



(10) **DE 10 2012 019 178 A1 2013.04.04**

(12)

## Offenlegungsschrift

(21) Aktenzeichen: **10 2012 019 178.2**

(22) Anmeldetag: **28.09.2012**

(43) Offenlegungstag: **04.04.2013**

(51) Int Cl.: **G06F 17/20 (2013.01)**

**G06F 3/16 (2013.01)**

(30) Unionspriorität:

**13/250,854**

**30.09.2011**

**US**

(72) Erfinder:

**Gruber, Thomas Robert, Cupertino, Calif., US; Brigham, Christopher Dean, Cupertino, Calif., US; Keen, Daniel S., Cupertino, Calif., US; Novick, Gregory, Cupertino, Calif., US; Phipps, Benjamin S., Cupertino, Calif., US**

(71) Anmelder:

**Apple Inc., Cupertino, Calif., US**

(74) Vertreter:

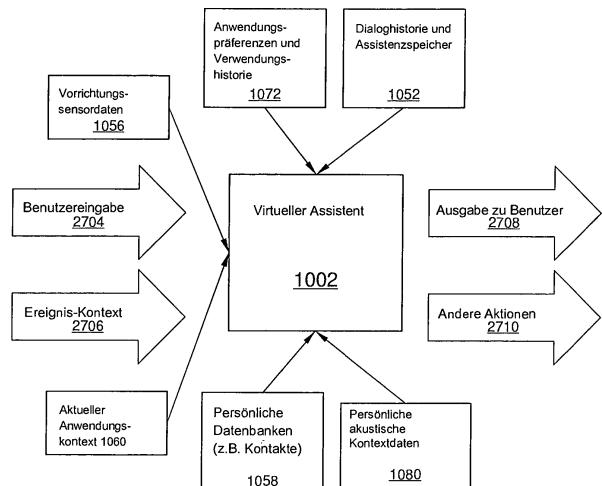
**BARDEHLE PAGENBERG Partnerschaft  
Patentanwälte, Rechtsanwälte, 81675, München,  
DE**

Prüfungsantrag gemäß § 44 PatG ist gestellt.

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen**

(54) Bezeichnung: **Verwendung von Kontextinformationen zum Erleichtern der Verarbeitung von Befehlen bei einem virtuellen Assistenten**

(57) Zusammenfassung: Ein virtueller Assistent verwendet Kontextinformation, um natürliche Sprache oder Gesteneingabe eines Benutzers zu ergänzen. Kontext hilft, die Absicht des Benutzers klarzustellen, und die Anzahl an Kandidateninterpretationen der Benutzereingabe zu reduzieren, und reduziert die Notwendigkeit für den Benutzer, übermäßige Klarstellungseingaben bereitzustellen. Kontext kann jegliche verfügbare Information beinhalten, die von dem Assistenten genutzt werden kann, um ausdrückliche Benutzereingabe zu ergänzen zum Beschränken eines Informationsverarbeitungsproblems und/oder zum Personalisieren von Ergebnissen. Kontext kann verwendet werden, um Lösungen während unterschiedlicher Phasen der Verarbeitung zu beschränken, einschließlich beispielsweise Spracherkennung, Verarbeitung natürlicher Sprache, Task-Fluss-Verarbeitung und Dialoggenerierung.



**Beschreibung****Gebiet der Erfindung**

Querverweise zu verbundenen Anmeldungen

**[0001]** Diese Anmeldung beansprucht als eine Continuation-in-part die Priorität der US-Utility Application Anmeldenr. 12/479,477, bezeichnet „Contextual Voice Commands“ („Kontextabhängige Sprachbefehle“), Aktenzeichen des Vertreters P7393US1, eingereicht am 5. Juni 2009, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0002]** Diese Anmeldung beansprucht weiterhin als Continuation-in-part die Priorität der US-Utility Application Anmeldenr. 12/987,982, bezeichnet „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0003]** US-Utility Application mit Anmeldenr. 12/987,982 beansprucht die Priorität der US-Provisional Patent Application mit Anmeldenr. 61/295,774, bezeichnet „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters SIRIP003P, eingereicht am 18. Januar 2010, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0004]** Diese Anmeldung beansprucht weiterhin die Priorität der US-Provisional Application mit Anmeldenr. 61/493,201, bezeichnet „Generating and Processing Data Items That Represent Tasks to Perform“ („Generieren und Verarbeiten von Daten-Items, die auszuführende Tasks darstellen“), Aktenzeichen des Vertreters P11337P1, eingereicht am 3. Juni 2011, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0005]** Diese Anmeldung ist verbunden mit US-Utility Application mit Anmeldenr. \_\_\_\_/\_\_\_\_, bezeichnet „Generating and Processing Task Items That Represent Tasks to Perform“ („Generieren und Verarbeiten von Task-Items, die auszuführende Tasks darstellen“), Aktenzeichen des Vertreters P11337US1, eingereicht am gleichen Tag wie die vorliegende Anmeldung, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0006]** Diese Anmeldung ist verbunden mit US-Utility Application mit Anmeldenr. \_\_\_\_/\_\_\_\_, bezeichnet „Automatically Adapting User Interfaces for Hands-Free Interaction“ („Automatisches Anpassen von Benutzerschnittstellen zur handfreien Interaktion“), Aktenzeichen des Vertreters P11357US1, eingereicht am gleichen Tag wie die vorliegende Anmeldung, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird.

**[0007]** Die vorliegende Erfindung bezieht sich auf virtuelle Assistenten, und spezifischer auf Mechanismen zum Verbessern der Interpretation und Verarbeitung von Befehlen, die einem solchen Assistenten bereitgestellt werden.

**Hintergrund der Erfindung**

**[0008]** Heutige elektronische Vorrichtungen sind in der Lage, auf eine große, wachsende und vielfältige Zahl von Funktionen, Diensten und Informationen zuzugreifen, sowohl über das Internet wie auch aus anderen Quellen. Die Funktionalität für solche Geräte wächst schnell an, da viele Vorrichtungen von Konsumenten, Smartphones, Tablet-Computer und Ähnliches in der Lage sind, Softwareanwendungen auszuführen, um verschiedene Tasks durchzuführen und unterschiedliche Typen von Information zur Verfügung zu stellen. Häufig weist jede(s) Anwendung, Funktion, Webseite oder Merkmal seine/ihre eigene Benutzerschnittstelle und seine/ihre eigenen Bedienungsparadigmen auf, wobei viele hiervon für Benutzer mühsam zu erlernen oder erdrückend sind. Zusätzlich haben viele Benutzer sogar Schwierigkeiten, zu entdecken, welche Funktionalität und/oder Information auf ihren elektronischen Vorrichtungen oder auf verschiedenen Webseiten verfügbar ist; folglich können solche Benutzer frustriert oder erdrückt werden, oder können schlicht nicht in der Lage sein, die ihnen zur Verfügung stehenden Ressourcen in einer effektiven Weise zu nutzen.

**[0009]** Insbesondere unbedarfte Benutzer oder Personen, die in irgendeiner Weise beeinträchtigt oder behindert sind, und/oder älter, beschäftigt, abgelenkt sind, und/oder ein Fahrzeug bedienen, können Schwierigkeiten haben, die Schnittstellen ihrer elektronischen Vorrichtungen effektiv zu bedienen und/oder Onlinedienste effektiv in Anspruch zu nehmen. Bei solchen Benutzern ist es besonders wahrscheinlich, dass sie Schwierigkeiten mit der großen Anzahl an vielfältigen und inkonsistenten Funktionen, Anwendungen und Webseiten, die zu ihrer Verwendung verfügbar sein können, haben.

**[0010]** Dementsprechend sind bestehende Systeme häufig schwierig zu benutzen und zu navigieren und präsentieren den Benutzern häufig inkonsistente und erdrückende Schnittstellen, die häufig die Benutzer davon abhalten, die Technologie effektiv zu verwenden.

**[0011]** Ein intelligenter, automatisierter Assistent, welcher hierin auch als ein virtueller Assistent bezeichnet wird, kann eine verbesserte Schnittstelle zwischen Mensch und Computer zur Verfügung stellen. Ein solcher Assistent, der implementiert werden kann, wie beschrieben in der verbundenen US-Utili-

ty Application mit Anmeldenr. 12/987,982, betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011, welche hiermit durch Verweis in ihrer Gesamtheit aufgenommen wird, erlaubt Benutzern mit einer Vorrichtung oder einem System unter Verwendung natürlicher Sprache in gesprochener und/oder Textform zu interagieren. Ein solcher Assistent interpretiert die Benutzereingaben, operationalisiert die Benutzerabsicht in Tasks und Parameter für diese Tasks, führt Dienste aus, um diese Tasks zu unterstützen und produziert Ausgaben, die für den Benutzer verständlich sind.

**[0012]** Ein virtueller Assistent kann auf jegliche aus einer Reihe von Informationsquellen zum Verarbeiten von Benutzereingaben zurückgreifen, einschließlich beispielsweise Wissensbasen, Modelle und/oder Daten. In vielen Fällen ist die Eingabe des Benutzers allein nicht ausreichend, um die Benutzerabsicht und den auszuführenden Task klar zu definieren. Dies kann auf das Rauschen in dem Eingabestrom, individuelle Unterschiede zwischen den Benutzern und/oder der inhärenten Mehrdeutigkeit von natürlicher Sprache zurückzuführen sein. Beispielsweise könnte der Benutzer einer Textnachrichtenwendung auf einem Telefon einen virtuellen Assistenten aufrufen und den Befehl „call her“ („rufe sie an“) sprechen. Während ein solcher Befehl völlig verständliches Englisch darstellt, handelt es sich nicht um eine präzise, ausführbare Aussage, da es viele Interpretationen und mögliche Lösungen für diese Anforderung gibt. Folglich mag ein virtueller Assistent, ohne weitere Information, nicht in der Lage sein, eine solche Eingabe korrekt zu interpretieren und zu verarbeiten. Mehrdeutigkeit dieses Typs kann zu Fehlern führen, dazu, dass falsche Aktionen ausgeführt werden und/oder dazu, dass der Benutzer übermäßig mit Anforderungen belastet wird, seine Eingabe klarzustellen.

### Zusammenfassung

**[0013]** Bei verschiedenen Ausführungsformen der vorliegenden Erfindung verwendet ein virtueller Assistent Kontextinformation (hierin auch als „Kontext“ bezeichnet), um die natürliche Sprache oder Gesteneingabe eines Benutzers zu ergänzen. Dies hilft, die Benutzerabsicht klarzustellen und die Anzahl an Kandidateninterpretationen der Eingabe des Benutzers zu reduzieren, und verringert den Bedarf für den Benutzer, übermäßige Klarstellungseingaben zur Verfügung zu stellen. Der Kontext kann jede verfügbare Information aufweisen, die von dem Assistenten verwendbar ist, um die ausdrückliche Benutzereingabe zu ergänzen, um ein Informationsverarbeitungsproblem zu beschränken und/oder die Ergebnisse zu personalisieren. Enthält beispielsweise die Eingabe des Benutzers ein Pronomen (so wie „sie“) in dem

Befehl „Rufe sie an“), kann der virtuelle Assistent den Kontext verwenden, um denjenigen herauszufinden, auf den sich das Pronomen bezieht, beispielsweise um die Identität der Person festzustellen, die anzurufen ist und/oder die zu verwendende Telefonnummer. Weitere Verwendung von Kontext wird hierin beschrieben.

**[0014]** Gemäß verschiedener Ausführungsformen der vorliegenden Erfindung kann jeder aus einer Reihe von Mechanismen zum Akquirieren und Anwenden von Kontextinformation implementiert werden, um Berechnungen in einem virtuellen Assistenten, der auf einer elektronischen Vorrichtung implementiert ist, durchzuführen. Bei verschiedenen Ausführungsformen stellt der virtuelle Assistent einen intelligenten, automatisierten Assistenten dar, wie er beschrieben ist in der US-Utility Application mit Anmeldenr. 12/987,982 betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011, welche in ihrer Gesamtheit durch Bezugnahme hierin eingebunden ist. Ein solcher Assistent interagiert mit dem Benutzer in einer integrierten, interaktiven Weise unter Verwendung eines natürlichen Sprachdialogs und ruft gegebenenfalls externe Dienste auf, um eine Information zu erhalten oder verschiedene Aktionen durchzuführen. Gemäß der hierin beschriebenen Techniken wird Kontextinformation in einem solchen Assistenten beispielsweise verwendet, um die Mehrdeutigkeit beim Durchführen von Informationsverarbeitungsfunktionen wie beispielsweise Spracherkennung, Verarbeitung natürlicher Sprache, Task-Fluss-Verarbeitung und Dialogerzeugung zu reduzieren.

**[0015]** Gemäß verschiedenen Ausführungsformen der vorliegenden Erfindung kann ein virtueller Assistent eingerichtet, gestaltet und/oder betreibbar sein, Kontext beim Durchführen mehrerer unterschiedlicher Typen von Operationen, Funktionalitäten und/oder Merkmalen zu verwenden, und/oder eine Vielzahl von Merkmalen, Operationen und Anwendungen auf einer elektronischen Vorrichtung, auf welcher dieser installiert ist, zu kombinieren. In einigen Ausführungsformen kann ein virtueller Assistent der vorliegenden Erfindung beim Durchführen beliebiger oder aller nachfolgenden Aktionen Kontext verwenden: aktives Auslösen von Eingaben eines Benutzers, Interpretieren von Benutzerabsicht, Eindeutigmachen zwischen konkurrierenden Interpretationen, Anfordern und Empfangen von klarstellenden erforderlichen Informationen und/oder Ausführen (oder Initiieren) von Aktionen basierend auf der erkannten Absicht.

**[0016]** Aktionen können beispielsweise durch das Aktivieren von und/oder dem Interagieren mit beliebigen Anwendungen oder Diensten, die auf einer elektronischen Vorrichtung verfügbar sind, wie

auch Dienste, die über ein elektronisches Netzwerk wie z. B. dem Internet verfügbar sind, durchgeführt werden. Bei verschiedenen Ausführungsformen kann ein solches Aktivieren von externen Diensten mittels Anwendungsprogrammierschnittstellen (APIs) oder mittels jeglichem anderen geeigneten Mechanismus ausgeführt werden. Auf diese Weise kann ein virtueller Assistent, der gemäß den verschiedenen Ausführungsformen der vorliegenden Erfindung implementiert ist, die Erfahrung des Benutzers im Hinblick auf viele unterschiedliche Anwendungen und Funktionen einer elektronischen Vorrichtung, und im Hinblick auf Dienste, die über das Internet verfügbar sind, vereinheitlichen, vereinfachen und verbessern. Dem Benutzer kann dadurch die Last abgenommen werden, zu erlernen, welche Funktionalität auf der Vorrichtung und von Web-verbundenen Diensten verfügbar ist, wie mit solchen Diensten zu interagieren ist, um zu erhalten, was er oder sie will, und wie die Ausgabe, die von diesen Diensten erhalten wird, zu verstehen ist; vielmehr kann der Assistent der vorliegenden Erfindung als ein Bindeglied zwischen dem Benutzer und solchen unterschiedlichen Diensten handeln.

**[0017]** Darüber hinaus kann in verschiedenen Ausführungsformen der virtuelle Assistent der vorliegenden Erfindung eine Konversationsschnittstelle zur Verfügung stellen, die der Benutzer intuitiver und weniger beschwerlich als herkömmliche grafische Benutzerschnittstellen finden kann. Der Benutzer kann in einer Form des interaktiven Dialogs mit dem Assistenten unter Verwendung beliebiger Mechanismen aus einer Reihe von verfügbaren Ein- und Ausgabemechanismen aktiv werden, wie beispielsweise Sprache, grafische Benutzerschnittstellen (Tasten und Links), Texteingabe und Ähnliches. Das System kann unter Verwendung jeglicher aus einer Reihe unterschiedlicher Plattformen implementiert werden, wie beispielsweise Vorrichtungs-APIs, das Web, E-Mail und Ähnliches, oder jegliche Kombination daraus. Anforderungen für zusätzliche Eingabe können dem Benutzer in dem Kontext einer solchen Konversation präsentiert werden. Kurz- und Langzeitspeicher können aktiviert werden, so dass die Benutzereingabe im richtigen Kontext interpretiert werden kann, angesichts von früheren Ereignissen und Kommunikationen innerhalb einer gegebenen Sitzung, wie auch angesichts von historischen Informationen und Profilinformationen über den Benutzer.

**[0018]** Zusätzlich kann bei verschiedenen Ausführungsformen die Kontextinformation, die aus einer Benutzerinteraktion mit einem Merkmal, einer Operation oder Anwendung abgeleitet wurde, verwendet werden, um die Bedienung von anderen Merkmalen, Operationen oder Anwendungen auf der Vorrichtung oder auf anderen Vorrichtungen zu optimieren. Beispielsweise kann der virtuelle Assistent den Kontext eines Telefonanrufes (wie beispielsweise die angeriefene Person) nutzen, um das Initiiieren einer Text-

nachricht zu optimieren (beispielsweise um zu bestimmen, dass die Textnachricht zu der gleichen Person gesendet werden soll, ohne dass der Benutzer den Empfänger der Textnachricht explizit spezifizieren muss). Der virtuelle Assistent der vorliegenden Erfindung kann dabei Instruktionen wie beispielsweise „Send ihm eine Textnachricht“ interpretieren, wobei das „ihm“ gemäß Kontextinformation, die aus einem gegenwärtigen Telefonanruf und/oder aus jedem Merkmal, jeglicher Bedienung oder Anwendung auf der Vorrichtung abgeleitet wird, interpretiert wird. Bei verschiedenen Ausführungsformen berücksichtigt der virtuelle Assistent verschiedene Typen von verfügbaren Kontextdaten, um zu bestimmen, welcher Adressbuchkontakt zu verwenden ist, welche Kontaktdata zu verwenden sind, welche Telefonnummer für den Kontakt zu verwenden ist und Ähnliches, so dass der Benutzer solche Informationen nicht manuell nochmals spezifizieren muss.

**[0019]** Kontextinformationsquellen beinhalten beispielsweise und ohne Einschränkung: den gegenwärtigen Zustand einer Vorrichtung, die als eine Schnittstelle zu dem Assistenten verwendet wird, wie beispielsweise die gegenwärtige Zeit, den Ort, die Anwendung oder das Datenobjekt; persönliche Daten wie beispielsweise ein Adressbuch des Benutzers, Kalender und Anwendungsverwendungshistorie; und den Zustand des Dialogs zwischen dem Benutzer und dem virtuellen Assistenten, wie beispielsweise zuletzt erwähnte Personen und/oder Orte.

**[0020]** Der Kontext kann für eine Vielfalt von Berechnungen und Rückschlüsse bei der Bedienung des virtuellen Assistenten angewendet werden. Beispielsweise kann der Kontext verwendet werden, um die Mehrdeutigkeit zu reduzieren, oder anderweitig die Anzahl von Lösungen zu beschränken bei Verarbeitung der Benutzereingabe. Der Kontext kann folglich verwendet werden, um die Lösungen während verschiedener Phasen der Verarbeitung zu beschränken, einschließlich beispielsweise und ohne Einschränkung:

- Spracherkennung – Empfangen von Spracheingabe und Generieren von Kandidateninterpretationen als Text, beispielsweise „Rufe sie an“ („call her“), „Kragen“ („collar“) und „Rufe Herb an“ („call Herb“). Kontext kann verwendet werden, um einzuschränken, welche Wörter und Phrasen von einem Spracherkennungsmodul berücksichtigt werden, wie sie eingestuft werden und welche als oberhalb eines Schwellwertes zur Berücksichtigung akzeptiert werden. Beispielsweise kann das Adressbuch des Benutzers persönliche Namen einem anderweitig allgemeinsprachigen Modell von Sprache hinzufügen, so dass diese Namen erkannt werden können und ihnen Vorrang gegeben werden kann.
- Natürliche Sprachverarbeitung (Natural Language Processing, NLP) – Parsen von Text und Asso-

ziieren der Wörter mit syntaktischen und semantischen Rollen, beispielsweise Bestimmen, dass die Benutzereingabe das Durchführen eines Telefonanrufes zu einer Person, welche durch das Pronomen „sie“ bezeichnet wurde, betrifft, und Auffinden einer spezifischen Datendarstellung für diese Person. Beispielsweise kann der Kontext einer Textnachrichtenanwendung helfen, die Interpretation von „sie“ einzuschränken, so dass gemeint ist „die Person mit der ich mich über Textnachricht unterhalte“.

- Task-Fluss-Verarbeitung – Identifizierung eines Tasks eines Benutzers, von Taskschritten und Task-Parametern, die verwendet werden, um den Task zu unterstützen, beispielsweise, welche Telefonnummer für die Person, die als „sie“ bezeichnet wurde, zu verwenden ist. Erneut kann der Kontext der Textnachrichtenanwendung die Interpretation der Telefonnummer einschränken, um anzugeben, dass das System die gegenwärtig oder kürzlich für eine Textnachrichtenunterhaltung verwendete Nummer verwenden soll.
- Dialoggenerierung – Generieren von Antworten des Assistenten als Teil einer Unterhaltung mit dem Benutzer über ihre Task, beispielsweise um die Benutzerabsicht mit der Antwort „OK, ich werde Rebecca auf ihrem Mobiltelefon... anrufen“ zu paraphrasieren. Das Ausmaß an Wortfülle und informellem Umgangston sind Wahlmöglichkeiten, die sich an Kontextinformation orientieren können.

**[0021]** Bei verschiedenen Ausführungsformen kann der virtuelle Assistent der vorliegenden Erfindung verschiedene Merkmale und Operationen einer elektronischen Vorrichtung steuern. Beispielsweise kann der virtuelle Assistent Dienste, die eine Schnittstelle mit Funktionalität und Anwendungen auf einer Vorrichtung haben, über APIs oder über andere Mittel aufrufen, um so Funktionen und Operationen auszuführen, die anderweitig unter Verwendung einer konventionellen Benutzerschnittstelle auf der Vorrichtung initiiert werden könnten. Solche Funktionen und Operationen können beispielsweise beinhalten: Einstellen eines Alarms, Durchführen eines Telefonanrufs, Senden einer Textnachricht oder E-Mail-Nachricht, Hinzufügen eines Kalenderereignisses und Ähnliches. Solche Funktionen und Operationen können als Add-on-Funktionen in dem Kontext eines interaktiven Dialogs zwischen einem Benutzer und dem Assistenten ausgeführt werden. Solche Funktionen und Operationen können durch den Benutzer in dem Kontext eines solchen Dialogs spezifiziert werden, oder sie können automatisch basierend auf dem Kontext des Dialogs ausgeführt werden. Ein Fachmann wird erkennen, dass der Assistent dabei als ein Steuermechanismus zum Initiiieren und Steuern verschiedener Operationen auf der elektronischen Vorrichtung verwendet werden kann, welcher als eine Alternative zu konventionellen Mechanismen wie beispielsweise Tasten oder grafische

Benutzerschnittstellen verwendet werden kann. Wie hierin beschrieben kann Kontextinformation verwendet werden, um über eine solche Verwendung des virtuellen Assistenten als einen Steuermechanismus zu informieren und diese zu verbessern.

#### Kurze Beschreibung der Zeichnungen

**[0022]** Die beigefügten Zeichnungen illustrieren mehrere Ausführungsformen der Erfindung und dienen, zusammen mit der Beschreibung, dazu, die Prinzipien der Erfindung gemäß der Ausführungsformen zu erklären. Ein Fachmann wird erkennen, dass die bestimmten in den Zeichnungen illustrierten Ausführungsformen lediglich beispielhaft sind, und sollen nicht den Bereich der vorliegenden Erfindung beschränken.

**[0023]** [Fig. 1](#) ist ein Blockdiagramm, welches einen virtuellen Assistenten und einige Beispiele von Quellen von Kontext zeigt, die seine Operation gemäß einer Ausführungsform beeinflussen können.

**[0024]** [Fig. 2](#) ist ein Flussdiagramm, welches ein Verfahren zur Verwendung von Kontext bei verschiedenen Stufen der Verarbeitung in einem virtuellen Assistenten zeigt, gemäß einer Ausführungsform.

**[0025]** [Fig. 3](#) ist ein Flussdiagramm, welches ein Verfahren zur Verwendung von Kontext bei der Ermittlung und Interpretation von Sprache zeigt, gemäß einer Ausführungsform.

