass, the city name in the second pass, and finally the street in the third pass will almost certainly be preferable." Ex.1024, 548. Dr. Houh explains that a POSITA would have been motivated to implement such a "*multi-pass speech recognition*" in Maes based on the well-known knowledge in the art related to the improvement increases consistent with such techniques, as reflected in Mao and Hetherington, for example. Ex.1003, ¶158.

c. **[13.2] transcribing the utterance using the multi-pass speech recognition module, the multi-pass speech recognition module configured to transcribe the utterance into text;**

See [12.1]. Although [12.1] recites "*receiving a transcription of a natural language utterance*," "*receiving a transcription of a natural language utterance*" encompasses "*transcribing [an] utterance.*" Ex.1003, ¶160.

d. **[13.3] identifying one or more contexts that completely or partially match one or more text combinations contained in the text of the transcribed utterance,**

See [12.2.0].

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

See [12.2.1].

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

See [12.3].

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

See [12.4].

h. [13.6] communicating a request to a domain agent configured to process requests in the most likely context for the utterance, the request formulated using at least one grammar expression entry in the context description grammar.

See [12.5].

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

To the extent one asserts that Coffman is not expressly and/or particularly incorporated by reference into Maes as if they were effectively the same document, claims 12 and 13 would have nonetheless been obvious in view of Maes, Coffman and Ross for the reasons articulated above in Ground 1. Moreover, as demonstrated above, a POSITA would have been motivated to combine the references and had a reasonable expectation of success for such a combination as articulated in § VII.C. Ex.1003, ¶166.

IX. THE BOARD SHOULD NOT EXERCISE ITS DISCRETION TO DENY INSTITUTION

A. The Board Should Not Deny Institution Under 35 U.S.C. § 325(d)

Denial under § 325(d) is not warranted because the challenges presented in this Petition based on the combination of Maes and Ross are neither cumulative nor redundant to the prosecution of the 160 Patent, as neither Maes nor Ross were cited or considered by the Examiner during prosecution of the 160 Patent. Coffman, incorporated by reference in Maes (Ex.1005, 9:30–37), is relied upon in this Petition to the extent Patent Owner contends that Maes does not teach or suggest certain claim limitations. A "relative" of Coffman—the U.S. National stage of Coffman ("Coffman126" or Ex1013)—was used to reject the-then pending claims. Ex.1002, 197-209; Ex.1013, cover. To overcome the Examiner's rejections, Patent Owner argued that Coffman126 does not disclose the claimed features of "*multi-pass speech recognition*" and "*knowledge-enhanced speech recognition*" recited in issued claims 12 and 13. Ex.1002, 228–231. But those claimed features are disclosed in Maes, which was not before the Examiner. Because the Examiner did not review Maes and Ross (alone or in a combination) the Examiner's rejections over Coffman126 does not address the arguments as highlighted by the strength of Petitioner's Grounds. *CrowdStrike, Inc. v. Webroot Inc.,* IPR2023-00126, Paper 9 at 14 (PTAB May 5, 2023) ("[the Board] cannot determine the extent to which [cited reference] was evaluated during the examination because it was never used in a rejection" and because "[cited reference] was not the basis of a rejection, there is no overlap of arguments."); *Scientific Design Co. Inc. v. Shell Oil Co.*, IPR2022-00158, Paper 7 at 24-26 (PTAB Apr. 4, 2022) (declining to exercise discretion under § 325(d) based on "credit[ing] Petitioner's argument that although the references 'were cited in an IDS, together with 111 references, none of them was used in any rejections, or otherwise addressed by the examiner'"); *Bowtech, Inc. v. MCP IP, LLC,* IPR2019-

00379, Paper 14 at 18 (PTAB Jul. 3, 2019). The first condition of *Advanced Bionics* is therefore not satisfied.

Nevertheless, the Examiner materially erred by not substantively evaluating the Maes and Ross references (in conjunction with Coffman) against the 160 Patent and further erred by failing to reject any claims as obvious over Maes as highlighted by the strength of Petitioner's Proposed Grounds. Where a reference was never "substantively discussed by the Examiner[,]" "a petitioner's showing that the challenged claims are unpatentable over the asserted prior art may itself be evidence of material error by the Office during prosecution." *Quasar Science LLC v. Colt International Clothing, Inc. d/b/a Colt LED*, IPR2023-00611, Paper 10 at 14 (PTAB Oct. 10, 2023).

Further, the Examiner did not have the opportunity to review Dr. Houh's detailed expert testimony supporting this petition. The Board has found the *Becton* factors favor institution where even a single noncumulative secondary reference distinguished the prior-art ground, regardless of other overlapping references. *Oticon Medical AB et al. v. Cochlear Limited*, IPR2019-00975, Paper 15 at 9-20 (P.T.A.B. Oct. 16, 2019) (precedential). The Board has declined to deny institution where Petitioners relied upon prior art cited on the face of the challenged patent but "not applied by the examiner ... in any rejection of claims." *Comcast Cable*

Communs., LLC v. Promptu Sys. Corp., Case No. IPR2018-00342, Paper 13 at 17 (P.T.A.B. July 19, 2018).

B. The Board Should Not Deny Institution Under 35 U.S.C. § 314(a)

1. Factor 1: Stay

Factor is **neutral or weights against discretionary denial** because no request for stay has been filed in the litigation involving BOA, and Microsoft is not a defendant in that litigation. *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 15, 12 (May 13, 2020) (informative) ("*Fintiv*"); *Google, LLC v. Parus Holdings Inc.,* IPR2020-00847, Paper 9 at 12 (PTAB Oct. 21, 2020).