**[0026]** [Fig. 4](#) ist ein Flussdiagramm, welches ein Verfahren zur Verwendung von Kontext bei natürlicher Sprachverarbeitung zeigt, gemäß einer Ausführungsform.

**[0027]** [Fig. 5](#) ist ein Flussdiagramm, welches ein Verfahren zur Verwendung von Kontext bei Task-Fluss-Verarbeitung zeigt, gemäß einer Ausführungsform.

**[0028]** [Fig. 6](#) ist ein Blockdiagramm, welches ein Beispiel von Quellen von Kontext, die zwischen einem Client und Server verteilt sind, zeigt, gemäß einer Ausführungsform.

**[0029]** [Fig. 7a](#) bis [Fig. 7d](#) sind Ereignisdiagramme, welche Beispiele von Mechanismen zum Erhalten und Koordinaten von Kontextinformation gemäß verschiedener Ausführungsformen zeigen.

**[0030]** [Fig. 8a](#) bis [Fig. 8d](#) zeigen Beispiele von verschiedenen Darstellungen von Kontextinformation wie sie in Verbindung mit verschiedenen Ausführungsformen der vorliegenden Erfindung verwendet werden kann.

[0031] [Fig. 9](#) zeigt ein Beispiel einer Konfigurations-tabelle, die Kommunikations- und Caching-Strategien für verschiedene Kontextinformationsquellen spezifiziert, gemäß einer Ausführungsform.

[0032] [Fig. 10](#) ist ein Ereignisdiagramm, welches ein Beispiel zum Zugreifen auf die in [Fig. 9](#) konfigurierten Kontextinformationsquellen während der Verarbeitung einer Interaktionssequenz zeigt, gemäß einer Ausführungsform.

[0033] [Fig. 11](#) bis [Fig. 13](#) stellen eine Serie von Bildschirmkopien dar, die ein Beispiel der Verwendung von Anwendungskontext bei einem Textnachrichtenbereich zeigt, um einen Bezug für ein Pronomen abzuleiten, gemäß einer Ausführungsform.

[0034] [Fig. 14](#) ist eine Bildschirmkopie, die einen virtuellen Assistenten darstellt, der zum Eindeutig-Machen eines Namens auffordert, gemäß einer Ausführungsform.

[0035] [Fig. 15](#) ist eine Bildschirmkopie, die einen virtuellen Assistenten darstellt, der Dialogkontext verwendet, um den Ort für einen Befehl zu erschließen, gemäß einer Ausführungsform.

[0036] [Fig. 16](#) ist eine Bildschirmkopie, die ein Beispiel der Verwendung einer Telefon-Favoriten-Liste als eine Quelle von Kontext zeigt, gemäß einer Ausführungsform.

[0037] [Fig. 17](#) bis [Fig. 20](#) stellen eine Serie von Bildschirmkopien dar, die ein Beispiel der Verwendung von gegenwärtigem Anwendungskontext zeigen, um einen Befehl zu interpretieren und zu operationalisieren, gemäß einer Ausführungsform.

[0038] [Fig. 21](#) ist eine Bildschirmkopie, die ein Beispiel der Verwendung des gegenwärtigen Anwendungskontext zeigt, um einen Befehl zu interpretieren, der eine unterschiedliche Anwendung aufruft.

[0039] [Fig. 22](#) bis [Fig. 24](#) sind eine Serie von Bildschirmkopien, die ein Beispiel der Verwendung von Ereigniskontext in der Form einer eingehenden Textnachricht zeigen, gemäß einer Ausführungsform.

[0040] [Fig. 25A](#) und [Fig. 25B](#) sind eine Serie von Bildschirmkopien, die ein Beispiel der Verwendung von vorangehendem Dialogkontext zeigen, gemäß einer Ausführungsform.

[0041] [Fig. 26A](#) und [Fig. 26B](#) sind Bildschirmkopien, die ein Beispiel einer Benutzerschnittstelle zur Auswahl zwischen Kandidateninterpretationen zeigen, gemäß einer Ausführungsform.

[0042] [Fig. 27](#) ist ein Blockdiagramm, welches ein Beispiel einer Ausführungsform eines virtuellen Assistenten-Systems zeigt.

[0043] [Fig. 28](#) ist ein Blockdiagramm, welches eine Rechenvorrichtung zeigt, die geeignet ist, zumindest einen Teil eines virtuellen Assistenten nach zumindest einer Ausführungsform zu implementieren.

[0044] [Fig. 29](#) stellt ein Blockdiagramm dar, welches eine Architektur zum Implementieren von zumindest einem Teil eines virtuellen Assistenten auf einem unabhängigen Computersystem zeigt, gemäß zumindest einer Ausführungsform.

[0045] [Fig. 30](#) stellt ein Blockdiagramm dar, welches eine Architektur zum Implementieren von zumindest einem Teil eines virtuellen Assistenten auf einem verteilten Rechnernetzwerk zeigt, gemäß zumindest einer Ausführungsform.

[0046] [Fig. 31](#) stellt ein Blockdiagramm dar, welches eine Systemarchitektur zeigt, die mehrere unterschiedliche Typen von Clients und Operationsmodi zeigt.

[0047] [Fig. 32](#) stellt ein Blockdiagramm dar, welches einen Client und einen Server zeigt, die miteinander kommunizieren, um die vorliegende Erfindung gemäß einer Ausführungsform zu implementieren.

#### Detaillierte Beschreibung der Ausführungsformen

[0048] Gemäß verschiedener Ausführungsformen der vorliegenden Erfindung wird eine Vielfalt von Kontextinformationen erfasst und angewendet, um Informations-Verarbeitungs-Funktionen zur Unterstützung der Operationen eines virtuellen Assistenten auszuführen. Für die Zwecke der Beschreibung ist der Begriff „virtueller Assistent“ äquivalent zu dem Begriff „intelligenter automatisierter Assistent“, wobei sich beide auf jegliches Informations-Verarbeitungs-System beziehen, welches eine oder mehrere der folgenden Funktionen ausführt:

- Interpretieren von menschlicher Spracheingabe in gesprochener Form und/oder Textform;
- Operationalisieren einer Darstellung einer Absicht eines Benutzers in eine Form, die ausgeführt werden kann, wie beispielsweise eine Darstellung eines Tasks mit Schritten und/oder Parametern;
- Ausführen von Task-Darstellungen, durch Aufrufen von Programmen, Methoden, Diensten, APIs und Ähnliches; und
- Generieren von Ausgabe-Antworten für den Benutzer in Sprachform und/oder grafischer Form.

[0049] Ein Beispiel eines solchen virtuellen Assistenten ist in der verbundenen US-Utility Application mit Anmeldenr. 12/987,982, betreffend „Intelligent Automated Assistant“ („Intelligenter Automatisierter

Assistent"), Aktenzeichen des Vertreters P10575 US1, eingereicht am 10. Januar 2011, beschrieben, wobei die gesamte Offenbarung dieser Anmeldung durch Bezugnahme hierin eingebunden ist.

**[0050]** Verschiedene Techniken werden nun im Einzelnen unter Bezugnahme auf die beispielhaften Ausführungsformen, wie sie in den beigefügten Zeichnungen dargestellt sind, beschrieben. In der nachfolgenden Beschreibung werden zahlreiche spezifische Einzelheiten dargelegt, um ein tiefgründiges Verständnis von einem oder mehreren hierin beschriebenen oder referenzierten Aspekten und/oder Merkmalen bereitzustellen. Es wird jedoch einem Fachmann offensichtlich sein, dass ein oder mehrere der hierin beschriebenen oder referenzierten Aspekte und/oder Merkmale ausgeführt werden können, ohne einige oder all diese spezifischen Einzelheiten. Bei anderen Gelegenheiten wurden wohl bekannte Prozessschritte und/oder Strukturen nicht im Einzelnen beschrieben, um einige der hierin beschriebenen oder referenzierten Aspekte und/oder Merkmale nicht unklar zu machen.

**[0051]** Eine oder mehrere unterschiedliche Erfindungen können in der vorliegenden Anmeldung beschrieben sein. Weiterhin können zahlreiche Ausführungsformen für eine oder mehrere der hierin beschriebenen Erfindung(en) in dieser Patentanmeldung beschrieben sein, und sind nur für erklärende Zwecke dargestellt. Die beschriebenen Ausführungsformen sollen nicht in irgendeinem Sinne beschränkend sein. Eine oder mehrere der Erfindung(en) können breit für zahlreiche Ausführungsformen anwendbar sein, wie aus der Offenbarung bereits offensichtlich ist. Diese Ausführungsformen werden in ausreichenden Einzelheiten beschrieben, um den Fachleuten zu ermöglichen, eine oder mehrere der Erfindung(en) auszuführen, und es soll verstanden werden, dass andere Ausführungsformen verwendet werden können, sowie, dass strukturelle, logische, Software-, elektrische und andere Änderungen gemacht werden können, ohne von dem Bereich der einen oder mehreren der Erfindung(en) abzurücken. Dementsprechend werden die Fachleute erkennen, dass die eine oder die mehreren der Erfindung(en) mit verschiedenen Modifikationen und Veränderungen ausgeführt werden können. Besondere Merkmale von einer oder mehrerer der Erfindung(en) können unter Bezugnahme auf eine oder mehrere besondere Ausführungsformen oder Figuren beschrieben sein, die einen Teil der vorliegenden Offenbarung darstellen und in welchen zum Zwecke der Darstellung spezifische Ausführungsformen von einer oder mehrerer der Erfindung(en) gezeigt werden. Es soll jedoch verstanden werden, dass solche Merkmale nicht auf die Verwendung in der einen oder den mehreren bestimmten Ausführungsformen oder Figuren, für welche sie beschrieben sind, beschränkt sind. Die vorliegende Offenbarung stellt weder eine wörtliche Beschreibung

aller Ausführungsformen von einer oder mehrerer der Erfindung(en) dar, noch eine Auflistung von Merkmalen von einer oder mehrerer der Erfindung(en), welche in allen Ausführungsformen vorhanden sein muss.

**[0052]** Überschriften von Abschnitten, die in dieser Patentanmeldung bereitgestellt werden und der Titel dieser Patentanmeldung dienen lediglich der Zweckmäßigkeit, und dürfen nicht so verstanden werden, dass sie die Offenbarung in irgendeiner Weise einschränken.

**[0053]** Vorrichtungen, welche miteinander in Kommunikation stehen, müssen nicht in kontinuierlicher Kommunikation miteinander stehen, soweit nicht anderweitig ausdrücklich spezifiziert. Darüber hinaus können Vorrichtungen, die in Kommunikation miteinander stehen, direkt oder indirekt über einen oder mehrere Vermittler miteinander kommunizieren.

**[0054]** Eine Beschreibung einer Ausführungsform mit mehreren miteinander in Kommunikation stehenden Komponenten impliziert nicht, dass alle solche Komponenten benötigt werden. Im Gegenteil: es wird eine Vielfalt von optionalen Komponenten beschrieben, um die breite Vielfalt von möglichen Ausführungsformen von einer oder mehrerer der Erfindung(en) darzustellen.

**[0055]** Obwohl weiterhin Prozessschritte, Verfahrensschritte, Algorithmen oder Ähnliches in einer sequenziellen Reihenfolge beschrieben sein können, können solche Prozesse, Verfahren und Algorithmen so eingerichtet sein, dass sie in jeder geeigneten Reihenfolge funktionieren. Anders gesagt stellt jegliche Sequenz oder Reihenfolge von Schritten, die in dieser Patentanmeldung beschrieben sein können, an und für sich, keine Anforderung dar, dass die Schritte in dieser Reihenfolge ausgeführt werden müssen. Weiterhin können einige Schritte gleichzeitig ausgeführt werden, obwohl beschrieben oder impliziert ist, dass sie nicht gleichzeitig auftreten (z. B. weil ein Schritt nach dem anderen Schritt beschrieben ist). Darüber hinaus impliziert die Darstellung eines Prozesses durch seine Wiedergabe in einer Zeichnung nicht, dass der dargestellte Prozess andere Varianten und Modifikationen daran ausschließt, dass der dargestellte Prozess oder jeglicher seiner Schritte für eine oder mehrere der Erfindung(en) notwendig ist und dass der dargestellte Prozess bevorzugt ist.

**[0056]** Wenn eine einzelne Vorrichtung oder ein einzelner Gegenstand beschrieben ist, wird es ohne Weiteres ersichtlich sein, dass mehr als ein(e) Vorrichtung/Gegenstand (unabhängig davon, ob sie kooperieren) anstelle einer einzelnen/eines einzelnen Vorrichtung/Gegenstandes verwendet werden kann. Ähnlich wird es ohne Weiteres ersichtlich sein, dass, wo mehr als eine Vorrichtung oder ein Gegenstand

beschrieben sind (unabhängig davon, ob sie kooperieren), eine einzelne Vorrichtung/ein einzelner Gegenstand verwendet werden kann anstelle der/des mehr als einen Vorrichtung oder Gegenstandes.

**[0057]** Die Funktionalität und/oder die Merkmale einer Vorrichtung können alternativ durch eine oder mehrere andere Vorrichtungen verkörpert sein, für die nicht ausdrücklich beschrieben ist, dass sie solche Funktionalität/Merkmale aufweisen. Folglich müssen andere Ausführungsformen von einer oder mehrerer der Erfindung(en) die Vorrichtung selbst nicht beinhalten.

**[0058]** Hierin beschriebene Techniken und Mechanismen werden gelegentlich zu Klarheitszwecken in Singularform beschrieben. Jedoch soll beachtet werden, dass bestimmte Ausführungsformen mehrere Iterationen einer Technik oder mehrere Instanziierungen eines Mechanismus beinhalten können, soweit nicht anderweitig angegeben.

**[0059]** Obwohl die hierin beschriebenen verschiedenen Aspekte und Techniken innerhalb des Kontexts von Technologie zum Implementieren eines intelligenten, automatisierten Assistenten, auch als virtueller Assistent bekannt, beschrieben werden, kann verstanden werden, dass diese auch in anderen Gebieten von Technologie, die menschliche und/oder rechnergestützte Interaktion mit Software involvieren, betrieben und/oder angewendet werden.

**[0060]** Andere Aspekte, die sich auf virtuelle Assistenz-Technologie beziehen (z. B. die von einem oder mehreren hierin beschriebener Ausführungsformen von virtuellen Assistenz-Systemen verwendet werden, zur Verfügung gestellt werden und/oder implementiert werden), werden in einer oder mehr der Folgenden beschrieben, deren gesamte Offenbarung hierin mittels Bezugnahme eingebunden ist:

- US-Utility Application Anmeldenr. 12/987,982 betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011;
- US-Provisional Patent Application Anmeldenr. 61/295,774 betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters SIRIP003P, eingereicht am 18. Januar 2010;
- US-Patent Application Anmeldenr. 11/518,292 betreffend „Method And Apparatus for Building an Intelligent Automated Assistant“ („Verfahren und Vorrichtung zum Errichten eines intelligenten automatisierten Assistenten“), eingereicht am 8. September 2006; und

- US-Provisional Patent Application Anmeldenr. 61/186,414 betreffend „System and Method for Semantic Auto-Completion“ („System und Verfahren für semantische Autovervollständigung“), eingereicht am 12. Jni 2009.

## Hardware Architektur

**[0061]** Im Allgemeinen können die hierin offenbarten virtuellen Assistenz-Techniken in Hardware oder einer Kombination von Software und Hardware implementiert werden. Beispielsweise können sie in einem Betriebssystemkern implementiert sein, in einem getrennten Benutzerprozess, in einem Bibliothekspaket, welches in Netzwerkanwendungen eingebunden ist, auf einer spezifisch konstruierten Maschine und/oder auf einer Netzwerkschnittstellenkarte. Bei einer spezifischen Ausführungsform können die hierin offenbarten Techniken in Software implementiert sein, wie beispielsweise ein Betriebssystem oder in einer Anwendung, die auf dem Betriebssystem läuft.

**[0062]** Hybride Software/Hardware-Implementierung(en) von zumindest einigen der Ausführungsform(en) des virtuellen Assistenten, die hierin beschrieben sind, können auf einer programmierbaren Maschine so implementiert sein, dass sie durch ein im Speicher vorgehaltenes Computerprogramm selektiv aktiviert oder rekonfiguriert werden. Solche Netzwerkvorrichtungen können mehrere Netzwerkschnittstellen aufweisen, die dazu eingerichtet oder gestaltet sind, unterschiedliche Typen von Netzwerkommunikationsprotokollen zu verwenden. Eine allgemeine Architektur für einige dieser Maschinen kann sich aus den hierin offenbarten Beschreibungen ergeben. Gemäß spezifischer Ausführungsformen können zumindest einige der Merkmale und/oder Funktionalitäten der verschiedenen hierin beschriebenen Ausführungsformen des virtuellen Assistenten auf einer oder mehreren Universal-Netzwerk-Host-Maschinen, wie einem Endbenutzercomputer-System, Computer, Netzwerk-Server oder Server-System, mobile Rechenvorrichtung (z. B. Persönlicher Digitaler Assistent, Mobiltelefon, Smartphone, Laptop, Tablet Computer oder Ähnliches), Unterhaltungselektronik-Vorrichtung, Musikplayer oder jegliche andere geeignete elektronische Vorrichtung, Router, Switch oder Ähnliches, oder jegliche Kombination davon implementiert sein. Bei zumindest einigen Ausführungsformen können zumindest einige der Merkmale und/oder Funktionalitäten der verschiedenen hierin offenbarten Ausführungsformen des virtuellen Assistenten auf einer oder mehreren virtualisierten Rechenumgebungen (z. B. Netzwerk-Rechen-Clouds oder Ähnliches) implementiert sein.

**[0063]** [Fig. 28](#), auf welche nun Bezug genommen wird, zeigt ein Blockdiagramm, welches eine Rechenvorrichtung **60** zeigt, die geeignet ist, zumindest einen Teil der hierin offenbarten Merkmale und/oder

Funktionalitäten des virtuellen Assistenten zu implementieren. Die Rechenvorrichtung **30** kann beispielsweise ein Endbenutzer-Computer-System, ein Netzwerk-Server oder -Server-System, eine mobile Rechenvorrichtung (z. B. Persönlicher Digitaler Assistent, Mobiltelefon, Smartphone, Laptop, Tablet-Computer oder Ähnliches), eine Unterhaltungselektronik-Vorrichtung, ein Musikplayer oder jegliche andere geeignete elektronische Vorrichtung oder jegliche(r) Kombination oder Teil hiervon sein. Die Rechenvorrichtung **60** kann angepasst sein, um mit anderen Rechenvorrichtungen, wie Clients und/oder Server, über ein Kommunikationsnetzwerk wie das Internet unter Verwendung bekannter Protokolle für solche Kommunikation, egal ob drahtlos oder drahtgebunden, zu kommunizieren.

**[0064]** Bei einer Ausführungsform enthält die Rechenvorrichtung **60** eine zentrale Prozessoreinheit (CPU) **62**, Schnittstellen **68**, und einen Bus **67** (wie einen Peripheral Component Interconnect (PCI) Bus). Wenn die CPU **62** unter der Steuerung von geeigneter Software oder Firmware agiert, kann sie verantwortlich sein, spezifische Funktionen, die mit den Funktionen einer spezifisch konfigurierten Rechenvorrichtung oder Maschine assoziiert sind, zu implementieren. Beispielsweise kann bei zumindest einer Ausführungsform ein Persönlicher Digitaler Assistent (PDA) oder Smartphone eines Benutzers eingerichtet oder gestaltet sein, um als ein virtuelles Assistent-System unter Verwendung von CPU **62**, Speicher **61**, **65** und Schnittstelle(n) **68** zu funktionieren. Bei zumindest einer Ausführungsform kann die CPU **62** veranlasst werden, einen oder mehrere der unterschiedlichen Typen von Funktionen und/oder Operationen des virtuellen Assistenten unter der Steuerung von Software-Modulen/Komponenten auszuführen, die beispielsweise ein Betriebssystem und jegliche geeignete Anwendungssoftware, Treiber und Ähnliches beinhalten.

**[0065]** Die CPU **62** kann einen oder mehrere Prozessor(en) **63** beinhalten, wie beispielsweise einen Prozessor aus der Motorola- oder Intel-Familie von Mikroprozessoren oder der MIPS-Familie von Mikroprozessoren. Bei einigen Ausführungsformen kann (können) Prozessor(en) **63** speziell gestaltete Hardware (z. B. anwendungsspezifische integrierte Schaltkreise (ASICs), elektrisch löschenbare programmierbare Nur-Lese-Speicher (EEPROMs), Feld-programmierbare Gatter-Anordnungen (FPGAs) und Ähnliches zur Steuerung von Operationen von Rechenvorrichtung **60** beinhalten. Bei einer spezifischen Ausführungsform stellt ein Speicher **61** (wie nichtflüchtiger Speicher mit wahlfreiem Zugriff (RAM) und/oder Nur-Lese-Speicher (ROM)) auch einen Teil von CPU **62** dar. Jedoch gibt es viele unterschiedliche Arten zum Verbinden von Speicher mit dem System. Speicherblock **61** kann für eine Vielfalt von Zwecken verwendet werden, wie beispielsweise Caching

und/oder Speichern von Daten, Programmier-Instruktionen und Ähnliches.

**[0066]** Wie hierin verwendet, ist der Begriff „Prozessor“ nicht nur beschränkt auf jene integrierten Schaltkreise, welche in der Technik als Prozessor bezeichnet werden, sondern bezieht sich umfassend auf einen Mikro-Controller, einen Mikro-Computer, einen programmierbaren Logik-Controller, einen anwendungsspezifischen integrierten Schaltkreis und jeglichen anderen programmierbaren Schaltkreis.

**[0067]** Bei einer Ausführungsform werden Schnittstellen **68** als Schnittstellen-Karten bereitgestellt (manchmal auch als „Netzkarten“ („Line Cards“) bezeichnet). Im Allgemeinen steuern sie das Senden und Empfangen von Datenpaketen über ein Rechenetzwerk und unterstützen manchmal andere Peripheriegeräte, die mit Rechenvorrichtung **60** verwendet werden. Unter den Schnittstellen, die bereitgestellt werden können, sind Ethernet-Schnittstellen, Frame Relay-Schnittstellen, Kabel-Schnittstellen, DSL-Schnittstellen, Token Ring-Schnittstellen und Ähnliches. Darüber hinaus können verschiedene Typen von Schnittstellen zur Verfügung gestellt werden, wie beispielsweise Universal Serial Bus (USB), Serial, Ethernet, Firewire, PCI, Parallel, Hochfrequenz (HF), Bluetooth™, Nahbereichskommunikation (z. B. unter Verwendung von Nahbereichsmagnetik), 802.11 (WiFi), Frame Relay, TCP/IP, ISDN, Fast-Ethernet-Schnittstellen, Gigabit-Ethernet-Schnittstellen, Asynchronous Transfer Mode(ATM)-Schnittstellen, High-Speed-Serial-Interface (HSSI)-Schnittstellen, Point of Sale-(POS)-Schnittstellen, Fiber Data Distributed Interfaces (FDDIs) und Ähnliches. Im Allgemeinen können solche Schnittstellen **68** Ports beinhalten, die für Kommunikation mit den geeigneten Medien geeignet sind. In einigen Fällen können sie auch einen unabhängigen Prozessor und, bei einigen Beispielen, flüchtigen und/oder nicht-flüchtigen Speicher (z. B. RAM) beinhalten.

**[0068]** Obwohl das in [Fig. 28](#) gezeigte System eine spezifische Architektur für eine Rechenvorrichtung **60** zur Implementierung der Techniken der hierin beschriebenen Erfahrung zeigt, ist es keinesfalls die einzige Vorrichtungs-Architektur, auf welcher zumindest ein Teil der Merkmale und Techniken, wie sie hierin beschrieben sind, implementiert werden kann. Beispielsweise können Architekturen, die einen oder eine beliebige Anzahl von Prozessoren **63** aufweisen, verwendet werden, und solche Prozessoren **63** können in einer einzigen Vorrichtung oder verteilt zwischen beliebig vielen Vorrichtungen vorhanden sein. Bei einer Ausführungsform verarbeitet ein einziger Prozessor **63** Kommunikationen wie auch Routing-Berechnungen. Bei verschiedenen Ausführungsformen können unterschiedliche Typen von Merkmalen und/oder Funktionalitäten des virtuellen Assistenten in einem virtuellen Assistent-System, welches ei-

ne Client-Vorrichtung (wie ein Persönlicher Digitaler Assistent oder ein Smartphone, auf welchem Client-Software läuft) und Server-System(e) (wie Server-Systeme, die nachfolgend genauer beschrieben werden) aufweist, implementiert sein.

**[0069]** Unabhängig von einer Netzwerkvorrichtungskonfiguration kann das System der vorliegenden Erfindung ein oder mehrere Speicher oder Speichermodule (wie beispielsweise Speicherblock **65**) verwenden, die zum Speichern von Daten, Programm-Instruktionen für die Universalnetzwerk-Operationen und/oder andere Information, die sich auf die Funktionalität der hierin beschriebenen virtuellen Assistenz-Techniken bezieht, eingerichtet sind. Die Programm-Instruktionen können beispielsweise die Operation eines Betriebssystems und/oder einer oder mehrerer Anwendung(en) steuern. Der Speicher oder die Speicher können auch eingerichtet sein, um Datenstrukturen, Schlüsselwort-Taxonomie-Information, Werbeinformation, Click- und Eindruck-Information eines Benutzers, und/oder andere hierin beschriebene spezifische nicht-programmbezogene Information zu speichern.

**[0070]** Da solche Informationen und Programm-Instruktionen verwendet werden können, um die hierin beschriebenen Systeme/Verfahren zu implementieren, können zumindest einige Ausführungsformen einer Netzwerkvorrichtung nicht-flüchtige, maschinenlesbare Speichermedien beinhalten, die beispielsweise dazu eingerichtet oder gestaltet sind, Programminstruktionen, Statusinformationen und Ähnliches zum Ausführen verschiedener hierin beschriebener Operationen zu speichern. Beispiele von solchen nicht-flüchtigen, maschinenlesbaren Speichermedien beinhalten, aber sind nicht beschränkt auf, magnetische Medien wie Festplatten, Floppy Disks und Magnetband; optische Medien wie CD-ROM-Disketten; magnetisch-optische Medien wie Floptical-Disketten, und Hardware-Vorrichtungen, die spezifisch zum Speichern und Ausführen von Programminstruktionen eingerichtet sind, wie Nur-Lese-Speichervorrichtungen (ROM), Flash-Speicher, Memristor-Speicher, Speicher mit wahlfreiem Zugriff (RAM) und Ähnliches. Beispiele von Programm-Instruktionen beinhalten sowohl Maschinen-Code, wie von einem Compiler erzeugt, als auch Dateien, die Code von höherer Ebene beinhalten, der von dem Computer unter Verwendung eines Interpreters ausgeführt werden kann.

**[0071]** Bei einer Ausführungsform ist das System der vorliegenden Erfindung auf einem unabhängigen Rechensystem implementiert. In [Fig. 29](#), auf die nun Bezug genommen wird, ist ein Blockdiagramm gezeigt, welches eine Architektur zur Implementierung zumindest eines Teils eines virtuellen Assistenten auf einem unabhängigen Rechensystem gemäß zumindest einer Ausführungsform darstellt. Rechenvor-

richtung **60** beinhaltet Prozessor(en) **63**, die Software zur Implementierung eines virtuellen Assistenten **1002** ausführen. Eingabevorrichtung **1206** kann jeglicher Typ sein, der geeignet ist, Benutzereingaben zu empfangen, einschließlich beispielsweise einer Tastatur, ein Berührungsbedienschirm, ein Mikrofon (z. B. für Spracheingabe), eine Maus, ein Berührungsbedienschirm, ein Trackball, ein Fünf-Wege-Schalter (Five-Way-Switch), ein Joystick und/oder jegliche Kombination hieraus. Ausgabevorrichtung **1207** kann ein Monitor, ein Lautsprecher, ein Drucker und/oder jegliche Kombination hieraus sein. Speicher **1210** kann ein Speicher mit wahlfreiem Zugriff sein, der eine Struktur und Architektur aufweist, wie sie in der Technik bekannt sind, zur Verwendung durch den/die Prozessor(en) **63** während des Ausführens von Software. Speichervorrichtung **1208** kann jegliche magnetische, optische und/oder elektrische Speichervorrichtung zum Speichern von Daten in digitaler Form sein; Beispiele beinhalten Flash-Speicher, magnetische Festplatte, CD-ROM und/oder Ähnliches.

**[0072]** Bei einer anderen Ausführungsform ist das System der vorliegenden Erfindung auf einem verteilten Rechennetzwerk, wie ein solches, das beliebig viele Clients oder Server aufweist, implementiert. In [Fig. 30](#), auf die nun Bezug genommen wird, ist ein Blockdiagramm dargestellt, welches eine Architektur zum Implementieren von zumindest einem Teil eines virtuellen Assistenten auf einem verteilten Rechenetzwerk gemäß zumindest einer Ausführungsform zeigt.

**[0073]** In der in [Fig. 30](#) gezeigten Anordnung werden beliebig viele Clients **1304** bereitgestellt; jeder Client **1304** kann Software ausführen, um Clientseitige Teile der vorliegenden Erfindung zu implementieren. Darüber hinaus können beliebig viele Server **1340** zum Verarbeiten von Anfragen, die von Clients **1304** empfangen werden, bereitgestellt werden. Clients **1304** und Server **1340** können miteinander über ein elektronisches Netzwerk **1361**, wie das Internet, kommunizieren. Das Netzwerk **1361** kann unter Verwendung jeglicher bekannter Netzwerkprotokolle, beinhaltend beispielsweise drahtgebundene und/oder drahtlose Protokolle, implementiert sein.

**[0074]** Darüber hinaus können bei einer Ausführungsform die Server **1340** bei Bedarf externe Dienste **1360** aufrufen, um zusätzliche Information zu erhalten, oder sich auf Speicherdaten, die frühere Interaktionen mit bestimmten Nutzern betreffen, beziehen. Die Kommunikation mit externen Diensten **1360** kann beispielsweise über das Netzwerk **1361** stattfinden. Bei verschiedenen Ausführungsformen können die externen Dienste **1360** Web-fähige Dienste und/oder Funktionalität, die sich auf die Hardwarevorrichtung selbst bezieht oder darauf installiert ist, beinhalten. Beispielsweise kann der Assistent **1002** bei einer Ausführungsform, bei welcher der Assistent **1002**

auf einem Smartphone oder auf einer anderen elektronischen Vorrichtung implementiert ist, Information erhalten, die in einer Kalenderanwendung ("App"), in Kontakten und/oder anderen Quellen gespeichert ist.

**[0075]** Bei verschiedenen Ausführungsformen kann der Assistent **1002** viele Merkmale und Operationen einer elektronischen Vorrichtung, auf der er installiert ist, steuern. Beispielsweise kann der Assistent **1002** die externen Dienste **1360**, die mit Funktionalität und Anwendungen auf einer Vorrichtung über APIs oder über andere Mittel interagieren, aufrufen, um Funktionen und Operationen auszuführen, die anderweitig unter Verwendung einer konventionellen Benutzerschnittstelle auf der Vorrichtung initiiert werden. Solche Funktionen und Operationen können beispielsweise beinhalten: Einstellen eines Alarms, Durchführen eines Telefonanrufs, Senden einer Textnachricht oder E-Mail-Nachricht, Hinzufügen eines Kalendereignisses und Ähnliches. Solche Funktionen und Operationen können als Add-on-Funktionen im Kontext eines interaktiven Dialogs zwischen einem Benutzer und Assistenten **1002** ausgeführt werden. Solche Funktionen und Operationen können von dem Benutzer im Kontext eines solchen Dialogs festgelegt werden, oder sie können automatisch basierend auf dem Kontext des Dialogs ausgeführt werden. Ein Fachmann wird erkennen, dass der Assistent **1002** dabei als ein Steuermechanismus zum Initiieren und Steuern verschiedener Operationen auf der elektronischen Vorrichtung verwendet werden kann, der als eine Alternative zu konventionellen Mechanismen, wie Tasten oder grafische Benutzerschnittstellen verwendet werden kann.

**[0076]** Beispielsweise kann der Benutzer dem Assistenten **1002** eine Eingabe wie "Ich muss morgen um 8 Uhr aufwachen" zur Verfügung stellen. Sobald der Assistent **1002** die Benutzerabsicht bestimmt hat, unter Verwendung der hierin beschriebenen Techniken, kann der Assistent **1002** externe Dienste **1340** aufrufen, um mit einer Weckerfunktion oder Anwendung auf der Vorrichtung zu interagieren. Der Assistent **1002** stellt den Alarm im Auftrag des Benutzers ein. Auf diese Weise kann der Benutzer den Assistenten **1002** als Ersatz für konventionelle Mechanismen zum Einstellen des Alarms oder Ausführen anderer Funktionen auf der Vorrichtung verwenden. Wenn die Anfragen des Benutzers mehrdeutig sind oder weiterer Klarstellung bedürfen, kann der Assistent **1002** die verschiedenen hierin beschriebenen Techniken nutzen, einschließlich aktiver Ermittlung, Paraphrasieren, Vorschläge und Ähnliches und einschließlich Erhalten von Kontextinformation, so dass die korrekten Dienste **1340** aufgerufen werden und die beabsichtigte Aktion ausgeführt wird. Bei einer Ausführungsform kann der Assistent **1002** den Benutzer zur Bestätigung auffordern und/oder zusätzliche Kontextinformation aus jeglicher geeigneten Quelle anfragen, ehe dieser einen Dienst **1340** zum Ausführen einer Funk-

tion aufruft. Bei einer Ausführungsform kann ein Benutzer die Fähigkeit des Assistenten **1002**, bestimmte Dienste **1340** aufzurufen, selektiv ausschalten, oder kann das gesamte Aufrufen von Diensten ausschalten, wenn gewünscht.

**[0077]** Das System der vorliegenden Erfindung kann mit jeglichem aus einer Reihe unterschiedlicher Typen von Clients **1304** und Operationsmodi implementiert werden. In [Fig. 31](#), auf die nun Bezug genommen wird, ist ein Blockdiagramm dargestellt, welches eine Systemarchitektur zeigt, die unterschiedliche Typen von Clients **1304** und Operationsmodi darstellt. Ein Fachmann wird erkennen, dass die unterschiedlichen Typen von Clients **1304** und Operationsmodi, wie sie in [Fig. 31](#) gezeigt sind, lediglich beispielhaft sind, und dass das System der vorliegenden Erfindung unter Verwendung anderer als der dargestellten Clients **1304** und/oder Operationsmodi implementiert werden kann. Darüber hinaus kann das System jegliche oder alle dieser Clients **1304** und/oder Operationsmodi beinhalten, allein oder in jeglicher Kombination. Dargestellte Beispiele beinhalten:

- Computervorrichtungen mit Eingabe/Ausgabevorrichtungen und/oder Sensoren **1402**. Eine Client-Komponente kann auf jeder solchen Computervorrichtungen **1402** betrieben werden. Zumindest eine Ausführungsform kann unter Verwendung eines Web-Browsers **1304A** oder einer anderen Softwareanwendung zum Ermöglichen von Kommunikation mit Servern **1340** über Netzwerk **1361** implementiert sein. Eingabe- und Ausgabe-Kanäle können von jeglichem Typ sein, einschließlich beispielsweise visueller und/oder akustischer Kanäle. Beispielsweise kann das System der Erfindung bei einer Ausführungsform unter Verwendung von sprachbasierten Kommunikationsverfahren implementiert sein, wodurch eine Ausführungsform des Assistenten für Blinde ermöglicht wird, dessen Äquivalent eines Web-Browsers durch Sprache gesteuert wird und Sprache als Ausgabe verwendet.
- Mobile Vorrichtungen mit E/A und Sensoren **1406**, für die der Client als eine Anwendung auf der mobilen Vorrichtung **1304B** implementiert sein kann. Dies beinhaltet, aber ist nicht beschränkt auf, Mobiltelefone, Smartphones, persönliche digitale Assistenten, Tablet-Vorrichtungen, vernetzte Spielekonsolen und Ähnliches.
- Verbrauchergeräte mit E/A und Sensoren **1410**, für die der Client als eine eingebettete Anwendung auf dem Gerät **1304C** implementiert sein kann.
- Automobile und andere Fahrzeuge mit Armaturenbrettschnittstellen und Sensoren **1414**, für die der Client als eine eingebettete Systemanwendung **1304D** implementiert sein kann. Dies beinhaltet, aber ist nicht beschränkt auf Autonavigationssysteme, Sprachsteuersysteme, Unterhaltungssysteme im Fahrzeuginnenraum und Ähnliches.

- Vernetzte Rechenvorrichtungen wie Router **1418** oder jegliche andere Vorrichtung, die auf einem Netzwerk liegt, oder eine Schnittstelle zu einem Netzwerk aufweist, bei welcher der Client als eine auf der Vorrichtung liegende Anwendung **1304E** implementiert sein kann.
- E-Mail-Clients **1424**, für die eine Ausführungsform des Assistenten über einen E-Mail-Modalitäten-Server **1426** verbunden ist. Der E-Mail-Modalitäten-Server **1426** agiert als eine Kommunikationsbrücke, beispielsweise nimmt er Eingaben von dem Benutzer als E-Mail-Nachrichten, die zu dem Assistenten gesendet wurden, entgegen, und versendet Ausgaben von dem Assistenten zum Benutzer als Antworten.
- Clients für Sofortnachrichtendienste **1428**, für die eine Ausführungsform des Assistenten über einen Nachrichten-Modalitäten-Server **1430** verbunden ist. Der Nachrichten-Modalitäten-Server **1430** agiert als Kommunikationsbrücke, nimmt Eingaben von dem Benutzer, die als Nachrichten zu dem Assistenten gesendet werden entgegen, und sendet in Beantwortung Ausgaben von dem Assistenten zum Benutzer als Nachrichten.
- Sprachtelefone **1432**, bei welchen eine Ausführungsform des Assistenten über einen Internet-Protokoll-Telefonie-(VoIP)-Modalitäten-Server **1430** verbunden ist. Der VoIP-Modalitäten-Server **1430** agiert als Kommunikationsbrücke, der Eingaben vom Benutzer als Sprache, die zu dem Assistenten gesprochen wurde, entgegennimmt, und in Beantwortung Ausgaben vom Assistenten zum Benutzer sendet, beispielsweise als synthetisierte Sprache.

**[0078]** Für Nachrichten-Plattformen, einschließlich, aber nicht beschränkt auf, E-Mail, Sofortnachrichtendienste, Diskussionsforen, Gruppen-Chat-Sitzungen, Live-Hilfe oder Kunden-Support-Sitzungen und Ähnliches, kann der Assistent **1002** als ein Teilnehmer in den Unterhaltungen agieren. Der Assistent **1002** kann die Unterhaltung beobachten und Einzelnen oder der Gruppe unter Verwendung einer oder mehrerer Techniken oder Methoden, wie hierin für Eins-zu-Eins-Interaktionen beschrieben, erwideren.

**[0079]** Bei verschiedenen Ausführungsformen kann die Funktionalität zum Implementieren der Techniken der vorliegenden Erfindung zwischen einer beliebigen Anzahl von Client- und/oder Server-Komponenten verteilt sein. Beispielsweise können verschiedene Software-Module zum Ausführen verschiedener Funktionen in Verbindung mit der vorliegenden Erfindung implementiert sein, und solche Module können unterschiedlich implementiert sein, um auf Server- und/oder Client-Komponenten ausgeführt zu werden. Weitere Einzelheiten einer solchen Anordnung sind in der verwandten US-Utility Application Anmeldenr. 12/987,982, betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Akten-

zeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011, enthalten, wobei die gesamte Offenbarung dieser Anmeldung hierin mittels Bezugnahme eingebunden ist.

**[0080]** In dem Beispiel der [Fig. 32](#) sind Eingabeermitlungsfunktionalität und Ausgabeverarbeitungsfunktionalität zwischen Client **1304** und Server **1340** verteilt, wobei ein Client-Teil der Eingabeermitzung **2794a** und ein Client-Teil der Ausgabeverarbeitung **2792a** auf dem Client **1304** vorliegen, und ein Server-Teil von Eingabeermitzung **2794b** und ein Server-Teil von Ausgabeverarbeitung **2792b** auf Server **1340** vorliegen. Die folgenden Komponenten befinden sich auf dem Server **1340**:

- komplettes Vokabular **2758b**;
- komplette Bibliothek von Sprachmuster-Erkennern **2760b**;
- Masterversion von persönlichem Kurzzeitspeicher **2752b**;
- Masterversion von persönlichem Langzeitspeicher **2754b**.

**[0081]** In einer Ausführungsform hält der Client **1304** Untermengen und/oder Teile dieser Komponenten lokal vor, um die Reaktionsfähigkeit zu verbessern und die Abhängigkeit von Netzwerkkommunikationen zu reduzieren. Solche Untermengen und/oder Teile können gemäß wohlbekannter Cache-Verwaltungstechniken unterhalten und aktualisiert werden. Solche Untermengen und/oder Teile beinhalten beispielsweise:

- Untermenge von Vokabular **2758a**;
- Untermenge der Bibliothek von Sprachmuster-Erkennern **2760a**;
- Cache des persönlichen Kurzzeitspeichers **2752a**;
- Cache des persönlichen Langzeitspeichers **2754a**.

**[0082]** Zusätzliche Komponenten können als Teil des Servers **1340** implementiert sein, die beispielsweise beinhalten:

- Sprach-Interpreter **2770**;
- Dialog-Fluss-Prozessor **2780**;
- Ausgabeprozessor **2790**;
- Bereichs-Entitäten-Datenbanken **2772**;
- Task-Fluss-Modelle **2786**;
- Dienste-Orchestrierung **2782**;
- Dienste-Fähigkeitsmodelle **2788**.

**[0083]** Jede dieser Komponenten wird nachfolgend genauer beschrieben. Der Server **1340** erhält bei Bedarf zusätzliche Information durch Interagieren mit externen Diensten **1360**.

## Konzeptionelle Architektur

**[0084]** In [Fig. 27](#), auf die nun Bezug genommen wird, ist ein vereinfachtes Blockdiagramm eines spezifischen Ausführungsbeispiels eines virtuellen As-

sistenten **1002** dargestellt. Wie mit mehr Einzelheiten in den voranstehend einbezogenen verwandten US-Utility Applications dargestellt wird, können unterschiedliche Ausführungsformen des virtuellen Assistenten **1002** eingerichtet, gestaltet und/oder betreibbar sein, um mehrere unterschiedliche Typen von Operationen, Funktionalitäten und/oder Merkmalen, die sich allgemein auf die Technologie des virtuellen Assistenten beziehen, zur Verfügung zu stellen. Weiterhin können, wie hierin genauer beschrieben, viele der unterschiedlichen Operationen, Funktionalitäten und/oder Merkmale des virtuellen Assistenten **1002**, die hierin offenbart sind, unterschiedliche Typen von Vorteilen und/oder Nutzen für unterschiedliche Entitäten, die mit dem virtuellen Assistenten **1002** interagieren, ermöglichen oder zur Verfügung stellen. Die in [Fig. 27](#) gezeigte Ausführungsform kann implementiert werden unter Verwendung jeglicher der oben beschriebenen Hardware-Architekturen oder unter Verwendung eines unterschiedlichen Typs von Hardware-Architektur.

**[0085]** Beispielsweise kann der virtuelle Assistent **1002** gemäß unterschiedlicher Ausführungsformen eingerichtet, gestaltet und/oder betreibbar sein, um mehrere unterschiedliche Typen von Operationen, Funktionalitäten und/oder Merkmalen, wie beispielsweise eine oder mehr der Folgenden (oder Kombinationen hieraus) zur Verfügung zu stellen:

- Automatisieren der Anwendung von Daten und Diensten, die über das Internet verfügbar sind, um Produkte und Dienste zu entdecken, aufzufinden, zwischen ihnen zu wählen, zu kaufen, zu stornieren, oder zu bestellen. Zusätzlich zur Automatisierung des Prozesses zur Verwendung dieser Daten und Dienste kann der virtuelle Assistent **1002** die kombinierte Verwendung von mehreren Quellen von Daten und Diensten auf einmal ermöglichen. Beispielsweise kann er Informationen über Produkte aus mehreren Rezessionsseiten kombinieren, Preise und Verfügbarkeit von mehreren Distributoren prüfen und ihre Beschränkungen hinsichtlich Ort und Zeit prüfen, und einem Benutzer helfen, eine personalisierte Lösung zu seinem Problem zu finden.
- Automatisieren der Verwendung von Daten und Diensten, die über das Internet verfügbar sind, zum Entdecken, Erforschen, Auswählen unter, Stornieren und anderweitig Lernen über Dinge, die getan werden können (einschließlich, aber nicht beschränkt auf, Filme, Events, Darbietungen, Ausstellungen, Shows und Attraktionen); Orte, die besucht werden können (einschließlich, aber nicht beschränkt auf, Reiseziele, Hotels und andere Bleiben, Wahrzeichen und andere Orte von Interesse und Ähnliches); Orte zum Essen oder Trinken (wie Restaurants und Bars), Zeiten und Orte, um andere zu treffen, und jegliche andere Quelle von Unterhaltung oder sozialer Interaktion, die im Internet gefunden werden kann.

- Ermöglichen des Betriebs von Anwendungen und Diensten über natürlichen Sprachdialog, die anderweitig von dedizierten Anwendungen mit grafischen Benutzerschnittstellen bereitgestellt wird, einschließlich Suche (einschließlich ortsbezogener Suche); Navigation (Karten und Richtungen); Datenbankabfragen (wie Auffinden von Geschäften oder Menschen mittels Namen oder anderer Eigenschaften); Erhalten von Wetterbedingungen und Vorhersagen, Prüfen des Preises von Marktobjekten oder Status von finanziellen Transaktionen; Überwachen von Verkehr oder des Status von Flügen; Zugreifen auf und Aktualisieren von Kalendern und Zeitplänen; Verwalten von Remindern, Alarmsignalen, Tasks und Projekten; Kommunizieren über E-Mail oder andere Nachrichtenplattformen; und Betreiben von Vorrichtungen von lokal oder von entfernt (z. B. Wählen eines Telefons, Steuern von Licht und Temperatur, Steuern von Heimsicherheitsvorrichtungen, Spielen von Musik oder Video und Ähnliches). Bei einer Ausführungsform kann der virtuelle Assistent **1002** verwendet werden, um viele Funktionen und Apps, die auf der Vorrichtung verfügbar sind, zu initiieren, zu betreiben und zu steuern.

- Anbieten persönlicher Empfehlungen für Aktivitäten, Produkte, Dienste, Quellen von Unterhaltung, Zeitmanagement oder jegliche andere Art von Empfehlungsdienst, die von einem interaktiven Dialog in natürlicher Sprache und automatisierten Zugriff auf Daten und Dienste profitiert.

**[0086]** Gemäß unterschiedlicher Ausführungsformen kann zumindest ein Teil der verschiedenen Typen von Funktionen, Operationen, Aktionen und/oder anderen Merkmalen, die von dem virtuellen Assistenten **1002** zur Verfügung gestellt werden, auf einem oder mehreren Client-System(en), auf einem oder mehreren Server-System(en) und/oder Kombinationen hieraus implementiert sein.

**[0087]** Gemäß unterschiedlicher Ausführungsformen kann zumindest ein Teil der verschiedenen Typen von Funktionen, Operationen, Aktionen und/oder anderer Merkmale, die von dem virtuellen Assistenten **1002** zur Verfügung gestellt werden, beim Interpretieren und Operationalisieren von Benutzereingabe Kontextinformation verwenden, wie hierin genauer beschrieben.

**[0088]** Beispielsweise kann der virtuelle Assistent **1002** bei zumindest einer Ausführungsform betreibbar sein, um mehrere unterschiedliche Typen von Daten und/oder andere Typen von Information beim Ausführen spezifischer Tasks und/oder Operationen zu nutzen und/oder zu generieren. Dies kann beispielsweise Eingabedaten/Information und/oder Ausgabedaten/Information beinhalten. Beispielsweise kann der virtuelle Assistent **1002** bei zumindest

einer Ausführungsform zum Zugreifen auf, Verarbeiten und/oder anderweitig Nutzen von Informationen aus einer oder mehreren unterschiedlichen Typen von Quellen betreibbar sein, wie beispielsweise einem oder mehrerer lokaler und/oder entfernter Speicher, Vorrichtungen und/oder Systemen. Zusätzlich kann der virtuelle Assistent **1002** bei zumindest einer Ausführungsform betrieben werden, um eine oder mehrere unterschiedliche Typen von Ausgabedaten/Information zu generieren, die beispielsweise in einem Speicher einer oder mehrerer lokaler und/oder entfernter Vorrichtungen und/oder Systeme gespeichert sein kann.