2. Factor 2: Trial Date

Microsoft is not a defendant in litigation involving this patent, although BOA—an RPI—is. Trial for BOA will not occur before January 26, 2026. Any FWD can be expected before the end of September 2026, which is eight months after the currently scheduled trial. "[A] court's general ability to set a fastpaced schedule is not particularly relevant … where, like here, the forum itself has not historically resolved cases so quickly." *In re Apple Inc.*, 979 F.3d 1332, 1344 (Fed. Cir. 2020). Indeed, there are **13 other trials** before Judge Gilstrap that are currently scheduled for January 26, 2026, so it appears likely that the trial date will slip. Ex.1029. Moreover, the trial date is 10 months away and much can change during this time. *Dish Network v. Broadband iTV*, IPR2020-01280, Paper 17 at 16 (PTAB Feb. 4,

2021) ("We cannot ignore the fact that the currently scheduled trial date is more than nine months away and much can change during this time."); *see also NetNut Ltd. v. Bright Data Ltd.,* IPR2021-01492, Paper 12 (PTAB Mar. 21, 2022) (granting institution when trial was to be held six months before the FWD).

Accordingly, this factor is **neutral or weighs against** discretionary denial.

3. Factor 3: Parallel Proceeding

Factor 3 **weighs against** discretionary denial. In the related litigation, claim construction briefing does not start until June 17, 2025, and the Claim Construction hearing is not scheduled until July 29, 2025. The investment by the parties to date has been relatively minimal.

4. Factor 4: Overlapping Issues

Consistent with *Sotera Wireless* Petitioner and BOA stipulate that if the PTAB institutes *inter partes* review of this proceeding, neither Petitioner nor BOA will pursue in any related district court case the grounds that were raised in this Petition, nor any other grounds that could have been reasonably raised in this Petition. *Sotera Wireless, Inc. v. Masimo Corp.*, IPR2020-01019, Paper 12 at 18-19 (PTAB Dec. 1, 2020) (precedential as to § II.A).

Accordingly, Factor 4 **weighs strongly against** discretionary denial and should be dispositive of the *Fintiv* analysis.

5. Factor 5: Overlapping Parties

Factor 5 is neutral or against discretionary denial. Microsoft provides software accused of infringement in related litigation. As *Fintiv* demonstrates, this factor should play a role only where a Petitioner is unrelated to a defendant in a district court proceeding. *See Fintiv*, IPR2020-00019, Paper 11 at 13–14 (PTAB Mar. 20, 2020).

6. Factor 6: Strength/Other Considerations

"[I]n circumstances where the Board determines that the other *Fintiv* factors 1–5 do not favor discretionary denial, the Board shall decline to discretionarily deny under *Fintiv* without reaching the compelling merits analysis." *CommScope Techs. LLC v. Dali Wireless, Inc.*, IPR2022-01242, Paper No. 23 at 4–5 (PTAB Feb. 27, 2023). For at least this reason, the Board need not reach the question on Factor 6.

Here, the merits of the Petition are particularly strong and compelling—for example, Maes (and, by incorporation Coffman) and Ross are strikingly similar disclosures to that of the 160 Patent, is directed at the same problem, proposes the same solutions as the 160 Patent, and discloses nearly identical functionality as the claimed "*method for processing natural language utterances.*" The evidence presented, if unrebutted, would lead to a conclusion that one or more claims are unpatentable by a preponderance of the evidence.

X. CONCLUSION

Claims 12 and 13 of the 160 Patent are unpatentable for the reasons discussed above.

Dated: March 10, 2025

Respectfully Submitted,

/Scott M. Border/
Scott M. Border
Reg. No. 77,744
Winston & Strawn LLP
1901 L Street, N.W.
Washington, D.C. 20036
T: 202-282-5054
sborder@winston.com

CERTIFICATE OF COMPLIANCE

I hereby certify that this brief complies with the type-volume limitations of 37 C.F.R. § 42.24, because it contains 13,201words (as determined by the Microsoft Word word-processing system used to prepare the brief and including annotated figures), excluding the parts of the brief exempted by 37 C.F.R. § 42.24.

Dated: March 10, 2025

Respectfully Submitted,

/Scott M. Border/

Scott M. Border
Reg. No. 77,744
Winston & Strawn LLP
1901 L Street, N.W.
Washington, D.C. 20036
T: 202-282-5054
sborder@winston.com

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6(e), I hereby certify that on this 10th day of March, 2025, I caused to be served a true and correct copy of the foregoing and any accompanying exhibits by Federal Express Priority Overnight on the following:

David Gerasimow
The Law Offices of David A. Gerasimow, P.C.
211 W. Wacker Dr.
Ste. 1717
Chicago, IL 60606

A courtesy copy of this Petition and supporting material was also served on litigation counsel for Patent Owner via email:

Garland Stephens (garland@bluepeak.law)
Justin Constant (justin@bluepeak.law)
Richard Koehl (richard@bluepeak.law)
Kate Falkenstien (kate@bluepeak.law)
Heng Gong (heng@bluepeak.law)
BLUE PEAK LAW GROUP LLP
3139 West Holcombe Blvd. PMB 8160
Houston, TX 77025

Dated: March 10, 2025 Respectfully Submitted,

/Scott M. Border/
Scott M. Border
Reg. No. 77,744
Winston & Strawn LLP
1901 L Street, N.W.
Washington, D.C. 20036
T: 202-282-5054
sborder@winston.com

CLAIM LISTING

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

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