**[0089]** Beispiele von unterschiedlichen Typen von Eingabedaten/Information, auf die der virtuelle Assistent **1002** zugreifen und/oder diese nutzen kann, können beinhalten, aber sind nicht beschränkt auf, eine oder mehr der Folgenden (oder Kombinationen hieraus):

- Spracheingabe: von mobilen Vorrichtungen wie Mobiltelefonen und Tablets, Computer mit Mikrofonen, Bluetooth-Headsets, Sprachsteuersystemen in Automobilen, System über das Telefon (over the telephone system), Aufnahmen auf Antwort-Diensten, Audio-Voicemail bei integrierten Nachrichtendiensten, Verbraucheranwendungen mit Spracheingabe, wie Radiowecker, Telefonstation, Heim-Unterhaltungs-Steuerungssysteme und Spielekonsolen.
- Texteingabe von Tastaturen auf Computern oder mobilen Vorrichtungen, Tastenfelder (Keypads) auf Fernsteuerungen oder andere Vorrichtungen von Unterhaltungselektronik, E-Mail-Nachrichten, die dem Assistenten gesendet werden, Sofortnachrichten oder ähnliche Kurznachrichten, die dem Assistenten gesendet werden, Text, der von Spielern in Mehrbenutzer-Spieleumgebungen empfangen wird, und Text, der in Nachrichten-Feeds strömt.
- Ortsangaben, die aus Sensoren oder standortbasierten Systemen stammen. Beispiele beinhalten Global Positioning System (GPS) und Assisted-GPS (A-GPS) auf mobilen Vorrichtungen. Bei einer Ausführungsform werden Ortsangaben mit ausdrücklicher Benutzereingabe kombiniert. Bei einer Ausführungsform ist das System der vorliegenden Erfindung in der Lage, zu erkennen, wenn ein Benutzer zu Hause ist, basierend auf bekannter Adressinformation und gegenwärtiger Standortbestimmung. Auf diese Weise können bestimmte Schlussfolgerungen gemacht werden im Hinblick auf den Typ von Information, die den Benutzer interessieren könnte, wenn er zu Hause ist, im Gegensatz dazu, wenn er außer Haus ist, wie auch im Hinblick auf den Typ von Diensten und Aktionen, die im Namen des Benutzers aufgerufen werden sollen, abhängig davon, ob er oder sie zu Hause ist.

- Zeitangaben von Uhren auf Client-Vorrichtungen. Dies kann beispielsweise die Zeit von Telefonen oder anderen Client-Vorrichtungen, welche die lokale Zeit und Zeitzone angeben, beinhalten. Darüber hinaus kann Zeit im Zusammenhang mit Benutzeranfragen verwendet werden, um beispielsweise Phrasen wie „in einer Stunde“ und „heute Abend“ zu interpretieren.

- Kompass, Beschleunigungsmesser, Gyroskop und/oder Reisegeschwindigkeitsdaten, wie auch andere Sensordaten von mobilen oder Handheld-Vorrichtungen oder eingebetteten Systemen wie Automobilsteuerungssystemen. Dies kann auch Vorrichtungspositionsdaten von Fernsteuerungen für Geräte und Spielekonsolen beinhalten.

- Klicken und Menüauswahl und andere Ereignisse von einer grafischen Benutzeroberfläche (GUI) auf jeglicher Vorrichtung, die eine GUI aufweist. Weitere Beispiele beinhalten Berührungen bei einem Berührungsbildschirm.

- Ereignisse von Sensoren und anderen datengetriebenen Auslösern, wie Wecker, Kalender-Alarne, Preisänderungs-Auslöser, Standort-Auslöser, Push-Benachrichtigung auf einer Vorrichtung von Servern und Ähnliches.

**[0090]** Die Eingabe bei den hierin beschriebenen Ausführungsformen beinhaltet auch den Kontext der Historie der Benutzerinteraktion, einschließlich Dialog- und Anforderungshistorie.

**[0091]** Wie in den in verwandten US-Utility Applications, auf welche oben querverwiesen wurde, beschrieben ist, können viele unterschiedliche Typen von Ausgabedaten/Information von dem virtuellen Assistenten **1002** generiert werden. Diese können beinhalten, aber sind nicht beschränkt auf, eine oder mehr der Folgenden (oder Kombinationen hieraus):

- Textausgabe, die direkt einer Ausgabevorrichtung und/oder zu der Benutzeroberfläche einer Vorrichtung gesendet wird;
- Text und Grafiken, die einem Benutzer über E-Mail gesendet werden;
- Text und Grafiken, die einem Benutzer über einen Nachrichtendienst gesendet werden;
- Sprachausgabe, die eine oder mehr der Folgenden beinhalten kann (oder Kombinationen hieraus):
  - synthetisierte Sprache;
  - abgetastete Sprache;
  - aufgenommene Nachrichten;
  - grafisches Layout von Information mit Fotos, Rich Text, Videos, Sounds und Hyperlinks (beispielsweise der Inhalt, der in einem Web-Browser wiedergegeben wird);
  - Aktuatorausgabe zum Steuern von physikalischen Aktionen auf einer Vorrichtung, wie Verlassen, dass sie ein- oder ausgeschaltet wird, einen Sound produziert, Farbe ändert, vibriert, ein Licht steuert und Ähnliches;

- Aufrufen von anderen Anwendungen auf einer Vorrichtung, wie Aufrufen einer Karten-Anwendung, Sprachwahl eines Telefons, Versenden einer E-Mail oder Sofortnachricht, Abspielen von Medien, Vornehmen von Einträgen in Kalendern, Taskverwaltungen und Notizanwendungen und anderen Anwendungen;
- Aktuatorausgabe zum Steuern von physikalischen Aktionen von Vorrichtungen, verbunden mit oder gesteuert von einer Vorrichtung, wie Betreiben einer entfernten Kamera, Steuern eines Rollstuhls, Spielen von Musik auf entfernten Lautsprechern, Spielen von Videos auf entfernten Anzeigen und Ähnliches.

**[0092]** Es soll erkannt werden, dass der virtuelle Assistent **1002** aus [Fig. 27](#) lediglich ein Beispiel aus einem breiten Bereich von Ausführungsformen eines virtuellen Assistenz-Systems ist, welche implementiert werden können. Andere Ausführungsformen des virtuellen Assistenz-Systems (nicht gezeigt) können zusätzliche, weniger und/oder unterschiedliche Komponenten/Merkmale als jene, die zum Beispiel in der beispielhaften Ausführungsform des virtuellen Assistenz-Systems von [Fig. 27](#) dargestellt sind, beinhalten.

**[0093]** Der virtuelle Assistent **1002** kann eine Vielzahl von unterschiedlichen Typen von Komponenten, Vorrichtungen, Modulen, Prozessen, Systemen und Ähnliches beinhalten, die beispielsweise durch die Verwendung von Hardware und/oder Kombinationen von Hardware und Software implementiert und/oder instanziert sein können. Beispielsweise kann der Assistent **1002**, wie dargestellt in der beispielhaften Ausführungsform von [Fig. 27](#), eine oder mehr der folgenden Typen von Systemen, Komponenten, Vorrichtungen, Prozessen und Ähnliches (oder Kombinationen hieraus) beinhalten:

- eine oder mehr aktive Ontologien **1050**;
- aktive Eingabeermittlungskomponente(n) **2794** (kann/können Client-Teil **2794a** und Server-Teil **2794b** beinhalten);
- persönliche Kurzzeitspeicherkomponente(n) **2752** (kann/können Master-Version **2752b** und Cache **2752a** beinhalten);
- persönliche Langzeitspeicherkomponente(n) **2754** (kann/können Master-Version **2754b** und Cache **2754a** beinhalten; kann/können beispielsweise persönliche Datenbanken **1058**, Anwendungspräferenzen und Verwendungshistorie **1072** und Ähnliches beinhalten);
- Bereichs-Modell-Komponente(n) **2756**;
- Vokabular-Komponente(n) **2758** (kann/können vollständiges Vokabular **2758b** und Untermenge **2758a** beinhalten);
- Komponente(n) mit Erkenner(n) von Sprachmustern **2760** (kann/können vollständige Bibliothek **2760b** und Untermenge **2760a** beinhalten);
- Komponente(n) mit Sprachinterpretierer **2770**;

- Bereichs-Entitäts-Datenbanke(n) **2772**;
- Dialog-Fluss-Prozessor-Komponente(n) **2780**;
- Komponente(n) zur Dienste-Orchestrierung **2782**;
- Dienste-Komponente(n) **2784**;
- Komponente(n) mit Task-Fluss-Modellen **2786**;
- Komponente(n) mit Dialog-Fluss-Modellen **2787**;
- Komponente(n) mit Dienst-Modellen **2788**;
- Ausgabeprozessorkomponente(n) **2790**.

**[0094]** Bei bestimmten Client/Server-basierten Ausführungsformen können einige oder alle dieser Komponenten zwischen Client **1304** und Server **1340** verteilt sein.

**[0095]** Bei einer Ausführungsform erhält der virtuelle Assistent **1002** Benutzereingabe **2704** mittels jeglicher geeigneter Eingabemodalität, einschließlich beispielsweise Eingabe mittels Berührungsbildschirm, Tastatur-Eingabe, gesprochene Eingabe und/oder jegliche Kombination hieraus. Bei einer Ausführungsform erhält der Assistent **1002** auch Kontext-Information **1000**, die Ereigniskontext **2706** und/oder jegliche andere Typen von Kontext wie hierin genauer beschrieben beinhalten kann.

**[0096]** Beim Verarbeiten der Benutzereingabe **2704** und der Kontext-Information **1000** gemäß den hierin beschriebenen Techniken generiert der virtuelle Assistent **1002** Ausgabe **2708** zur Darstellung für den Benutzer. Die Ausgabe **2708** kann gemäß jeglicher geeigneter Ausgabemodalität generiert werden, die gegebenenfalls durch den Kontext **1000** wie auch durch anderen Faktoren geprägt sein kann. Beispiele von Ausgabemodalitäten beinhalten visuelle Ausgabe, wie sie auf einem Bildschirm dargestellt wird, akut Ausgabe (die gesprochene Ausgabe und/oder Pieptöne und andere Sounds beinhalten kann), haptische Ausgabe (wie Vibration) und/oder jegliche Kombination hieraus.

**[0097]** Zusätzliche Einzelheiten betreffend die Operation der verschiedenen Komponenten, die in [Fig. 27](#) gezeigt sind, sind in der verwandten US-Utility Application Anmeldenr. 12/987,982 betreffend „Intelligent Automated Assistant“ („Intelligenter automatisierter Assistent“), Aktenzeichen des Vertreters P10575US1, eingereicht am 10. Januar 2011, bereitgestellt, wobei die gesamte Offenbarung dieser Anmeldung hierin mittels Bezugnahme eingebunden ist.

## Kontext

**[0098]** Wie oben beschrieben erfasst der virtuelle Assistent **1002** bei einer Ausführungsform eine Vielfalt von Kontextinformation, um Informationsverarbeitungsfunktionen auszuführen, und wendet diese Vielfalt an. Die folgende Beschreibung legt dar:

- einen Bereich von Quellen von Kontextinformation zur Benutzung des virtuellen Assistenten **1002**;
- Techniken zur Darstellung, zum Organisieren und Suchen von Kontextinformation;
- Verfahren mittels welcher Kontextinformation den Betrieb von mehreren Funktionen des virtuellen Assistenten unterstützen kann; und
- Verfahren zum effizienten Erfassen, Zugreifen auf und Anwenden von Kontextinformation in einem verteilten System.

**[0099]** Ein Fachmann wird erkennen, dass die nachfolgende Beschreibung von Quellen, Techniken und Verfahren zur Verwendung von Kontextinformation lediglich beispielhaft ist, und dass andere Quellen, Techniken und Verfahren verwendet werden können, ohne dass von den wesentlichen Charakteristiken der vorliegenden Erfindung abgewichen wird.

#### Quellen von Kontext

**[0100]** Während Phasen von Informationsverarbeitung, die von dem virtuellen Assistenten **1002** ausgeführt werden, können mehrere unterschiedliche Arten von Kontext verwendet werden, um mögliche Interpretationen einer Benutzereingabe zu reduzieren. Beispiele beinhalten Anwendungskontext, Kontext persönlicher Daten und vorangegangene Dialoghistorie. Ein Fachmann wird erkennen, dass auch andere Quellen von Kontext verfügbar sein können.

**[0101]** In [Fig. 1](#), auf die nun Bezug genommen wird, ist ein Blockdiagramm dargestellt, welches den virtuellen Assistenten **1002** und einige Beispiele von Quellen von Kontext zeigt, die seine Operation gemäß einer Ausführungsform beeinflussen können. Der virtuelle Assistent **1002** nimmt Benutzereingabe **2704**, wie gesprochene oder getippte Sprache entgegen, verarbeitet die Eingabe und generiert Ausgabe **2708** für den Benutzer und/oder führt **2710** Aktionen im Namen des Benutzers aus. Es soll erkannt werden, dass der wie in [Fig. 1](#) dargestellte virtuelle Assistent **1002** lediglich ein Beispiel aus einem breiten Bereich von implementierbaren Ausführungsformen eines virtuellen Assistenz-Systems darstellt. Andere Ausführungsformen von virtuellen Assistenz-Systemen (nicht gezeigt) können zusätzliche, weniger und/oder unterschiedliche Komponenten/Merkmale enthalten, als jene, die beispielsweise in dem beispielhaften virtuellen Assistenten **1002**, wie in [Fig. 1](#) gezeigt, dargestellt sind.

**[0102]** Wie hierin im Einzelnen beschrieben kann der virtuelle Assistent **1002** auf jegliche aus einer Reihe unterschiedlicher Quellen von Wissen und Daten zurückgreifen, wie Wörterbücher, Bereichs-Modelle und/oder Task-Modelle. Aus der Perspektive der vorliegenden Erfindung sind solche Quellen, die als Hintergrundquellen hierin bezeichnet sind, intern für den Assistenten **1002**. Zusätzlich zur Benutzereingabe

**2704** und den Hintergrundquellen kann der Assistent **1002** auch auf Informationen aus mehreren Quellen von Kontext zurückgreifen, einschließlich beispielsweise Vorrichtungssensordaten **1056**, Anwendungspräferenzen und Verwendungshistorie **1072**, Dialoghistorie und Speicher des Assistenten **1052**, persönliche Datenbanken **1058**, persönliche akustische Kontextdaten **1080**, gegenwärtiger Anwendungskontext **1060** und Ereigniskontext **2706**. Diese werden hierin im Einzelnen beschrieben.

#### Anwendungskontext **1060**

**[0103]** Anwendungskontext **1060** bezieht sich auf den Zustand der Anwendung oder ähnlicher Software, in welcher der Benutzer etwas macht. Beispielsweise könnte der Benutzer eine Textnachrichtenanwendung benutzen, um mit einer bestimmten Person zu chatten. Der virtuelle Assistent **1002** muss nicht spezifisch für oder Teil der Benutzerschnittstelle der Textnachrichtenanwendung sein. Vielmehr kann der virtuelle Assistent **1002** Kontext aus jeglicher Anzahl von Anwendungen erhalten, wobei jede Anwendung ihren Kontext einbringt, um den virtuellen Assistenten **1002** zu informieren.

**[0104]** Falls der Benutzer gegenwärtig eine Anwendung benutzt, wenn der virtuelle Assistent **1002** aufgerufen wird, kann der Zustand dieser Anwendung hilfreiche Kontextinformation zur Verfügung stellen. Wird beispielsweise der virtuelle Assistent **1002** von innerhalb einer E-Mail-Anwendung aufgerufen, kann die Kontextinformation Absenderangaben, Empfängerangaben, Sendedatum und/oder Sendezzeit, Betreff, Daten, die aus dem E-Mail-Inhalt extrahiert wurden, Mailbox- oder Verzeichnisname und Ähnliches beinhalten.

**[0105]** In [Fig. 11](#) bis [Fig. 13](#), auf die nun Bezug genommen wird, wird eine Menge von Bildschirmkopien gemäß einer Ausführungsform dargestellt, die Beispiele der Verwendung von Anwendungskontext in einem Textnachrichten-Bereich zeigen, um einen Bezug für ein Pronomen abzuleiten. [Fig. 11](#) zeigt Bildschirm **1150**, welcher angezeigt werden kann, während der Benutzer in einer Textnachrichtenanwendung ist. [Fig. 12](#) zeigt den Bildschirm **1250**, nachdem der virtuelle Assistent **1002** im Kontext der Textnachrichtenanwendung aktiviert wurde. In diesem Beispiel präsentiert der virtuelle Assistent **1002** dem Benutzer Aufforderung **1251**. Bei einer Ausführungsform kann der Benutzer Spracheingabe durch Antippen von Mikrofon-Icon **1252** bereitstellen. Bei einer anderen Ausführungsform ist der Assistent **1002** in der Lage, sprachliche Eingabe zu jeder Zeit zu akzeptieren und erfordert nicht, dass der Benutzer Mikrofon-Icon **1252** antippt, bevor er Eingabe bereitstellt; folglich kann Icon **1252** eine Erinnerung sein, dass der Assistent **1002** auf gesprochene Eingabe wartet.

[0106] In [Fig. 13](#) wurde der Benutzer durch einen Dialog mit dem virtuellen Assistenten **1002** aktiv, wie auf Bildschirm **1253** gezeigt. Die Spracheingabe des Benutzers „Rufe ihn an“ wurde zurückgegeben und der virtuelle Assistent **1002** antwortet, dass er eine bestimmte Person auf einer bestimmten Telefonnummer anrufen wird. Um die mehrdeutige Eingabe des Benutzers zu interpretieren, verwendet der virtuelle Assistent **1002** eine Kombination aus mehreren Quellen von Kontext, um einen Bezug für ein Pronomen, wie hierin genauer dargelegt, abzuleiten.

[0107] In [Fig. 17](#) bis [Fig. 20](#), auf die nun Bezug genommen wird, wird ein anderes Beispiel der Verwendung von gegenwärtigem Anwendungskontext dargestellt, um gemäß einer Ausführungsform einen Befehl zu interpretieren und zu operationalisieren.

[0108] In [Fig. 17](#) wird dem Benutzer sein oder ihr E-Mail-Eingang **1750** dargestellt, und er/sie wählt eine bestimmte E-Mail-Nachricht **1751** zur Ansicht aus. [Fig. 18](#) zeigt die E-Mail-Nachricht **1751**, nachdem sie zur Ansicht ausgewählt wurde. In diesem Beispiel enthält die E-Mail-Nachricht **1751** ein Bild.

[0109] In [Fig. 19](#) hat der Benutzer den virtuellen Assistenten **1002** von innerhalb der E-Mail-Anwendung aktiviert, während der Ansicht der E-Mail-Nachricht **1751**. Bei einer Ausführungsform bewegt sich die Anzeige der E-Mail-Nachricht **1751** nach oben, um auf dem Bildschirm für Aufforderung **150** des virtuellen Assistenten **1002** Platz zu machen. Diese Anzeige untermauert die Vorstellung, dass der virtuelle Assistent **1002** Hilfestellung in dem Kontext der gegenwärtig betrachteten E-Mail-Nachricht **1751** anbietet. Dementsprechend wird die Eingabe des Benutzers für den virtuellen Assistenten **1002** in dem gegenwärtigen Kontext, wobei die E-Mail-Nachricht **1751** betrachtet wird, interpretiert.

[0110] In [Fig. 20](#) hat der Benutzer einen Befehl **2050** zur Verfügung gestellt: „Antworte lass uns dies sofort zum Marketing bringen“. Kontextinformation, einschließlich Information über die E-Mail-Nachricht **1751** und die E-Mail-Anwendung, in welcher sie dargestellt wird, wird verwendet, um den Befehl **2050** zu interpretieren. Dieser Kontext kann verwendet werden, um die Bedeutung der Wörter „Antworte“ und „dies“ bei Befehl **2050** zu bestimmen, und aufzuklären, wie eine E-Mail-Komposition-Transaktion für einen bestimmten Empfänger bei einem bestimmten Nachrichtenstrang aufzubauen ist. In diesem Fall ist der virtuelle Assistent **1002** in der Lage, auf Kontextinformationen zuzugreifen, um zu bestimmen, dass „Marketing“ sich auf einen Empfänger namens John Applecore bezieht, und ist in der Lage, eine zu verwendende E-Mail-Adresse für den Empfänger zu bestimmen. Dementsprechend stellt der virtuelle Assistent **1002** E-Mail **2052** für den Benutzer zur Genehmigung und zum Versenden zusammen. Auf diese

Weise ist der virtuelle Assistent **1002** in der Lage, einen Task (Zusammenstellen einer E-Mail-Nachricht) basierend auf Benutzereingabe zusammen mit Kontextinformation, die den Zustand der gegenwärtigen Anwendung beschreibt, zu operationalisieren.

[0111] Anwendungskontext kann auch helfen, die Bedeutung der Benutzerabsicht zwischen Anwendungen zu identifizieren. In [Fig. 21](#), auf die nun Bezug genommen wird, wird ein Beispiel gezeigt, bei welchem der Benutzer den virtuellen Assistenten **1002** in dem Kontext der Darstellung einer E-Mail-Nachricht (wie E-Mail-Nachricht **1751**) aufgerufen hat, aber der Befehl des Benutzers **2150** sagt „Sende ihm einen Text ...“. Der Befehl **2150** wird von dem virtuellen Assistenten **1002** so interpretiert, dass eine Textnachricht anstelle einer E-Mail, gesendet werden soll. Jedoch indiziert das Wort „ihm“, dass der gleiche Empfänger (John Appleseed) beabsichtigt ist. Der virtuelle Assistent **1002** erkennt also, dass die Kommunikation zu diesem Empfänger gehen soll, jedoch auf einem anderen Kanal (eine Textnachricht zu der Telefonnummer der Person), die aus Kontaktinformationen erhalten wird, die auf der Vorrichtung gespeichert sind. Dementsprechend stellt der virtuelle Assistent **1002** Textnachricht **2152** für den Benutzer zur Genehmigung und zum Senden zusammen.

[0112] Beispiele von Kontextinformation, die von Anwendung(en) erhalten werden können, beinhalten, ohne Beschränkung:

- Identität der Anwendung;
- gegenwärtige(s) Objekt oder Objekte, die in der Anwendung bearbeitet werden, wie gegenwärtige E-Mail-Nachricht, gegenwärtig gespieltes Lied oder Liedliste oder Kanal, gegenwärtiges Buch oder Film oder Foto, gegenwärtiger Kalender/Tag/Woche/Monat, gegenwärtige Erinnerungsliste, gegenwärtiger Telefonanruf, gegenwärtige Textnachrichtenunterhaltung, gegenwärtiger Kartenstandort, gegenwärtige Webseite oder Suchanfrage, gegenwärtige Stadt oder anderer Standort für Standort-sensitive Anwendungen, gegenwärtiges soziales Netzwerk-Profil oder jegliche andere anwendungsspezifische Vorstellung von gegenwärtigen Objekten;
- Namen, Orte, Daten und andere identifizierbare Entitäten oder Werte, die aus den gegenwärtigen Objekten extrahiert werden können.

### Persönliche Datenbanken **1058**

[0113] Eine andere Quelle von Kontextdaten ist/sind die persönlichen Datenbank(en) **1058** des Benutzers auf einer Vorrichtung, wie einem Telefon, wie beispielsweise ein Adressbuch, welches Namen und Telefonnummern enthält. In [Fig. 14](#), auf die nun Bezug genommen wird, ist ein Beispiel einer Bildschirmkopie **1451** dargestellt, gemäß einer Ausführungsform, bei welcher der virtuelle Assistent **1002** zum Eindeu-

tig-Machen eines Namens auffordert. Hier hat der Benutzer gesagt „Rufe Herb an“; der virtuelle Assistent **1002** fordert den Benutzer auf, zwischen den passenden Kontakten in dem Adressbuch des Benutzers zu wählen. Folglich wird das Adressbuch als eine Quelle von persönlichem Daten-Kontext verwendet.

**[0114]** Bei einer Ausführungsform wird persönliche Information des Benutzers aus den persönlichen Datenbanken **1058** zur Verwendung als Kontext zum Interpretieren und/oder Operationalisieren der Benutzerabsicht oder anderer Funktionen des virtuellen Assistenten **1002** erhalten. Beispielsweise können Daten in einer Kontaktdatenbank eines Benutzers verwendet werden, um Mehrdeutigkeit beim Interpretieren eines Befehls eines Benutzers zu reduzieren, wenn der Benutzer zu jemandem lediglich mittels Vorname Bezug nimmt. Beispiele von Kontextinformation, die aus persönlichen Datenbanken **1058** erhalten werden können, beinhalten, ohne Beschränkung:

- die Kontaktdatenbank des Benutzers (Adressbuch) – beinhaltend Information über Namen, Telefonnummern, physikalische Adressen, Netzwerkadressen, Kontoidentifizierung, wichtige Daten – über Personen, Firmen, Organisationen, Orte, Webseiten und andere Identitäten, auf die sich ein Benutzer beziehen könnte;
- der eigene Name des Benutzers, bevorzugte Aussprachen, Adressen, Telefonnummern und Ähnliches;
- die vom Benutzer benannten Beziehungen, wie Mutter, Vater, Schwester, Chef und Ähnliches;
- die Kalenderdaten des Benutzers, beinhaltend Kalenderereignisse, Benennung spezieller Tage oder jegliche andere benannte Einträge, auf die sich ein Benutzer beziehen könnte;
- die Erinnerungen oder Tasklisten des Benutzers, beinhaltend Listen von zu erledigenden, zu erinnernden oder zu besorgenden Dingen, auf welcher sich der Benutzer beziehen könnte;
- Namen von Liedern, Genres, Liedlisten und andere Daten, die mit der Musikbibliothek des Benutzers assoziiert sind, auf die sich der Benutzer beziehen könnte;
- Personen, Orte, Kategorien, Tags, Label oder andere symbolische Namen auf Fotos oder Videos oder anderen Medien in der Medienbibliothek des Benutzers;
- Titel, Autoren, Genres oder andere symbolische Namen in Büchern oder anderer Literatur in der persönlichen Bibliothek des Benutzers.

### Dialoghistorie **1052**

**[0115]** Eine andere Quelle von Kontextdaten ist die Dialoghistorie **1052** des Benutzers mit dem virtuellen Assistenten **1002**. Eine solche Historie kann beispielsweise Bezugnahmen auf Bereiche, Personen, Orte usw. beinhalten. In [Fig. 15](#), auf die nun Bezug

genommen wird, ist ein Beispiel gezeigt, in welchem der virtuelle Assistent **1002** Dialogkontext verwendet, um gemäß einer Ausführungsform den Standort für einen Befehl zu erschließen. In Bildschirm **1551** fragt der Benutzer zunächst „Wie viel Uhr ist es in New York?“; der virtuelle Assistent **1002** antwortet **1552**, indem er die gegenwärtige Uhrzeit in New York City bereitstellt. Der Benutzer fragt dann „Wie ist das Wetter?“. Der virtuelle Assistent **1002** verwendet die vorausgegangene Dialoghistorie, um zu erschließen, dass der für die Wetteranfrage beabsichtigte Standort der in der Dialoghistorie zuletzt erwähnte Standort ist. Deshalb stellt seine Antwort **1553** Wetterinformationen für New York City zur Verfügung.

**[0116]** Falls der Benutzer, in einem weiteren Beispiel, sagt „Finde Kamera-Geschäfte in der Nähe“ und anschließend, nach dem Prüfen der Ergebnisse, sagt „Wie sieht's in San Francisco aus?“, kann der Assistent den Dialogkontext verwenden, um zu bestimmen, dass „Wie sieht's aus?“ bedeutet „Mache den gleichen Task (Finde Kamera-Geschäfte)“ und „in San Francisco“ bedeutet „Ändere den Ort der Suche von hier auf San Francisco“. Der virtuelle Assistent **1002** kann als Kontext auch vorangegangene Einzelheiten eines Dialogs, wie vorangegangene Ausgabe, die dem Benutzer bereitgestellt wird, verwenden. Beispielsweise könnte der virtuelle Assistent **1002**, falls er eine kluge Antwort, die als Humor beabsichtigt ist, wie „Auf jeden Fall, Du bist der Boss“ verwendet, erinnern, dass er dies bereits gesagt hat und so verhindern, diese Phrase innerhalb einer Dialogsitzung zu wiederholen.

**[0117]** Beispiele von Kontextinformation aus Dialoghistorie und Speicher des virtuellen Assistenten beinhalten, ohne Beschränkung:

- Personen, die in einem Dialog erwähnt sind;
- Orte und Standorte, die in einem Dialog erwähnt sind;
- gegenwärtig fokussierter Zeitrahmen;
- gegenwärtig fokussierter Anwendungs-Bereich, wie E-Mail oder Kalender;
- gegenwärtig fokussierter Task, wie Lesen einer E-Mail oder Erstellen eines Kalendereintrags;
- gegenwärtig fokussierte Bereichs-Objekte, wie eine E-Mail-Nachricht, die gerade gelesen wurde oder ein Kalendereintrag, der gerade erzeugt wurde;
- gegenwärtiger Zustand eines Dialogs oder Transaktionsflusses, wie ob eine Frage gestellt wird und welche möglichen Antworten erwartet werden;
- Historie von Benutzeranfragen, wie „gute italienische Restaurants“;
- Historie von Ergebnissen von Benutzeranfragen, wie Mengen von zurückgegebenen Restaurants;

- Historie von Phrasen, die von dem Assistenten im Dialog verwendet werden;
- Fakten, die dem Assistenten vom Benutzer mitgeteilt wurden, wie „Meine Mutter ist Rebecca Richards“ und „Ich möchte dieses Restaurant“.

[0118] In [Fig. 25A](#) und [Fig. 25B](#), auf die nun Bezug genommen wird, wird gemäß einer Ausführungsform eine Serie von Bildschirmkopien dargestellt, welche ein Beispiel der Verwendung von früherem Dialogkontext zeigen. In [Fig. 25A](#) hat der Benutzer eine Anfrage **2550** für irgendeine neue E-Mail von John eingegeben. Der virtuelle Assistent **1002** antwortet, indem er eine E-Mail-Nachricht **2551** von John anzeigt. In [Fig. 25B](#) hat der Benutzer den Befehl **2552** „Antworte lass uns dies sofort zum Marketing bringen“ eingegeben. Der virtuelle Assistent **1002** interpretiert den Befehl **2552** unter Verwendung von früherem Dialogkontext; konkret wird der Befehl so interpretiert, dass er sich auf die E-Mail-Nachricht **2551**, die in [Fig. 25](#) angezeigt wird, bezieht.

#### Vorrichtungssensordaten **1056**

[0119] In einer Ausführungsform kann eine physikalische Vorrichtung, auf die der virtuelle Assistent **1002** läuft, einen oder mehrere Sensoren aufweisen. Solche Sensoren können Quellen von Kontextinformation zur Verfügung stellen. Beispiele solcher Information beinhalten, ohne Beschränkung:

- den gegenwärtigen Standort des Benutzers;
- die lokale Zeit am gegenwärtigen Standort des Benutzers;
- die Position, Orientierung und Bewegung der Vorrichtung;
- das gegenwärtige Lichtlevel, die Temperatur und andere Maße der Umwelt;
- die Eigenschaften der verwendeten Mikrofone und Kameras;
- die gegenwärtig verwendeten Netzwerke und Signaturen von verbundenen Netzwerken, beinhaltend Ethernet, Wi-Fi und Bluetooth. Signaturen beinhalten MAC-Adressen von Netzwerk-zugriffspunkten, zugewiesene IP-Adressen, Vorrichtungsidentifizierungen wie Bluetooth-Namen, Frequenzkanäle und andere Eigenschaften von drahtlosen Netzwerken.

[0120] Sensoren können von jeglichem Typ sein, einschließlich beispielsweise: ein Beschleunigungsmesser, Kompass, GPS-Einheit, Höhenmesser, Lichtsensor, Thermometer, Barometer, Uhr, Netzwerkschnittstelle, Stromspeicher-Test-Schaltkreis und Ähnliches.

#### Anwendungspräferenzen und Verwendungshistorie **1072**

[0121] Bei einer Ausführungsform werden Informationen, die die Präferenzen und Einstellungen des

Benutzers für verschiedene Anwendungen beschreiben, wie auch seine oder ihre Verwendungshistorie **1072**, als Kontext zum Interpretieren und/oder Operationalisieren der Benutzerabsicht oder anderer Funktionen des virtuellen Assistenten **1002** verwendet. Beispiele von solchen Präferenzen und Historie beinhalten, ohne Beschränkung:

- Verknüpfungen, Favoriten, Lesezeichen, Listen von Freunden, oder jegliche andere Zusammenstellung von Nutzerdaten über Personen, Firmen, Adressen, Telefonnummern, Orte, Webseiten, E-Mail-Nachrichten oder jegliche andere Referenzen;
- auf der Vorrichtung kürzlich ausgeführte Anrufe;
- kürzliche Textnachrichten-Unterhaltungen, einschließlich der an den Konversationen Beteiligten;
- kürzliche Anfragen nach Karten oder Richtungen;
- kürzliche Web-Suchen und URLs;
- Aktien, die in einer Aktien-Anwendung gelistet sind;
- kürzlich gespielte(s) Lieder oder Video oder andere Medien;
- die Bezeichnung von Alarmen, die bei Alarm-Anwendungen eingestellt sind;
- die Bezeichnungen von Anwendungen oder anderen digitalen Objekten auf der Vorrichtung;
- die bevorzugte Sprache des Benutzers oder die am Standort des Benutzers verwendete Sprache.

[0122] In [Fig. 16](#), auf die nun Bezug genommen wird, ist ein Beispiel der Verwendung einer Telefon-Favoritenliste als eine Quelle von Kontext gemäß einer Ausführungsform dargestellt. Bildschirm **1650** zeigt eine Liste von Favoriten-Kontakten **1651**. Falls der Benutzer die Eingabe „Rufe John an“ bereitstellt, kann diese Liste von Favoriten-Kontakten **1651** verwendet werden, um zu bestimmen, dass sich „John“ auf die Mobilnummer von John Appleseed bezieht, da diese Nummer in der Liste erscheint.

#### Ereigniskontext **2706**

[0123] Bei einer Ausführungsform ist der virtuelle Assistent **1002** in der Lage, Kontext, der mit asynchronen Ereignissen, die unabhängig von der Interaktion des Benutzers mit dem virtuellen Assistenten **1002** auftreten, zu verwenden. In [Fig. 22](#) bis [Fig. 24](#), auf die nun Bezug genommen wird, ist ein Beispiel gemäß einer Ausführungsform dargestellt, welches Aktivieren des virtuellen Assistenten **1002** zeigt, nachdem ein Ereignis auftritt, welches Ereigniskontext oder Alarmkontext bereitstellt. In diesem Fall ist das Ereignis eine eingehende Textnachricht **2250**, wie in [Fig. 22](#) gezeigt. In [Fig. 23](#) wurde der virtuelle Assistent **1002** aufgerufen und die Textnachricht **2250** wird zusammen mit Eingabeaufforderung **1251** angezeigt. Bei [Fig. 24](#) hat der Benutzer den Befehl „Rufe ihn an“ **2450** eingegeben. Der virtuelle Assistent **1002** verwendet den Ereigniskontext, um den Befehl eindeu-

tig zu machen, indem er interpretiert, dass „ihn“ die Person meint, die die eingehende Textnachricht **2250** gesendet hat. Der virtuelle Assistent **1002** verwendet weiterhin den Ereigniskontext, um zu bestimmen, welche Telefonnummer für den ausgehenden Anruf zu verwenden ist. Bestätigungs Nachricht **2451** wird gezeigt, um anzugeben, dass der Anruf ausgeführt wird.

**[0124]** Beispiele von Alarmkontextinformation beinhalten, ohne Beschränkung:

- eingehende Textnachrichten oder Seiten;
- eingehende E-Mail-Nachrichten;
- eingehende Telefonanrufe;
- Erinnerungs-Mitteilungen oder Task-Alarne;
- Kalenderalarme;
- Wecker, Timer oder andere zeitbasierte Alarne;
- Mitteilungen von Ergebnissen oder anderen Ereignissen von Spielen;
- Mitteilungen von finanziellen Ereignissen wie Aktienpreis-Alarme;
- Nachrichten-Kurzmeldungen (News Flashes) oder andere Rundfunkmitteilungen;
- Push-Mitteilungen aus jeglicher Anwendung.

#### Persönliche akustische Kontextdaten **1080**

**[0125]** Beim Interpretieren von Spracheingabe kann der virtuelle Assistent **1002** auch die akustische Umgebung, in welcher die Sprache eingegeben wird, berücksichtigen. Beispielsweise sind die Geräuschanprofile eines ruhigen Büros unterschiedlich von denen von Automobilen oder öffentlichen Orten. Falls ein Spracherkennungssystem akustische Profildaten identifizieren und speichern kann, können diese Daten auch als Kontextinformation zur Verfügung gestellt werden. Wenn diese mit anderer Kontextinformation wie den Eigenschaften des verwendeten Mikrofons, dem gegenwärtigen Ort und dem gegenwärtigen Dialogzustand kombiniert wird, kann akustischer Kontext beim Erkennen und Interpretieren von Eingaben behilflich sein.

#### Darstellen und Zugreifen auf Kontext

**[0126]** Wie voranstehend beschrieben kann der virtuelle Assistent **1002** Kontextinformationen aus jeglichen aus einer Reihe unterschiedlicher Quellen verwenden. Jeglicher aus einer Reihe von Mechanismen kann zur Darstellung von Kontext verwendet werden, so dass dieser für den virtuellen Assistenten **1002** verfügbar gemacht werden kann. In [Fig. 8a](#) bis [Fig. 8d](#), auf die nun Bezug genommen wird, werden mehrere Beispiele von Darstellungen von Kontextinformation dargestellt, wie sie in Verbindung mit verschiedenen Ausführungsformen der vorliegenden Erfindung verwendet werden können.

Darstellen von Personen, Orten, Zeiten, Bereichen, Tasks und Objekten

**[0127]** [Fig. 8a](#) zeigt Beispiele **801** bis **809** von Kontextvariablen, die einfache Eigenschaften wie Geo-Koordinaten des gegenwärtigen Standorts des Benutzers darstellen. Bei einer Ausführungsform können die gegenwärtigen Werte für eine Kernmenge von Kontextvariablen erhalten werden. Beispielsweise kann es einen gegenwärtigen Benutzer, einen gegenwärtig fokussierten Ort, einen gegenwärtig fokussierten Zeitrahmen, einen gegenwärtig fokussierten Anwendungs-Bereich, einen gegenwärtig fokussierten Task und ein gegenwärtig fokussiertes Bereichs-Objekt geben. Eine Datenstruktur, wie beispielsweise in [Fig. 8a](#) gezeigt, kann für eine solche Darstellung verwendet werden.

**[0128]** [Fig. 8b](#) zeigt Beispiel **850** einer komplexeren Darstellung, die zum Speichern von Kontextinformation für einen Kontakt verwendet werden kann. Auch wird ein Beispiel **851** einer Darstellung beinhaltend Daten für einen Kontakt gezeigt. Bei einer Ausführungsform kann ein Kontakt (oder eine Person) als ein Objekt mit Eigenschaften für Name, Geschlecht, Adresse, Telefonnummer und anderen Eigenschaften, die in einer Kontaktdatenbank gehalten werden können, dargestellt sein. Ähnliche Darstellungen können verwendet werden für Orte, Zeiten, Anwendungs-Bereiche, Tasks, Bereichs-Objekte und Ähnliches.

**[0129]** Bei einer Ausführungsform sind Mengen von gegenwärtigen Werten eines bestimmten Typs dargestellt. Solche Mengen können sich auf gegenwärtige Personen, gegenwärtige Orte, gegenwärtige Zeiten und Ähnliches beziehen.

**[0130]** Bei einer Ausführungsform werden Kontextwerte in einer Historie angeordnet, so dass bei Iteration N ein Rahmen von aktuellen Kontextwerten und ebenso ein Rahmen von Kontextwerten, die bei Iteration N – 1 aktuell waren, vorliegt, und zurückgeht bis zu einer gewünschten Grenze der Länge der Historie. [Fig. 8c](#) stellt ein Beispiel eines Arrays **811** dar, welches eine Historie von Kontextwerten beinhaltet. Konkret stellt jede Spalte von [Fig. 8c](#) eine Kontextvariable dar, wobei die Zeilen unterschiedlichen Zeiten entsprechen.

**[0131]** Bei einer Ausführungsform werden Mengen von typbezogenen Kontextvariablen in Historien wie in [Fig. 8d](#) gezeigt angeordnet. In dem Beispiel wird eine Menge **861** von Kontextvariablen, die sich auf Personen beziehen, zusammen mit einer anderen Menge **871** von Kontextvariablen, die sich auf Orte beziehen, gezeigt. Folglich können relevante Kontextdaten für eine bestimmte Zeit in der Historie abgerufen und angewendet werden.

**[0132]** Ein Fachmann wird erkennen, dass die bestimmten in [Fig. 8a](#) bis [Fig. 8d](#) gezeigten Darstellungen lediglich beispielhaft sind, und dass viele andere Mechanismen und/oder Datenformate zur Darstellung von Kontext verwendet werden können. Beispiele beinhalten:

- Bei einer Ausführungsform kann der gegenwärtige Benutzer des Systems auf spezielle Weise dargestellt sein, so dass der virtuelle Assistent **1002** weiß, wie der Benutzer zu adressieren ist und wie auf die Heim-, Arbeitsplatz-, Mobiltelefonnummer des Benutzers und Ähnliches Bezug zu nehmen ist.
- Bei einer Ausführungsform können die Beziehungen zwischen Personen dargestellt sein, was dem virtuellen Assistenten **1002** ermöglicht, Bezugnahmen wie „meine Mutter“ oder „das Haus meines Chefs“ zu verstehen.
- Orte können als Objekte mit Eigenschaften wie Namen, Straßenadresse, Geo-Koordinaten und Ähnliches dargestellt sein.
- Zeiten können als Objekte mit Eigenschaften beinhaltend Weltzeit, Zeitzonenversatz, Auflösung (wie Jahr, Monat, Tag, Stunde, Minute oder Sekunde) dargestellt sein. Zeitobjekte können auch symbolische Zeiten wie „heute“, „diese Woche“, „dieses (bevorstehende) Wochenende“, „nächste Woche“, „Geburtstag von Annie“ und Ähnliches darstellen. Zeitobjekte können auch Zeitdauer oder Zeitpunkte darstellen.
- Kontext kann auch bezogen auf einen Anwendungsbereich, der einen Dienst oder Anwendung oder Bereich eines Diskurs darstellt, zur Verfügung gestellt sein, wie E-Mail, Textnachricht, Telefon, Kalender, Kontakte, Fotos, Videos, Karten, Wetter, Erinnerungen, Uhr, Web-Browser, Facebook, Pandora usw. Der gegenwärtige Bereich gibt an, welche dieser Bereiche relevant sind.
- Kontext kann auch eine oder mehrere Tasks oder auszuführende Operationen innerhalb eines Bereichs definieren. Beispielsweise gibt es innerhalb des E-Mail-Bereichs Tasks wie E-Mail-Nachricht lesen, E-Mail suchen, neue E-Mail erstellen und Ähnliches.
- Bereichs-Objekte sind Datenobjekte, die mit den verschiedenen Bereichen assoziiert sind. Beispielsweise operiert der E-Mail-Bereich auf E-Mail-Nachrichten, der Kalender-Bereich operiert auf Kalenderereignisse und Ähnliches.

**[0133]** Zum Zwecke der hierin wiedergegebenen Beschreibung wird auf diese Darstellungen von Kontextinformation als Kontextvariablen eines bestimmten Typs Bezug genommen. Beispielsweise ist eine Darstellung des gegenwärtigen Benutzers eine Kontextvariable vom Typ Person.

#### Darstellen von Kontext Ableitung

**[0134]** Bei einer Ausführungsform ist die Ableitung von Kontextvariablen ausdrücklich dargestellt, so dass sie bei der Informationsverarbeitung verwendet werden kann. Die Ableitung von Kontextinformation ist eine Charakterisierung der Quelle und/oder Mengen von Rückschlüssen, die gemacht worden sind, um auf die Information zu schließen oder diese zu erhalten. Beispielsweise könnte ein Person-Kontextwert **851** wie in [Fig. 8b](#) gezeigt von einem Text-Nachrichten-Bereichs-Objekt abgeleitet worden sein, welches von Ereigniskontext **2706** erfasst wurde. Diese Quelle des Kontextwertes **851** kann dargestellt sein.

#### Darstellen einer Historie von Benutzeranfragen und/oder Absicht

**[0135]** Bei einer Ausführungsform kann eine Historie der Anfragen des Benutzers gespeichert sein. Bei einer Ausführungsform kann auch eine Historie der Tiefenstruktur, die die Benutzerabsicht (wie aus der Verarbeitung natürlicher Sprache abgeleitet) darstellt, gespeichert sein. Dies ermöglicht dem virtuellen Assistenten **1002** den Sinn von neuen Eingaben in dem Kontext von vorangegangenen interpretierten Eingaben zu verstehen. Fragt der Benutzer beispielsweise „Wie ist das Wetter in New York?“ könnte Sprachinterpretierer **2770** die Frage so interpretieren, dass sie sich auf den Ort New York bezieht. Falls der Benutzer anschließend sagt „Wie ist es an diesem Wochenende?“ kann der virtuelle Assistent **1002** zu dieser vorangegangenen Interpretation Bezug nehmen, um zu bestimmen, dass „Wie ist es“ als „Wie ist das Wetter“ interpretiert werden soll.

#### Darstellen einer Historie von Ergebnissen

**[0136]** Bei einer Ausführungsform kann eine Historie der Ergebnisse von Benutzeranfragen in der Form von Bereichsobjekten gespeichert sein. Beispielsweise kann die Benutzeranfrage „Finde mir einige gute italienische Restaurants“ eine Menge von Bereichsobjekten zurückgeben, die Restaurants darstellen. Falls der Benutzer dann einen Befehl wie „Rufe bei Amilio's an“ eingibt, kann der virtuelle Assistent **1002** die Ergebnisse nach Restaurants namens Amilio's innerhalb der Suchergebnisse durchsuchen, welches eine geringere Menge als alle möglichen Orte, die angerufen werden können, darstellt.

#### Verzögertes Anbinden von Kontextvariablen

**[0137]** Bei einer Ausführungsform können Kontextvariablen Informationen darstellen, die auf Abruf (on demand) abgerufen oder abgeleitet werden. Beispielsweise kann eine Kontextvariable, die den aktuellen Ort darstellt, wenn auf sie zugegriffen wird, eine API aufrufen, die aktuelle Ortsdaten von einer Vorrhaltung abruft, und führt anschließend andere Ver-

arbeitung zum Berechnen beispielsweise einer Straßenadresse aus. Der Wert dieser Kontextvariable kann für eine Zeitspanne erhalten werden, abhängig von einer Caching-Strategie.

### Suchen von Kontext

**[0138]** Der virtuelle Assistent **1002** kann jeglichen aus Reihe von unterschiedlichen Ansätzen zum Suchen nach relevanter Kontextinformation verwenden, um Informationsverarbeitungsprobleme zu lösen. Beispiele von unterschiedlichen Typen von Suchen beinhalten, ohne Beschränkung:

- Suche mit Kontextvariablen-Name. Falls der Name einer benötigten Kontextvariable bekannt ist, wie „Aktueller Benutzer-Vorname“, kann der virtuelle Assistent **1002** nach Instanzen hiervon suchen. Falls eine Historie gehalten wird, kann der virtuelle Assistent **1002** aktuelle Werte zuerst und anschließend frühere Daten konsultieren, bis ein Treffer gefunden wird.
- Suche mit Kontextvariablen-Typ. Falls der Typ einer benötigten Kontextvariable bekannt ist, wie Person, kann der virtuelle Assistent **1002** nach Instanzen von Kontextvariablen dieses Typs suchen. Falls eine Historie gehalten wird, kann der virtuelle Assistent **1002** zunächst die aktuellen Werte suchen, und anschließend frühere Werte konsultieren, bis ein Treffer gefunden wird.

**[0139]** In einer Ausführungsform wird die Suche beendet, sobald ein Treffer gefunden wurde, falls das aktuelle Informationsverarbeitungsproblem einen einzigen Treffer erfordert. Falls mehrere Treffer erlaubt sind, können passende Treffer abgerufen werden, bis eine Grenze erreicht wird.

**[0140]** Falls situationsgerecht kann der virtuelle Assistent **1002** bei einer Ausführungsform seine Suche auf Daten, die eine bestimmte Ableitung aufweisen, beschränken. Falls der virtuelle Assistent **1002** beispielsweise nach Personen-Objekten innerhalb eines Task-Flusses für E-Mail sucht, kann er nur Kontextvariablen in Betracht ziehen, deren Ableitung eine Anwendung, die mit diesem Bereich assoziiert ist, darstellt.

**[0141]** Bei einer Ausführungsform verwendet der virtuelle Assistent **1002** Regeln, um Treffer gemäß Heuristiken einzustufen, unter Verwendung jeglicher verfügbarer Eigenschaften von Kontextvariablen. Beispielsweise interpretiert der virtuelle Assistent **1002** bei der Verarbeitung von Benutzereingabe beinhaltet einen Befehl „Sage ihr, dass ich mich verspäte“ den Begriff „ihr“ durch Kontextreferenz. Dadurch kann der virtuelle Assistent **1002** eine Einstufung anwenden, um eine Präferenz für Personen-Objekte anzugeben, deren Ableitung Anwendungsverwendungshistorien für Kommunikationsanwendungen wie Textnachrichten und E-Mail darstellt.

Bei einem weiteren Beispiel kann der virtuelle Assistent **1002** beim Interpretieren eines Befehls „Rufe sie an“ eine Einstufung anwenden, um Personen-Objekte, die Telefonnummern aufweisen, gegenüber denjenigen, deren Telefonnummern nicht bekannt sind, zu bevorzugen. Bei einer Ausführungsform können Einstufungsregeln mit Bereichen assoziiert sein. Beispielsweise können unterschiedliche Einstufungsregeln verwendet werden, um Personen-Variablen für E-Mail- und Telefonbereiche einzustufen. Ein Fachmann wird erkennen, dass jegliche solche Einstufungsregel(n) erzeugt und/oder angewendet werden kann/können, abhängig von der bestimmten Darstellung und Zugriff auf benötigte Kontextinformation.

### Verwendung von Kontext zur Verbesserung der Verarbeitung des virtuellen Assistenten

**[0142]** Wie voranstehend beschrieben lässt sich Kontext auf eine Vielfalt von Berechnungen und Rückschlüssen in Zusammenhang mit der Operation des virtuellen Assistenten **1002** anwenden. In [Fig. 2](#), auf die nun Bezug genommen wird, ist ein Flussdiagramm dargestellt, welches gemäß einer Ausführungsform ein Verfahren **10** zur Verwendung von Kontext bei verschiedenen Verarbeitungsstufen beim virtuellen Assistenten **1002** zeigt.

**[0143]** Das Verfahren **10** kann in Zusammenhang mit einer oder mehrerer Ausführungsformen des virtuellen Assistenten **1002** implementiert sein.

**[0144]** Bei zumindest einer Ausführungsform kann das Verfahren **10** betreibbar sein, um verschiedene Typen von Funktionen, Operationen, Aktionen und/oder anderen Merkmalen wie beispielsweise eine oder mehr der folgenden (oder Kombinationen hieraus) auszuführen und/oder zu implementieren:

- Ausführen einer Flussschleife einer Schnittstellensteuerung von einer interaktiven Schnittstelle zwischen dem Benutzer und virtuellen Assistenten **1002**. Zumindest eine Iteration des Verfahrens **10** kann als eine Lage (ply) in der Unterhaltung dienen. Eine interaktive Schnittstelle ist eine Schnittstelle, bei der der Benutzer und Assistent **1002** kommunizieren, indem sie in einer interaktiven Weise Äußerungen hin und her machen.
- Bereitstellen von ausführendem Steuerungsfluss für virtuellen Assistenten **1002**. Dies bedeutet, dass die Prozedur das Sammeln von Eingabe, Verarbeiten der Eingabe, Generieren von Ausgabe und Präsentieren der Ausgabe zu dem Benutzer steuert.
- Koordinieren von Kommunikationen zwischen Komponenten des virtuellen Assistenten **1002**. Dies bedeutet er kann anweisen, wo die Ausgabe einer Komponente eine andere versorgt, und wo die gesamte Eingabe aus der Umgebung und Aktion auf der Umgebung auftreten kann.

**[0145]** Bei zumindest einigen Ausführungsformen können Teile des Verfahrens **10** auch auf anderen Vorrichtungen und/oder Systemen eines Computer-Netzwerks implementiert sein.

**[0146]** Gemäß spezifischer Ausführungsformen können mehrere Instanzen oder Threads von Verfahren **10** gleichzeitig implementiert und/oder initiiert sein, durch die Verwendung von einem oder mehreren Prozessor(en) **63** und/oder anderen Kombinationen von Hardware und/oder Hardware und Software. Bei zumindest einer Ausführungsform kann/können ein oder mehrere oder ausgewählte Teil(e) von Verfahren **10** auf einem oder mehreren Client(s) **1304**, auf einem oder mehreren Server(n) **1340** und/oder Kombinationen hieraus implementiert sein.

**[0147]** Beispielsweise können bei zumindest einigen Ausführungsformen verschiedene Aspekte, Merkmale und/oder Funktionalitäten von Verfahren **10** mittels Software-Komponenten, Netzwerkdiensten, Datenbanken und/oder Ähnlichem oder jeglicher Kombination hieraus durchgeführt, implementiert und/oder initiiert sein.

**[0148]** Gemäß unterschiedlicher Ausführungsformen kann/können ein(e) oder mehrere unterschiedliche Threads oder Instanzen von Verfahren **10** in Reaktion auf das Erkennen von einer oder mehreren Bedingungen oder Ereignissen, die ein Kriterium oder mehrere unterschiedliche Typen von Kriterien (wie beispielsweise minimale Schwellwertkriterien) zum Auslösen des Initiierens von zumindest einer Instanz von Verfahren **10** erfüllt/erfüllen, initiiert werden. Beispiele von verschiedenen Typen von Bedingungen oder Ereignissen, die Initiiieren und/oder Implementieren von einem/einer oder mehreren unterschiedlichen Threads oder Instanzen des Verfahrens auslösen, können enthalten, aber sind nicht beschränkt auf eine oder mehr der folgenden (oder Kombinationen hieraus):

- Eine Benutzer-Sitzung mit einer Instanz des virtuellen Assistenten **1002**, wie beispielsweise, aber nicht beschränkt auf, eine oder mehrere von:
  - eine Anwendung einer mobilen Vorrichtung, die beispielsweise eine Anwendung einer mobilen Vorrichtung startet, die eine Ausführungsform des virtuellen Assistenten **1002** implementiert;
  - eine Computer-Anwendung, die beispielsweise eine Anwendung startet, die eine Ausführungsform des virtuellen Assistenten **1002** implementiert;
  - Drücken einer dedizierten Taste auf einer mobilen Vorrichtung, wie „Spracheingabe-Taste“;
  - eine Taste auf einer peripheren Vorrichtung, die mit einem Computer oder einer mobilen Vorrichtung verbunden ist, wie ein Headset, Telefon-Handgerät oder Basisstation, ein(e) GPS-Navigationssystem, Unterhaltungsgerät, Fernsteuerung oder jegliche andere Vorrichtung mit einem Tas-

te, der mit dem Aufrufen von Assistenz assoziiert sein könnte;

- eine Web-Sitzung, die von einem Web-Browser gestartet wird, auf einer Webseite, die den virtuellen Assistenten **1002** implementiert;
- eine Interaktion, die von innerhalb einer existierenden Web-Browser-Sitzung gestartet wird, auf eine Webseite, die den virtuellen Assistenten **1002** implementiert, bei welcher beispielsweise der Dienst des virtuellen Assistenten **1002** angefordert wird;
- eine E-Mail-Nachricht, die zu einem Modalitäten-Server **1426** gesendet wird, welcher Kommunikation mit einer Ausführungsform des virtuellen Assistenten **1002** vermittelt;
- eine Textnachricht wird einem Modalitäten-Server **1426** gesendet, welcher Kommunikation mit einer Ausführungsform des virtuellen Assistenten **1002** vermittelt;
- ein Telefonanruf wird zu einem Modalitäten-Server **1434** gemacht, welcher Kommunikation mit einer Ausführungsform des virtuellen Assistenten **1002** vermittelt;
- ein Ereignis wie ein Alarm oder eine Mitteilung wird einer Applikation gesendet, die eine Ausführungsform des virtuellen Assistenten **1002** zur Verfügung stellt,
- wenn eine Vorrichtung, die den virtuellen Assistenten **1002** zur Verfügung stellt, angeschaltet wird und/oder gestartet wird.

**[0149]** Gemäß unterschiedlichen Ausführungsformen können ein(e) oder mehrere unterschiedliche Threads oder Instanzen von Verfahren **10** manuell, automatisch, statisch, dynamisch, gleichzeitig und/oder aus Kombinationen hieraus initiiert werden und/oder implementiert sein. Zusätzlich können unterschiedliche Instanzen und/oder Ausführungsformen von Verfahren **10** zu einem oder mehreren unterschiedlichen Zeitintervallen initiiert werden (z. B. während eines spezifischen Zeitintervalls, zu regelmäßigen periodischen Intervallen, zu unregelmäßigen periodischen Intervallen, auf Abruf und Ähnliches).

**[0150]** Bei zumindest einer Ausführungsform kann eine gegebene Instanz von Verfahren **10** mehrere unterschiedliche Typen von Daten und/oder andere Typen von Information nutzen und/oder generieren, wenn sie spezifische Tasks und/oder Operationen, einschließlich hierin beschriebener Kontextdaten, ausführt. Daten können auch jeglichen anderen Typ von Eingabedaten/Information und/oder Ausgabedaten/Information beinhalten. Beispielsweise kann bei zumindest einer Ausführungsform zumindest eine Instanz von Verfahren **10** auf Information aus einer Quelle oder mehreren unterschiedlichen Typen von Quellen, wie beispielsweise einer oder mehreren Datenbanken, zugreifen, diese verarbeiten und/oder anderweitig nutzen. Bei zumindest einer Aus-

führungsform kann zumindest auf einen Teil der Datenbank-Information mittels Kommunikation mit einer oder mehreren lokaler und/oder entfernter Speicher-vorrichtungen zugegriffen werden. Zusätzlich kann zumindest eine Instanz von Verfahren **10** eine oder mehrere unterschiedliche Typen von Ausgabe-Daten-Information erzeugen, die beispielsweise auf lokalen Speicher- und/oder entfernten Speicher-Vorrichtungen abgelegt sein können.

**[0151]** Bei zumindest einer Ausführungsform kann eine anfängliche Konfiguration einer gegebenen Instanz von Verfahren **10** unter Verwendung von einem oder mehreren unterschiedlichen Typen von Initialisierungsparametern ausgeführt werden. Bei zumindest einer Ausführungsform kann zumindest auf einen Teil der Initialisierungsparameter mittels Kommunikation mit einer oder mehreren lokalen und/oder entfernten Speicher-vorrichtungen zugegriffen werden. Bei zumindest einer Ausführungsform kann zumindest ein Teil der Initialisierungsparameter, die einer Instanz von Verfahren **10** zur Verfügung gestellt sind, dem/der Eingabedaten/Information entsprechen und/oder können hieraus abgeleitet sein.

**[0152]** In dem bestimmten Beispiel von [Fig. 2](#) wird angenommen, dass ein einzelner Benutzer auf eine Instanz des virtuellen Assistenten **1002** über ein Netzwerk von einer Client-Anwendung mit Sprach-eingabefähigkeiten aus zugreift.

**[0153]** Spracheingabe wird ermittelt und interpretiert **100**. Das Ermitteln kann das Präsentieren von Aufforderungen in jeglichem geeigneten Modus beinhalten. Bei verschiedenen Ausführungsformen bietet die Benutzerschnittstelle des Clients mehrere Eingabe-Modi an. Diese können beispielsweise beinhalten:

- eine Schnittstelle für getippte Eingabe, die eine aktive Tipp-Eingabe-Ermittlungs-Prozedur aufrufen kann;
- eine Schnittstelle für Spracheingabe, die eine aktive Spracheingabe-Ermittlungs-Prozedur aufrufen kann;
- eine Schnittstelle zur Auswahl von Eingaben aus einem Menü, die eine aktive GUI-basierte Eingabe-Ermittlung aufrufen kann.

**[0154]** Techniken zum Ausführen jeder dieser Modi sind in den oben einbezogenen verwandten Patentanmeldungen beschrieben. Ein Fachmann wird erkennen, dass andere Eingabe-Modi bereitgestellt werden können. Die Ausgabe bei Schritt **100** ist eine Menge von Kandidateninterpretationen **190** der Eingabesprache.

**[0155]** Die Menge an Kandidateninterpretationen **190** wird von einem Sprachinterpretierer **2770** (auch als natürlicher Sprachprozessor, oder NLP bezeichnet), welcher die Texteingabe parst und eine Menge

von möglichen Interpretationen der Benutzerabsicht **290** generiert, verarbeitet **200**.

**[0156]** Bei Schritt **300** wird/werden die Darstellung(en) der Benutzerabsicht **290** einem Dialog-Fluss-Prozessor **2780** übergeben, der eine Ausführungsform einer Dialog- und Fluss-Analyse-Prozedur wie in Zusammenhang mit [Fig. 5](#) beschrieben implementiert. Der Dialog-Fluss-Prozessor **2780** bestimmt, welche Interpretation der Absicht am wahrscheinlichsten ist, bildet diese Interpretation auf Instanzen von Bereichs-Modellen und Parametern eines Task-Modells ab und bestimmt den nächsten Flusschritt in einem Task-Fluss.

**[0157]** Bei Schritt **400** wird der identifizierte Flusschritt ausgeführt. Bei einer Ausführungsform wird der Aufruf des Flussschrittes durch Dienste-Orchesterungs-Komponente **2782** ausgeführt, welche eine Menge an Diensten im Auftrag der Anfrage des Benutzers aufruft. Bei einer Ausführungsform tragen diese Dienste einige Daten für ein gemeinsames Ergebnis bei.

**[0158]** Bei Schritt **500** wird eine Dialogantwort generiert. Bei Schritt **700** wird die Antwort zu der Client-Vorrichtung zur Ausgabe hierauf gesendet. Client-Software gibt diese auf dem Bildschirm (oder anderen Ausgabevorrichtungen) der Client-Vorrichtung wieder.

**[0159]** Falls der Benutzer, nach Ansicht der Antwort fertig ist **790**, endet das Verfahren. Falls der Benutzer nicht fertig ist, wird eine weitere Iteration der Schleife durch Zurückgehen auf Schritt **100** initiiert.

**[0160]** Kontextinformation **1000** kann von verschiedenen Komponenten des Systems zu verschiedenen Punkten im Verfahren **10** verwendet werden. Beispielsweise kann Kontextinformation **1000**, wie in [Fig. 2](#) gezeigt, bei Schritten **100**, **200**, **300** und **500** verwendet werden. Weitergehende Beschreibung der Verwendung von Kontext **1000** in diesen Schritten wird nachstehend bereitgestellt. Ein Fachmann wird jedoch erkennen, dass die Verwendung von Kontextinformation nicht auf diese spezifischen Schritte beschränkt ist, und dass das System Kontextinformation bei anderen Punkten ebenso nutzen kann, ohne von den wesentlichen Charakteristiken der vorliegenden Erfindung abzuweichen.

**[0161]** Darüber hinaus wird ein Fachmann erkennen, dass unterschiedliche Ausführungsformen von Verfahren **10** zusätzliche Merkmale und/oder Operationen als jene, die für die spezifische in [Fig. 2](#) gezeigte Ausführungsform dargestellt sind, beinhalten können, und/oder können zumindest einen Teil der Merkmale und/oder Operationen von Verfahren **10**, wie für die spezifische Ausführungsform von [Fig. 2](#) dargestellt, weglassen.

## Verwendung von Kontext bei Sprachermittlung und Interpretation

**[0162]** In [Fig. 3](#), auf die nun Bezug genommen wird, ist ein Flussdiagramm dargestellt, welches ein Verfahren zur Verwendung von Kontext bei Sprachermittlung und Interpretation **100** verwendet, um gemäß einer Ausführungsform Spracherkennung zu verbessern. Kontext **1000** kann beispielsweise zum Eindeutig-Machen bei Spracherkennung verwendet werden, um die Generierung, das Einstufen und Filtern von Kandidaten-Hypothesen, die Phoneme zu Wörtern zuordnen, zu führen. Unterschiedliche Spracherkennungssysteme verwenden verschiedene Mischungen von Generierung, Einstufung und Filter, aber Kontext **1000** kann allgemein angewendet werden, um den Hypothesen-Raum bei jeder Stufe zu reduzieren.

**[0163]** Das Verfahren beginnt **100**. Der Assistent **1002** empfängt **121** Stimmen- oder Spracheingabe in der Form eines auditiven Signals. Ein Sprache-zu-Text-Dienst **122** oder Prozessor generiert eine Menge von Kandidatentextinterpretationen **124** des auditiven Signals. Bei einer Ausführungsform wird der Sprache-zu-Text-Dienst **122** unter Verwendung von beispielsweise Nuance Recognizer, verfügbar von Nuance Communications, Inc. aus Burlington, Massachusetts, implementiert.

**[0164]** Bei einer Ausführungsform verwendet der Assistent **1002** statistische Sprachmodelle **1029**, um Kandidatentextinterpretationen **124** von Spracheingabe **121** zu generieren. Bei einer Ausführungsform wird Kontext **1000** verwendet, um die Generierung, Filterung und/oder Einstufung von Kandidateninterpretationen **124**, die von dem Sprache-zu-Text-Dienst **122** generiert wurden, zu beeinflussen. Beispielsweise:

- Sprache-zu-Text-Dienst **122** kann Vokabular aus persönlicher/persönlichen Benutzer-Datenbank(en) **1058** verwenden, um statistische Sprachmodelle **1029** zu beeinflussen.
- Sprache-zu-Text-Dienst **122** kann einen Dialog-Zustand-Kontext verwenden, um ein benutzerdefiniertes statistisches Sprachmodell **1029** auszuwählen. Beispielsweise kann ein statistisches Sprachmodell **1029** beim Stellen einer Ja/Nein-Frage ausgewählt werden, welches bevorzugt diese Wörter hört.
- Sprache-zu-Text-Dienst **122** kann aktuellen Anwendungskontext verwenden, um relevante Wörter zu bevorzugen. Beispielsweise kann „Rufe sie an“ („call her“) gegenüber „Kragen“ („collar“) in einem Textnachrichtenanwendungs-Kontext bevorzugt sein, da ein solcher Kontext Personen-Objekte zur Verfügung stellt, die angerufen werden können.

**[0165]** Beispielsweise kann eine gegebene Spracheingabe den Sprache-zu-Text-Dienst **122** dazu führen, Interpretationen „Rufe sie an“ („call her“) und „Kragen“ („collar“) zu generieren. Geführt von statistischen Sprachmodellen (SLMs) **1029** kann der Sprache-zu-Text-Dienst **122** durch grammatischen Beschränkungen abgestimmt werden, so dass er Namen hört, nachdem er „rufe an“ („call“) gehört hat. Der Sprache-zu-Text-Dienst **122** kann auch basierend auf Kontext **1000** abgestimmt sein. Ist beispielsweise „Herb“ ein Vorname in dem Adressbuch des Benutzers, kann dieser Kontext verwendet werden, um den Schwellwert zu verringern, um „Herb“ als eine Interpretation der zweiten Silbe in Betracht zu ziehen. Dies bedeutet, dass die Präsenz von Namen in dem persönlichen Datenkontext des Benutzers die Auswahl und Abstimmung des statistischen Sprachmodells **1029**, welches zum Generieren von Hypothesen verwendet wird, beeinflussen kann. Der Name „Herb“ kann Teil eines allgemeinen SLM **129** sein, oder er kann direkt durch Kontext **1000** hinzugefügt werden. Bei einer Ausführungsform kann er als zusätzliche SLM **1029** hinzugefügt werden, welche basierend auf Kontext **1000** abgestimmt wird. Bei einer Ausführungsform kann er ein Abstimmen eines existierenden SLM **1029** sein, welches basierend auf Kontext **1000** abgestimmt wird.

**[0166]** Bei einer Ausführungsform werden die statistischen Sprachmodelle **1029** auch darauf abgestimmt, nach Wörtern, Namen und Phrasen aus Anwendungspräferenzen und Verwendungshistorie **1072** und/oder persönlichen Datenbanken **1058**, die im persönlichen Langzeitspeicher **2754** abgelegt sein können, zu suchen. Beispielsweise kann statistischen Sprachmodellen **1029** Text aus zu erledigenden Punkten (to-do items) Listenpunkten (list items), persönlichen Notizen, Kalendereinträgen, Personennamen in Kontakt/Adressbüchern, E-Mail-Adressen, Straßen oder Stadtnamen, welche in Kontakt/Adressbüchern erwähnt sind und Ähnliches gegeben werden.

**[0167]** Eine Einstufungs-Komponente analysiert Kandidateninterpretationen **124** und stuft **126** diese danach ein, wie gut sie zu syntaktischen und/oder semantischen Modellen des virtuellen Assistenten **1002** passen. Jegliche Quelle von Beschränkungen von Benutzereingabe kann verwendet werden. Beispielsweise kann bei einer Ausführungsform der Assistent **1002** die Ausgabe des Sprache-zu-Text-Interpretierers danach einstufen, wie gut die Interpretationen in einem syntaktischen und/oder semantischen Sinne, einem Bereichs-Modell, Task-Fluss-Modell und/oder Dialogmodell und/oder Ähnliches parsen: er evaluier, wie gut verschiedene Kombinationen von Wörtern von Kandidateninterpretationen **124** zu den Konzepten, Beziehungen, Entitäten und Eigenschaften einer aktiven Ontologie und ihrer assoziierten Model-

le passen, wie in den oben eingebundenen verwandten US-Utility Applications beschrieben.

**[0168]** Einstufen **126** von Kandidateninterpretationen kann auch durch Kontext **1000** beeinflusst sein. Führt der Benutzer beispielsweise aktuell eine Unterhaltung in einer Textnachrichtenanwendung durch, wenn der virtuelle Assistent **1002** aufgerufen wird, ist es wahrscheinlicher, dass die Phrase „Rufe sie an“ („call her“) eine richtige Interpretation ist, als das Wort „Kragen“ („collar“), denn es gibt eine mögliche „sie“ („her“), die in diesem Zusammenhang angerufen werden kann. Eine solche Beeinflussung kann durch Abstimmen der Einstufung von Hypothesen **126** erreicht werden, indem Phrasen wie „Rufe sie an“ („call her“) oder „Rufe <Kontaktnname> an“ („call <contact name>“) bevorzugt werden, wenn der aktuelle Anwendungskontext auf eine Anwendung deutet, die „anrufbare Entitäten“ bereitstellen kann.

**[0169]** Bei unterschiedlichen Ausführungsformen können Algorithmen oder Prozeduren, die von dem Assistenten **1002** beim Interpretieren von Texteingaben verwendet werden, einschließlich jeglicher Ausführungsform der Verarbeitungsprozeduren mit natürlicher Sprache, wie in [Fig. 3](#) gezeigt, verwendet werden, um Kandidatentextinterpretationen **124**, welche von dem Sprache-zu-Text-Dienst **122** generiert werden, einzustufen und zu bewerten.

**[0170]** Kontext **1000** kann auch verwendet werden, um Kandidateninterpretationen **124** zu filtern, anstelle des oder zusätzlich zum Beschränken ihrer Generierung oder Beeinflussung ihrer Einstufung. Beispielsweise könnte eine Filterregel vorschreiben, dass der Kontext des Adressbucheneintrags für „Herb“ ausreichend angibt, dass die Phrase, welche diesen enthält, als Spitzenkandidat **130** angesehen werden soll, selbst wenn dieser anderweitig unterhalb eines Filter-Schwellwertes wäre. Abhängig von der bestimmten Spracherkennungstechnologie, die verwendet wird, können Beschränkungen, die auf Kontext-Beeinflussung basieren, bei den Generierungs-, Einstufungs- und/oder Filterstufen verwendet werden.

**[0171]** Bei einer Ausführungsform kann, falls die Einstufungs-Komponente **126** bestimmt **128**, dass die höchst-eingestufte Sprach-Interpretation von Interpretationen **124** über einem spezifizierten Schwellwert liegt, diese höchst-eingestufte Interpretation automatisch selektiert werden **130**. Falls keine Interpretation oberhalb eines spezifizierten Schwellwerts liegt, werden mögliche Kandidateninterpretationen von Sprache **134** dem Benutzer präsentiert **132**. Der Benutzer kann dann zwischen den angezeigten Möglichkeiten auswählen **136**.

**[0172]** In [Fig. 26A](#) und [Fig. 26B](#), auf die nun Bezug genommen wird, sind Bildschirmkopien gezeigt,

welche gemäß einer Ausführungsform ein Beispiel einer Benutzerschnittstelle zur Auswahl zwischen Kandidateninterpretationen zeigen. [Fig. 26A](#) stellt eine Präsentation der Benutzersprache mit Punkten dar, welchen eine mehrdeutige Interpretation **2651** zugrunde liegt. Falls der Benutzer den Text antippt, zeigt sie alternative Interpretationen **2652A**, **2652B** wie in [Fig. 26B](#) dargestellt. Bei einer Ausführungsform kann Kontext **1000** beeinflussen, welche der Kandidateninterpretationen **2652A**, **2652B** eine bevorzugte Interpretation ist (die als eine anfängliche Vorgabe in [Fig. 26A](#) dargestellt ist) und auch die Auswahl einer endlichen Menge von darzustellenden Alternativen wie in [Fig. 26B](#).

**[0173]** Bei verschiedenen Ausführungsformen kann Benutzerauswahl **136** zwischen den angezeigten Möglichkeiten durch jeglichen Eingabemodus erreicht werden, einschließlich beispielsweise multimodaler Eingabe. Solche Eingabemodi beinhalten, ohne Beschränkung, aktiv ermittelte getippte Eingabe, aktiv ermittelte Spracheingabe, aktiv präsentierte GUI zur Eingabe und/oder Ähnliches. Bei einer Ausführungsform kann der Benutzer zwischen Kandidateninterpretationen **134** beispielsweise durch Antippen oder Sprechen wählen. Im Falle von Sprechen ist die mögliche Interpretation der neuen Spracheingabe stark beschränkt durch die kleine Menge an angebotenen Möglichkeiten **134**.

**[0174]** Egal ob Eingabe automatisch selektiert **130** oder durch den Benutzer selektiert **136** wird, wird/werden die eine oder mehr resultierende(n) Textinterpretation(en) **190** zurückgegeben. Bei zumindest einer Ausführungsform ist die zurückgegebene Eingabe kommentiert, so dass Information darüber, welche Auswahlen bei Schritt **136** gemacht wurden, zusammen mit der Texteingabe erhalten bleibt. Dies ermöglicht beispielsweise, dass die semantischen Konzepte oder Entitäten, denen eine Zeichenkette zugrunde liegt, mit der Zeichenkette assoziiert werden, wenn sie zurückgegeben wird, was die Genauigkeit nachfolgender Sprachinterpretationen verbessert.

**[0175]** Jegliche der Quellen, die in Zusammenhang mit [Fig. 1](#) beschrieben wurden, können Kontext **1000** dem Sprachermittlungs- und Interpretationsverfahren, welches in [Fig. 3](#) gezeigt ist, zur Verfügung gestellt werden. Beispielsweise:

- Persönliche Akustische Kontextdaten **1080** werden verwendet, um aus möglichen SLMs **1029** zu wählen, oder diese zum Optimieren für erkannten akustischen Kontext abzustimmen.
- Vorrichtungssensordaten **1056**, welche Eigenschaften von verwendeten Mikrofonen und/oder Kameras beschreiben, können verwendet werden, um aus möglichen SLMs **1029** auszuwählen, oder diese anderweitig zum Optimieren für erkannten akustischen Kontext abzustimmen.

- Vokabular aus persönlichen Datenbanken **1058** und Anwendungspräferenzen und Verwendungshistorie **1072** kann als Kontext **1000** verwendet werden. Beispielsweise können die Titel von Medien und Namen von Künstlern verwendet werden, um Sprachmodelle **1029** abzustimmen.
- Aktueller Dialogzustand, Teil von Dialoghistorie und Assistenzspeicher **1052** können verwendet werden, um Generieren/Filtern/Einstufen von Kandidateninterpretation **124** durch den Text-zu-Sprache-Dienst **122** zu beeinflussen. Beispielsweise stellt eine Art von Dialogzustand eine Ja/Nein-Frage. In einem solchen Zustand kann Verfahren **100** ein SLM **1029** auswählen, welches Hören dieser Wörter bevorzugt, oder es kann das Einstufen und Filtern dieser Wörter bei einer Kontext-spezifischen Anpassung bei **122** beeinflussen.

#### Verwendung von Kontext bei Verarbeitung natürlicher Sprache

**[0176]** Kontext **1000** kann verwendet werden, um Verarbeitung natürlicher Sprache (NLP) zu erleichtern – das Parsen von Texteingabe zu semantischen Strukturen stellt die möglichen Parses (parses) dar. In [Fig. 4](#), auf die nun Bezug genommen wird, ist ein Flussdiagramm dargestellt, welches gemäß einer Ausführungsform ein Verfahren zur Verwendung von Kontext bei Verarbeitung natürlicher Sprache zeigt, welches von einem Sprachinterpretierer **2770** ausgeführt werden kann.

**[0177]** Das Verfahren beginnt **200**. Eingabetext **202** wird empfangen. Bei einer Ausführungsform wird der Eingabetext **202** mit Wörtern und Phrasen unter Verwendung eines Muster-Erkenners **2760**, Vokabular-Datenbanken **2758**, Ontologien und anderer Modelle abgeglichen, um Assoziationen zwischen Benutzereingabe und Konzepten zu identifizieren. Schritt **210** führt zu einer Menge von syntaktischen Kandidaten-Parses **212**, die mit semantischer Relevanz abgeglichen werden **220**, um semantische Kandidaten-Parses **222** zu erzeugen. Anschließend werden Kandidaten-Parses verarbeitet, um bei **230** mehrdeutige Alternativen zu entfernen, werden gefiltert und nach Relevanz sortiert **232** und zurückgegeben.

**[0178]** Während der Verarbeitung natürlicher Sprache kann Kontextinformation **1000** angewendet werden, um den Hypothesen-Raum zu reduzieren und mögliche Parses zu beschränken. Falls der Sprachinterpretierer **2770** beispielsweise zwei Kandidaten „Rufe sie an“ („call her“) und „Rufe Herb an“ („call Herb“) empfängt, würde der Sprachinterpretierer **2770** Anbindungen **212** für die Wörter „Rufe an“ („call“), „sie“ („her“) und „Herb“ finden. Anwendungskontext **1060** kann verwendet werden, um die möglichen Wortbedeutungen für „Rufe an“ („call“) zu beschränken, so dass gemeint ist „Telefonan-

ruf“ („phone call“). Kontext kann auch verwendet werden, um die Bezüge für „sie“ („her“) und „Herb“ zu finden. Für „sie“ („her“) könnten die Kontextquellen **1000** nach einer Quelle für anrufbare Entitäten durchsucht werden. In diesem Beispiel ist der Beteiligte einer Textnachrichten-Unterhaltung eine anrufbare Entität und diese Information ist Teil des Kontexts, der aus der Textnachrichtenanwendung stammt. Im Falle von „Herb“ ist das Adressbuch des Benutzers eine Quelle von Kontext zum Eindeutig-Machen, wie auch andere persönliche Daten wie Anwendungspräferenzen (wie Favoritenummern aus Bereichs-Entitäts-Datenbanken **2772**) und Anwendungsverwendungshistorie (wie kürzliche Telefonanrufe aus Bereichs-Entitäts-Datenbanken **2772**) sind. Bei einem Beispiel, bei dem die aktuelle Beteiligte einer Textnachrichtunterhaltung RebeccaRichards ist, und es einen HerbGowen in dem Adressbuch des Benutzers gibt, wären die zwei Parses, die vom Sprachinterpretierer **2770** erzeugt werden, semantische Strukturen, die „TelefonAnruf (RebeccaRichards)“ („Phone-Call (RebeccaRichards)“) und „TelefonAnruf (HerbGowen)“ („Phone-Call (HerbGowen)“) darstellen.

**[0179]** Daten aus Anwendungspräferenzen und Verwendungshistorie **1072**, Dialoghistorie und Assistenzspeicher **1052** und/oder persönliche Datenbanken **1058** können auch von dem Sprachinterpretierer **2770** beim Generieren von syntaktischen Kandidaten-Parses **212** verwendet werden. Solche Daten können beispielsweise aus Kurz- und/oder Langzeitspeicher **2752**, **2754** erhalten werden. Auf diese Weise kann Eingabe, die früher in der gleichen Sitzung bereitgestellt wurde und/oder bekannte Information über den Benutzer verwendet werden, um Leistung zu verbessern, Mehrdeutigkeit zu reduzieren und die Unterhaltungsnatur der Interaktion zu stärken. Daten aus aktiver Ontologie **1050**, Bereichs-Modelle **2756** und Task-Fluss-Modelle **2786** können ebenso verwendet werden, um beweiskräftige Schlussfolgerungen beim Bestimmen valider syntaktischer Kandidaten-Parses **212** zu implementieren. Beim semantischen Abgleichen **220** betrachtet der Sprachinterpretierer **2770** Kombinationen von möglichen Parse-Ergebnissen danach, wie gut sie zu semantischen Modellen, wie Bereichs-Modellen und Datenbanken passen. Semantisches Abgleichen **220** kann beispielsweise Daten aus aktiver Ontologie **1050**, persönlichem Kurzzeitspeicher **2752** und persönlichem Langzeitspeicher **2754** verwenden. Beispielsweise kann semantisches Abgleichen **220** Daten aus vorangegangenen Bezugnahmen auf Orte oder lokale Ereignisse im Dialog (aus Dialoghistorie und Assistenzspeicher **1052**) oder persönliche Favoriten-Orte (aus Anwendungspräferenzen und Verwendungshistorie **1072**) verwenden. Der Schritt des semantischen Abgleichens **220** verwendet auch Kontext **1000**, um Phrasen zu Bereichs-Absichts-Strukturen zu interpretieren. Eine Menge Kandidaten oder

möglichen, semantischen Parse-Ergebnissen wird generiert **222**.

**[0180]** In einem Schritt des Eindeutig-Machens **230** wähgt der Spracheninterpretierer **2770** die Beweiskraftstärke von semantischen Kandidaten-Parse-Ergebnissen **222** ab. Das Eindeutig-Machen **230** beinhaltet Reduzieren der Anzahl von semantischen Kandidaten-Parse **222** durch Eliminieren unwahrscheinlicher oder redundanten Alternativen. Das Eindeutig-Machen **230** kann beispielsweise Daten aus der Struktur von aktiver Ontologie **1050** verwenden. Bei zumindest einer Ausführungsform stellen die Verbindungen zwischen Knoten in einer aktiven Ontologie beweiskräftige Unterstützung zum Eindeutig-Machen zwischen semantischen Kandidaten-Parse-Ergebnissen **222** dar. Bei einer Ausführungsform wird Kontext **1000** verwendet, um bei diesem Eindeutig-Machen zu helfen. Beispiele eines solchen Eindeutig-Machens beinhalten: Bestimmen einer von mehreren Person(en), die den gleichen Namen hat/haben; Bestimmen eines Bezugs in einem Befehl wie „Antworte“ („reply“) (E-Mail- oder Textnachricht); Pronomen-Dereferenzieren und Ähnliches.

**[0181]** Beispielsweise bezieht sich eine Eingabe wie „Rufe Herb an“ („call Herb“) möglicherweise auf jegliche Entität, die zu „Herb“ passt. Es kann eine beliebige Anzahl solcher Entitäten geben, nicht nur in dem Adressbuch des Benutzers (persönliche Datenbanken **1058**), sondern auch in Datenbanken mit Namen von Unternehmen von persönlichen Datenbanken **1058** und/oder Bereichs-Entitäts-Datenbanken **2772**. Mehrere Quellen von Kontext können die Menge an passenden „Herbs“ beschränken, und/oder diese einstufen und filtern in Schritt **232**. Beispielsweise:

- Andere Anwendungspräferenzen und Verwendungshistorie **1072**, wie ein Herb, der auf einer Favoriten-Telefonnummernliste ist, oder kürzlich angerufen wurde, oder kürzlich Beteiligter bei einer Textnachrichten-Unterhaltung oder E-Mail-Thread gewesen ist;
- Herb erwähnt in persönlichen Datenbanken **1058**, wie ein Herb, der als Beziehung bezeichnet ist, wie Vater oder Bruder, oder in einem kürzlichen Kalenderereignis als Beteiligter gelistet ist. Wäre der Task ein Abspielen von Medien anstelle eines Telefonanrufs, würden die Namen aus Medien-Titeln, Erzeugern und Ähnliches Quellen für Einschränkung darstellen;
- eine kürzliche Lage (ply) eines Dialogs **1052**, entweder als Anfrage oder Ergebnisse. Wie beispielsweise oben in Zusammenhang mit **Fig. 25A** und **Fig. 25B** beschrieben, kann der Benutzer, nachdem er nach einer E-Mail von John gesucht hat, eine Antwort erstellen, mit dem Suchergebnis noch im Dialog-Kontext. Der Assistent **1002** kann den Dialog-Kontext verwenden, um den spezifischen Anwendungsbereichs-Objekt-Kontext zu identifizieren.

**[0182]** Kontext **1000** kann auch helfen, die Mehrdeutigkeit bei anderen Wörtern als richtigen Namen zu reduzieren. Sagt der Benutzer einer E-Mail-Anwendung dem Assistenten **1002** beispielsweise zu „Antworten“ (wie in **Fig. 20** gezeigt), so hilft der Kontext der Anwendung, um zu bestimmen, dass das Wort mit E-MailErwiderung (E-MailReply) anstelle von Text-NachrichtenErwiderung (TextMessagingReply) assoziiert werden soll.

**[0183]** Bei Schritt **232** filtert und sortiert **232** der Sprachinterpretierer **2770** die oberen semantischen Parses als die Darstellung von Benutzerabsicht **290**. Kontext **1000** kann verwendet werden, um solches Filtern und Sortieren **232** anzuregen. Das Ergebnis ist eine Darstellung von Benutzerabsicht **290**.

#### Verwendung von Kontext in Task-Fluss-Verarbeitung

**[0184]** In **Fig. 5**, auf die nun Bezug genommen wird, ist gemäß einer Ausführungsform ein Flussdiagramm dargestellt, welches ein Verfahren zur Verwendung von Kontext bei Task-Fluss-Verarbeitung zeigt, wie es von Dialog-Fluss-Prozessor **2780** ausgeführt werden kann. Bei der Task-Fluss-Verarbeitung werden Kandidaten-Parses, die von dem Verfahren von **Fig. 4** generiert wurden, eingestuft und instanziert, um operationale Task-Beschreibungen, die ausgeführt werden können, zu erzeugen.

**[0185]** Das Verfahren beginnt **300**. Mehrere Kandidatendarstellungen von Benutzerabsicht **290** werden empfangen. Wie im Zusammenhang mit **Fig. 4** beschrieben, enthalten bei einer Ausführungsform Darstellungen von Benutzerabsicht **290** eine Menge von semantischen Parses.

**[0186]** Bei Schritt **312** bestimmt Dialog-Fluss-Prozessor **2780** die bevorzugte Interpretation des/der semantischen Parse(s) mit anderer Information, um einen auszuführenden Task und seine Parameter zu bestimmen, basierend auf einer Bestimmung der Benutzerabsicht. Information kann beispielsweise aus Bereichs-Modellen **2756**, Task-Fluss-Modellen **2786** und/oder Dialog-Fluss-Modellen **2787** oder jeglicher Kombination hieraus erhalten werden. Beispielsweise könnte ein Task TelefonAnruf (PhoneCall) sein, und ein Task-Parameter ist die anzurufende Telefon-Nummer (PhoneNumber).

**[0187]** Bei einer Ausführungsform wird Kontext **1000** beim Ausführen von Schritt **312** verwendet, um das Anbinden von Parametern **312** zu führen, durch Erschließen von Vorgabewerten und Auflösen von Mehrdeutigkeit. Beispielsweise kann Kontext **1000** die Instanziierung der Task-Beschreibungen führen und bestimmen, ob es eine beste Interpretation der Benutzerabsicht gibt.

**[0188]** Beispielsweise angenommen die Absichts-Eingaben **290** sind „Telefon-Anruf (RebeccaRichards)“ („PhoneCall (RebeccaRichards)“) und „TelefonAnruf (HerbGowen)“ („PhoneCall (HerbGowen)“). Der Task Telefon-Anruf erfordert Parameter TelefonNummer. Mehrere Quellen von Kontext **1000** können angewendet werden, um zu bestimmen, welche Telefonnummer für Rebecca und Herb funktionieren würde. In diesem Beispiel hat der Adressbucheintrag für Rebecca in einer Kontaktdatenbank zwei Telefonnummern, und der Eintrag für Herb hat keine Telefonnummern, jedoch eine E-Mail-Adresse. Die Verwendung von Kontextinformation **1000** aus persönlichen Datenbanken **1058**, wie die Kontaktdatenbank, ermöglicht dem virtuellen Assistenten **1002**, Rebecca gegenüber Herb zu bevorzugen, da es eine Telefonnummer für Rebecca gibt, und keine für Herb. Um zu bestimmen, welche Telefonnummer für Rebecca zu verwenden ist, kann Anwendungskontext **1060** konsultiert werden, um die Nummer auszuwählen, die beim Ausführen von Textnachrichtenunterhaltung mit Rebecca verwendet wird. Der virtuelle Assistent **1002** kann folglich bestimmen, dass „Rufe sie an“ („call her“) im Zusammenhang mit einer Textnachrichtenunterhaltung mit Rebecca Richards bedeutet, führe einen Telefonanruf zu dem Mobiltelefon, welches Rebecca für den Textnachrichtendienst verwendet, durch. Diese spezifische Information wird in Schritt **390** zurückgegeben.

**[0189]** Kontext **1000** kann für mehr als das Reduzieren von Telefonnummer-Mehrdeutigkeit verwendet werden. Er kann immer verwendet werden, wenn es mehrere mögliche Werte für einen Task-Parameter gibt, solange eine Quelle von Kontext **1000**, die Werte für diesen Parameter aufweist, verfügbar ist. Andere Beispiele, bei denen Kontext **1000** die Mehrdeutigkeit reduzieren kann (und verhindert, den Benutzer aufzufordern, zwischen Kandidaten zu wählen) beinhalten, ohne Beschränkung: E-Mail-Adressen; physikalische Adressen; Zeit und Daten; Orte; Namen von Listen; Titel von Medien; Künstlernamen; Geschäftsnamen; oder jeglicher anderer Wertebereich.

**[0190]** Andere Arten von Rückschlüssen, die zur Task-Fluss-Verarbeitung **300** erforderlich sind, können auch von Kontext **1000** profitieren. Beispielsweise kann ein voreingestellter Rückschlusswert den/die aktuelle/aktuellen Zeit, Ort und andere aktuelle Werte verwenden. Der voreingestellte Rückschlusswert ist hilfreich, um die Werte von Task-Parametern, die implizit in der Benutzeranfrage sind, zu bestimmen. Falls beispielsweise jemand sagt „Wie ist das Wetter?“, meint er/sie implizit, wie das Wetter hier in der Gegend ist.

**[0191]** Bei Schritt **310** bestimmt der Dialog-Fluss-Prozessor **2780**, ob diese Interpretation von Benutzerabsicht stark genug unterstützt wird, um fortzufah-

ren, und/oder ob sie besser als alternative, mehrdeutige Parses unterstützt wird. Falls es konkurrierende Mehrdeutigkeiten oder ausreichende Unsicherheit gibt, wird Schritt **322** ausgeführt, um den Dialog-Flussschritt einzustellen, sodass die Ausführungsphase den Dialog veranlasst, eine Aufforderung für weitere Information von dem Benutzer auszugeben. Ein Beispiel einer Bildschirmkopie zur Aufforderung des Benutzers, eine Mehrdeutigkeit aufzulösen, ist in [Fig. 14](#) dargestellt. Kontext **1000** kann in Schritt **322** verwendet werden, zum Sortieren und Kommentieren des angezeigten Menüs von Kandidatenelementen für den Benutzer zur Auswahl hieraus.

**[0192]** Bei Schritt **320** wird das Task-Fluss-Modell konsultiert, um einen angemessenen nächsten Schritt zu bestimmen. Information kann beispielweise aus Bereichs-Modellen **2756**, Task-Fluss-Modellen **2786** und/oder Dialog-Fluss-Modellen **2787** oder jeglicher Kombination hieraus erhalten werden.

**[0193]** Das Ergebnis von Schritt **320** oder Schritt **322** ist eine Darstellung der Benutzeranfrage **390**, die die für den Dialog-Fluss-Prozessor **2780** und Dienst-Orchestrierung **2782** ausreichenden Task-Parameter enthält, um den angemessenen Dienst zu entsenden.

#### Verwendung von Kontext um Dialog-Generierung zu verbessern

**[0194]** Während Dialog-Antwort-Generierung **500** kann der Assistent **1002** sein Verständnis der Benutzerabsicht und wie dieses in einem Task operationalisiert wird zurückparaphrasieren. Ein Beispiel einer solchen Ausgabe ist „Okay, ich werde Rebecca auf ihrer Mobilnummer... anrufen“. Dies ermöglicht dem Benutzer, den Assistenten **1002** zu autorisieren, die assoziierte Task-Automatisierung, wie Durchführen eines Anrufes, auszuführen. Bei Dialog-Generierungs-Schritt **500** bestimmt der Assistent **1002**, wie viele Einzelheiten dem Benutzer beim Paraphrasieren des Verständnisses der Benutzerabsicht zurück zu übertragen sind.

**[0195]** Bei einer Ausführungsform kann Kontext **1000** auch verwendet werden, um das angemessene Niveau von Einzelheiten im Dialog auszuwählen, wie auch, um basierend auf vorangegangener Ausgabe zu filtern (um somit zu verhindern, dass Information wiederholt wird). Beispielsweise kann der Assistent **1002** das Wissen, dass die Person und Telefonnummer aus Kontext **1000** erschlossen wurden, verwenden, um zu bestimmen, ob der Name und die Telefonnummer zu erwähnen sind, und mit welchem DetAILlierungsgrad. Beispiele von Regeln, die angewendet werden können, beinhalten ohne Beschränkung:

- Wenn ein Pronomen durch Kontext aufgelöst wird, erwähne die anzurufende Person mit Namen.
- Wenn eine Person aus bekanntem Kontext, wie Texnachrichtendienst, erschlossen wird, verwenden den Vornamen.
- Wenn eine Telefonnummer aus Anwendungs- oder persönlichem Datenkontext erschlossen wird, verwende den symbolischen Namen der Telefonnummer, wie „Mobiltelefon“, anstelle der eigentlichen zu wählenden Nummer.

**[0196]** Zusätzlich zum Anleiten des angemessenen Detaillierungsgrads, kann Kontext **1000** auch bei Dialog-Generierungs-Schritt **500** verwendet werden, um beispielsweise vorangegangene Aussagen zu filtern, um so eine Wiederholung zu vermeiden, und auf früher erwähnte Entitäten in der Unterhaltung Bezug zu nehmen.

**[0197]** Ein Fachmann wird erkennen, dass Kontext **1000** auch auf andere Weise verwendet werden kann. Zum Beispiel kann Kontext **1000** im Zusammenhang mit den hierin beschriebenen Techniken gemäß Mechanismen verwendet werden, wie sie in der verwandten US-Utility Application Anmeldenummer 12/479,477 betreffend „Contextual Voice Commands“ („Interaktive Sprachbefehle“), Aktenzeichen des Vertreters P7393US1, eingereicht am 5. Juni 2009, deren gesamte Offenbarung hierin mittels Bezugnahme eingebunden ist, beschrieben sind.

#### Sammeln von Kontext und Kommunikationsmechanismen

**[0198]** Bei verschiedenen Ausführungsformen werden unterschiedliche Mechanismen zum Sammeln und Kommunizieren von Kontextinformation in dem virtuellen Assistenten **1002** verwendet. Beispielsweise ist bei einer Ausführungsform der virtuelle Assistent **1002** in einer Client-/Serverumgebung implementiert, so dass seine Dienste zwischen dem Client und dem Server verteilt sind, und Quellen von Kontext **1000** auch verteilt sein können.

**[0199]** In [Fig. 6](#), auf die nun Bezug genommen wird, ist ein Beispiel einer Verteilung von Quellen von Kontext **1000** zwischen Client **1304** und Server **1340** gemäß einer Ausführungsform dargestellt. Die Client-Vorrichtung **1304**, die eine mobile Rechenvorrichtung oder andere Vorrichtung sein kann, kann die Quelle von Kontextinformation **1000**, wie Vorrichtungssensordaten **1056**, aktueller Anwendungskontext **1060**, Ereigniskontext **2706** und Ähnliches sein. Andere Quellen von Kontext **1000** können verteilt auf Client **1304** oder Server **1340** oder einer Kombination von beiden sein. Beispiele beinhalten Anwendungspräferenzen und Verwendungshistorie **1072c**, **1072s**; Dialoghistorie und Assistenzspeicher **1052c**, **1052s**; persönliche Datenbanken **1058c**, **1058s**; und persön-

liche akustische Kontextdaten **1080c**, **1080s**. In jedem dieser Beispiele können Quellen von Kontext **1000** auf Server **1340**, auf Client **1304**, oder auf beiden vorliegen. Weiterhin können, wie vorangehend beschrieben, die verschiedenen in [Fig. 2](#) dargestellten Schritte von Client **1304** oder Server **1340** oder einer Kombination von beiden ausgeführt werden.

**[0200]** Bei einer Ausführungsform kann Kontext **1000** zwischen verteilten Komponenten, wie Client **1304** und Server **1340**, kommuniziert werden. Solche Kommunikation kann über eine lokale API oder über ein verteiltes Netzwerk, oder durch andere Mittel, stattfinden.

**[0201]** In [Fig. 7a](#) bis [Fig. 7d](#), auf die nun Bezug genommen wird, sind Ereignisdiagramme dargestellt, die Beispiele von Mechanismen zum Erhalten und Koordinieren von Kontextinformation **1000** gemäß verschiedener Ausführungsformen zeigen. Verschiedene Techniken existieren zum Laden oder Kommunizieren von Kontext, so dass dieser dem virtuellen Assistenten **1002** zur Verfügung steht, wenn benötigt oder hilfreich. Jeder dieser Mechanismen ist hinsichtlich vier Ereignissen, die beim Betrieb des virtuellen Assistenten **1002** auftreten können, beschrieben: Initialisierung von Vorrichtung oder Anwendung **601**; ursprüngliche Benutzereingabe **602**; ursprüngliche Eingabeverarbeitung **603** und kontextabhängiges Verarbeiten **604**.

**[0202]** [Fig. 7a](#) stellt einen Ansatz dar, bei welchem Kontextinformation **1000** unter Verwendung eines „Pull“-Mechanismus geladen wird, sobald Benutzereingabe begonnen hat **602**. Sobald der Benutzer den virtuellen Assistenten **1002** aufruft und zumindest irgendeine Eingabe **602** bereitstellt, lädt **610** der virtuelle Assistent **1002** Kontext **1000**. Das Laden **610** kann durch Anfordern oder Abrufen von Kontextinformation **1000** aus einer angemessenen Quelle durchgeführt werden. Eingabeverarbeitung **603** beginnt, sobald Kontext **1000** geladen worden ist **610**.

**[0203]** [Fig. 7b](#) stellt einen Ansatz dar, bei welchem irgendeine Kontextinformation **1000** geladen wird **620**, wenn eine Vorrichtung oder Anwendung initialisiert wird **601**; zusätzliche Kontextinformation **1000** wird unter Verwendung eines „Pull“-Mechanismus geladen, sobald Benutzereingabe begonnen hat **602**. Bei einer Ausführungsform kann Kontextinformation **1000**, die beim Initialisieren geladen wird **620**, statischen Kontext (d. h. Kontext, der sich nicht häufig ändert) beinhalten; Kontextinformation **1000**, die geladen wird **621**, sobald Benutzereingabe beginnt **602**, enthält dynamischen Kontext (d. h. Kontext, der sich geändert haben kann, seitdem der statische Kontext geladen wurde **620**). Ein solcher Ansatz kann Leistung verbessern, indem die Kosten für das Laden statischer Kontextinformation **1000** von der Laufzeit-(runtime)-Leistung des Systems entfernt werden.

[0204] [Fig. 7c](#) stellt eine Variation des Ansatzes von [Fig. 7b](#) dar. In diesem Beispiel darf die dynamische Kontextinformation **1000** weiterhin geladen werden **621**, nachdem Verarbeitung der Eingabe begonnen hat **603**. Folglich kann das Laden **621** parallel zur Eingabeverarbeitung stattfinden. Das Verfahren des virtuellen Assistenten **1002** ist nur bei Schritt **604** blockiert, wenn die Verarbeitung von empfangener Kontextinformation **1000** abhängt.

[0205] [Fig. 7d](#) stellt eine vollständig konfigurierbare Version, welche Kontext in jeglicher von bis zu fünf unterschiedlichen Weisen bearbeitet, dar:

- Statistische Kontextinformation **1000** wird synchronisiert **640** in einer Richtung, von Kontextquelle zur Umgebung oder Vorrichtung, auf welcher der virtuelle Assistent **1002** läuft. Da sich Daten in der Kontextquelle verändern, werden die Änderungen zu dem virtuellen Assistenten **1002** geschoben (pushed). Beispielsweise könnte ein Adressbuch mit dem virtuellen Assistenten **1002** synchronisiert werden, wenn es anfänglich erstellt oder aktiviert wird. Wann immer das Adressbuch modifiziert wird, werden Änderungen zu dem virtuellen Assistenten **1002** geschoben, entweder sofort oder in einer in Schüben angeordneten (batched) Weise. Wie in [Fig. 7d](#) dargestellt, kann eine solche Synchronisierung **640** zu jeder Zeit, einschließlich bevor Benutzereingabe beginnt **602**, stattfinden.
- Bei einer Ausführungsform können statische Kontextquellen auf einen Synchronisierungsstatus geprüft werden, wenn Benutzereingabe beginnt **602**. Falls nötig, wird ein Prozess zum Synchronisieren verbleibender statischer Kontextinformation **1000** begonnen **641**.
- Wenn Benutzereingabe beginnt **602**, wird irgendein dynamischer Kontext **1000** geladen **642**, wie es der Fall war bei **610** und **621**. Prozeduren, die Kontext **1000** brauchen, werden nur blockiert, um auf die bislang ungeladene Kontextinformation **1000**, die sie benötigen, zu warten.
- Andere Kontextinformation **1000** wird durch Prozesse auf Abruf **643** geladen, wenn diese sie benötigen.
- Ereigniskontext **2706** wird von einer Quelle zu der Vorrichtung, auf welcher der virtuelle Assistent **1002** läuft, gesendet **644**, wenn Ereignisse auftreten. Prozesse, die Ereigniskontext **2706** brauchen, warten nur darauf, dass der Cache von Ereignissen bereit ist und können ohne zu blockieren zu jeder Zeit danach fortfahren. Ereigniskontext **2706**, der auf diese Weise geladen wird, kann eine oder mehrere der folgenden beinhalten:
  - Ereigniskontext **2706**, der geladen wird, bevor die Benutzereingabe beginnt **602**, z. B. Mitteilungen ungelesener Nachrichten. Solche Information kann beispielsweise unter Verwendung eines synchronisierten Cache gehalten werden.

• Ereigniskontext **2706**, der gleichzeitig mit oder nachdem die Benutzereingabe begonnen hat **602**, geladen wird. Zum Beispiel kann eine Textnachricht ankommen, während der Benutzer mit dem virtuellen Assistenten **1002** interagiert; der Ereigniskontext, der den Assistenten **1002** dieses Ereignis mitteilt, kann parallel mit der Verarbeitung des Assistenten **1002** geschoben werden.

[0206] Bei einer Ausführungsform wird Flexibilität beim Erhalten und Koordinieren von Kontextinformation **1000** dadurch erreicht, dass für jede Quelle von Kontextinformation **1000** eine Kommunikationsstrategie und eine Zugangs-API, die die Kosten von Kommunikation mit dem Wert der Verfügbarkeit dieser Information bei jeder Anfrage abwägt, vorgegeben werden. Zum Beispiel können Variablen, die für jede Sprache-zu-Text-Anfrage relevant sind, die persönliche akustische Kontextdaten **1080** oder Vorrichtungssensordaten **1056**, die Parameter von Mikrofonen beschreiben, bei jeder Anfrage geladen werden. Solche Kommunikationsstrategien können beispielsweise in einer Konfigurationstabelle festgelegt sein.

[0207] In [Fig. 9](#), auf die nun Bezug genommen wird, ist ein Beispiel einer Konfigurationstabelle **900** gezeigt, die gemäß einer Ausführungsform zum Festlegen von Kommunikations- und Caching-Strategien für verschiedene Quellen von Kontextinformation **1000** verwendet werden kann. Für jede aus einer Reihe von unterschiedlichen Kontextquellen einschließlich Benutzername, Adressbuchnamen, Adressbuchnummern, SMS-Ereigniskontext und Kalenderdatenbank, wird ein bestimmter Typ des Ladens von Kontext für jeden der Schritte von [Fig. 2](#) spezifiziert: ermittle und interpretiere Spracheingabe **100**, interpretiere natürliche Sprache **200**, identifizierte Task **300** und generiere Dialogantwort **500**. Jeder Eintrag in Tabelle **900** gibt eine oder mehrere der folgenden an:

- Sync: Kontextinformation **1000** wird auf der Vorrichtung synchronisiert;
- auf Abruf (on demand): Kontextinformation **1000** wird in Reaktion auf eine Anfrage des virtuellen Assistenten **1002** bereitgestellt;
- Push: Kontextinformation **1000** wird auf die Vorrichtung geschoben (pushed).

[0208] Das vollständig konfigurierbare Verfahren ermöglicht das Verfügbarmachen eines großen Raumes von möglicherweise relevanter Kontextinformation **1000**, um die Interaktion mit natürlicher Sprache zwischen Mensch und Maschine zu optimieren. Anstelle diese gesamte Information die ganze Zeit zu laden, was zu Ineffizienzen führen könnte, wird manche Information sowohl in der Kontextquelle wie auch beim virtuellen Assistenten **1002** gehalten, während andere Information auf Abruf abgefragt wird. Zum Beispiel wird, wie oben beschrieben, Information, wie Namen, die in Echtzeit-Operationen, wie Spracherkennung, verwendet werden, lokal gehal-

ten, während Information, die lediglich von irgendwelchen möglichen Anfragen, wie ein persönlicher Kalender eines Benutzers, verwendet wird, auf Abruf abgefragt. Daten, die nicht antizipiert werden können, wenn ein Benutzer den Assistenten abruft, wie eingehende SMS-Ereignisse, werden geschoben (pushed), wenn sie auftreten.

**[0209]** In [Fig. 10](#), auf die nun Bezug genommen wird, ist ein Ereignisdiagramm **950** dargestellt, welches gemäß einer Ausführungsform ein Beispiel des Zugriffs auf Kontextinformationsquellen, die in [Fig. 9](#) eingerichtet wurden, während der Verarbeitung einer Interaktionssequenz, bei welcher der Assistent **1002** sich in einem Dialog mit einem Benutzer befindet, zeigt.

**[0210]** Die in [Fig. 10](#) gezeigte Sequenz stellt die folgende Interaktionssequenz dar:

- T<sub>1</sub>: Assistent **1002**: „Hallo Steve, was kann ich für dich tun?“
- T<sub>2</sub>: Benutzer: „Wann ist mein nächstes Meeting?“
- T<sub>3</sub>: Assistent **1002**: „Dein nächstes Meeting ist um 1:00 p.m. im Sitzungssaal.“
- T<sub>4</sub>: [Sound einer eingehenden SMS-Nachricht]
- T<sub>5</sub>: Benutzer: „Lese mir diese Nachricht vor.“
- T<sub>6</sub>: Assistent **1002**: „Deine Nachricht von Johnny sagt 'Wie sieht's mit Mittagessen aus?'“
- T<sub>7</sub>: Benutzer: „Sage Johnny, dass ich es heute nicht schaffe.“
- T<sub>8</sub>: Assistent **1002**: „Ok, ich werde es ihm sagen.“

**[0211]** Zur Zeit T<sub>0</sub>, bevor die Interaktion beginnt, wird Benutzername synchronisiert **770** und Adressbuchnamen werden synchronisiert **771**. Diese stellen Beispiele von statischem Kontext dar, der zur Initialisierungszeit geladen wird, wie bei Element **640** bei [Fig. 7d](#) gezeigt. Dies ermöglicht dem Assistenten **1002**, sich an den Benutzer mit seinem Vornamen („Steve“) zu wenden.

**[0212]** Zur Zeit T<sub>1</sub> sind Synchronisierungsschritte **770** und **771** abgeschlossen. Zur Zeit T<sub>2</sub> spricht der Benutzer eine Anfrage, die gemäß Schritte **100**, **200** und **300** von [Fig. 2](#) verarbeitet wird. Bei dem Task-Identifizierungsschritt **300** fragt **774** der virtuelle Assistent **1002** persönliche Datenbanken des Benutzers **1058** als eine Quelle von Kontext **1000** ab: konkret fordert der virtuelle Assistent **1002** Information aus der Kalenderdatenbank des Benutzers, die gemäß Tabelle **900** für Zugriff auf Abruf eingerichtet ist. Zur Zeit T<sub>3</sub> wird Schritt **500** ausgeführt und eine Dialogantwort wird generiert.

**[0213]** Zur Zeit T<sub>4</sub> wird eine SMS-Nachricht erhalten; dies ist ein Beispiel von Ereigniskontext **2706**. Eine Mitteilung des Ereignisses wird zu dem virtuellen Assistenten **1002** geschoben **773**, basierend auf der Konfiguration in Tabelle **900**.

**[0214]** Zur Zeit T<sub>5</sub> bittet der Benutzer den virtuellen Assistenten **1002**, die SMS-Nachricht zu lesen. Das Vorliegen des Ereigniskontexts **2706** leitet die NLP-Komponente beim Ausführen von Schritt **200** an, um so „diese Nachricht“ als eine neue SMS-Nachricht zu interpretieren. Zur Zeit T<sub>6</sub> kann Schritt **300** von der Task-Komponente ausgeführt werden, um eine API aufzurufen, um die SMS-Nachricht dem Benutzer vorzulesen. Zur Zeit T<sub>7</sub> führt der Benutzer eine Anfrage mit einem mehrdeutigen Wert („sage“) und Namen („Johnny“) aus. Die NLP-Komponente interpretiert natürliche Sprache **200**, in dem diese Mehrdeutigkeiten unter Verwendung verschiedener Quellen von Kontext **1000**, einschließlich dem Ereigniskontext **2706**, der in Schritt **773** erhalten wurde, aufgelöst werden; dies sagt der NLP-Komponente, dass sich der Befehl auf eine SMS-Nachricht von einer Person namens Johnny bezieht. Bei Schritt T<sub>7</sub>, wird Ausführungsflussschritt **400** durchgeführt, beinhaltend ein Zuordnen des Namens **771** durch Nachschlagen der zu verwenden Nummer aus dem empfangenen Ereigniskontextobjekt. Der Assistent **1002** ist folglich in der Lage, eine neue SMS-Nachricht zusammen zu stellen, und diese an Johnny zu senden, wie in Schritt T<sub>8</sub> bestätigt.

**[0215]** Die vorliegende Erfindung wurde mit besonderen Einzelheiten im Hinblick auf mögliche Ausführungsformen beschrieben. Die Fachleute werden erkennen, dass die Erfindung auch bei anderen Ausführungsformen ausgeführt werden kann. Zunächst ist die bestimmte Benennung der Komponenten, Großschreibung von Begriffen, die Attribute, Datenstrukturen oder jeglicher anderer Programmieraspekt oder struktureller Aspekt nicht verpflichtend oder wesentlich, und die Mechanismen, die die Erfindung oder ihre Merkmale implementieren, können unterschiedliche Namen, Formate oder Protokolle aufweisen. Darüber hinaus kann das System mittels einer Kombination von Hardware und Software, wie beschrieben, oder vollständig mit Hardware-Elementen oder vollständig mit Software-Elementen implementiert sein. Auch ist die hierin beschriebene bestimmte Aufteilung von Funktionalität zwischen den verschiedenen Systemkomponenten lediglich beispielhaft und nicht zwingend; Funktionen, die von einer einzelnen Systemkomponente ausgeführt werden, können stattdessen von mehreren Komponenten ausgeführt werden, und Funktionen, die von mehreren Komponenten ausgeführt werden, können stattdessen von einer einzelnen Komponente ausgeführt werden.

**[0216]** Bei verschiedenen Ausführungsformen kann die vorliegende Erfindung als ein System oder ein Verfahren zum Ausführen der oben beschriebenen Techniken, entweder einzeln oder in jeglicher Kombination, implementiert sein. Bei einer anderen Ausführungsform kann die vorliegende Erfindung als ein Computerprogrammprodukt, welches ein nichtflüchtiges computerlesbares Speichermedium und auf

dem Medium codierten Computerprogrammcode aufweist, implementiert sein, um einen Prozessor in einer Rechenvorrichtung oder anderen elektronischen Vorrichtung zu veranlassen, die oben beschriebenen Techniken auszuführen.

**[0217]** Bezugnahme in der Spezifikation auf „eine Ausführungsform“ („one embodiment“ oder „an embodiment“) bedeuten, dass ein/eine bestimmte/bestimmtes Merkmal, Struktur oder Charakteristikum, welches/welche im Zusammenhang mit den Ausführungsformen beschrieben ist, bei zumindest einer Ausführungsform der Erfindung enthalten ist. Die Vorkommen der Phrase „bei einer Ausführungsform“ an verschiedenen Stellen in der Spezifikation beziehen sich nicht notwendigerweise alle auf die gleiche Ausführungsform.

**[0218]** Einige Teile des Voranstehenden wurden bezüglich Algorithmen und symbolischen Darstellungen von Operationen auf Datenbits innerhalb eines Speichers einer Rechenvorrichtung präsentiert. Diese algorithmischen Beschreibungen und Darstellungen sind die Mittel, die von den Fachleuten im Bereich der Datenverarbeitung verwendet werden, um die Substanz ihrer Arbeit am effektivsten anderen Fachleuten zu vermitteln. Ein Algorithmus wird hier, und im Allgemeinen, als eine selbstständige Sequenz von Schritten (Instruktionen), die zu einem gewünschten Ergebnis führen, verstanden. Die Schritte sind jene, die physikalische Manipulationen von physikalischen Quantitäten erfordern. Üblicherweise, wenn auch nicht notwendigerweise, nehmen diese Quantitäten die Form von elektrischen, magnetischen oder optischen Signalen an, die geeignet sind, gespeichert, übertragen, kombiniert, verglichen oder anderweitig manipuliert zu werden. Es ist bisweilen passend, vorwiegend aus Gründen gemeinsamer Verwendung, diese Signale als Bits, Werte, Elemente, Symbole, Zeichen (characters), Terme, Nummern oder Ähnliches zu bezeichnen. Darüber hinaus ist es bisweilen auch passend, bestimmte Anordnungen von Schritten, die physikalische Manipulationen von physikalischen Quantitäten erfordern, als Module oder Code-Vorrichtungen zu bezeichnen, ohne Beschränkung der Allgemeinheit.

**[0219]** Es sollte jedoch beachtet werden, dass all diese und ähnliche Begriffe mit den zugehörigen physikalischen Quantitäten verbunden werden müssen und lediglich für diese Quantitäten geeignete und angewandte Kennungen sind. Wenn aus der nachfolgenden Darstellung nichts spezifisch anderes hervorgeht, werden in der gesamten Beschreibung die Darlegungen unter Verwendung von Begriffen wie „Verarbeiten“ oder „Rechnen“ oder „Berechnen“ oder „Anzeigen“ oder „Bestimmen“ oder Ähnliches so verstanden, dass sie sich auf den Vorgang und die Prozesse eines Computersystems oder ähnliches elektronisches Rechenmodul und/oder Vorrichtung beziehen,

welche Daten, die als physikalische (elektronische) Quantitäten innerhalb der Computersystemspeicher oder Register oder anderer solcher Informationsspeicher, Übertragungs- oder Anzeigevorrichtungen manipuliert und transformiert.

**[0220]** Bestimmte Aspekte der vorliegenden Erfindung beinhalten Prozessschritte und Instruktionen, die hierin in der Form eines Algorithmus beschrieben sind. Es soll beachtet werden, dass die Prozessschritte und Instruktionen der vorliegenden Erfindung in Software, Firmware und/oder Hardware eingebettet sein können, und, wenn sie in Software eingebettet sind, heruntergeladen werden können, um auf unterschiedlichen Plattformen, die von einer Vielfalt von Betriebssystemen verwendet werden, abgelegt und betrieben zu werden.

**[0221]** Die vorliegende Erfindung bezieht sich auch auf eine Vorrichtung zur Durchführung der Operationen hierin. Diese Vorrichtung kann spezifisch für die erforderlichen Zwecke konstruiert sein, oder sie kann eine Universalrechenvorrichtung umfassen, die durch ein Computerprogramm, das in der Rechenvorrichtung gespeichert ist, selektiv aktiviert oder rekonfiguriert wird. Solch ein Computerprogramm kann in einem computerlesbaren Speichermedium abgelegt sein, wie z. B., worauf es jedoch nicht beschränkt ist, jeglicher Typ von Diskette, einschließlich Floppy-Disks, optische Disketten, CD-ROMs, magnetisch-optische Disketten, Nur-Lese-Speicher (ROMs), Speicher mit wahlfreiem Zugriff (RAMs), EPROMs, EEPROMs, magnetische oder optische Karten, anwendungsspezifische integrierte Schaltkreise (ASICs) oder jeglicher Typ von Medien, die zum Speichern elektronischer Instruktionen geeignet sind, wobei jedes an einen Computersystembus angeschlossen ist. Weiterhin können die hierin in Bezug genommenen Rechenvorrichtungen einen einzelnen Prozessor beinhalten, oder können Architekturen sein, die Mehr-Prozessor-Designs zur erhöhten Rechenfähigkeit benutzen.

**[0222]** Die hierin präsentierten Algorithmen und Anzeigen sind nicht inhärent bezogen auf eine bestimmte/ein bestimmtes Rechenvorrichtung, virtuelles System oder andere Vorrichtung. Verschiedene Universalsysteme können auch mit Programmen gemäß der Lehre hierin benutzt werden, oder es kann sich als geeignet erweisen, spezialisierte Vorrichtung zum Durchführen der erforderlichen Verfahrensschritte zu konstruieren. Die benötigte Struktur für eine Vielfalt dieser Systeme wird aus der hierin bereitgestellten Beschreibung offensichtlich sein. Darüber hinaus ist die vorliegende Erfindung nicht unter Bezugnahme auf eine bestimmte Programmiersprache beschrieben. Es soll erkannt werden, dass eine Vielfalt von Programmiersprachen verwendet werden kann, um die Lehren der vorliegenden Erfindung, wie hierin beschrieben, zu implementieren, und jeg-

liche vorangegangene Bezüge auf spezifische Sprachen werden zur Offenbarung von Befähigung (enablement) und besten Modus (best mode) der vorliegenden Erfindung bereitgestellt.

**[0223]** Dementsprechend kann die vorliegende Erfindung bei verschiedenen Ausführungsformen mit Software, Hardware und/oder anderen Elementen zur Steuerung eines Computersystems, einer Rechenvorrichtung oder einer anderen elektronischen Vorrichtung oder jeglicher Kombination oder Vielzahl hieraus implementiert sein. Eine solche elektronische Vorrichtung kann beispielsweise einen Prozessor, eine Eingabevorrichtung (wie ein/eine/einen Tastatur, Maus, Berührungspad, Trackpad, Joystick, Trackball, Mikrofon und/oder jegliche Kombination hieraus), eine Ausgabevorrichtung (wie einen Bildschirm, Lautsprecher und/oder Ähnliches), Speicher, Langzeitspeicher (wie magnetischer Speicher, optischer Speicher und/oder Ähnliches) und/oder Netzwerkkonnektivität, gemäß Techniken, die in der Technik wohl bekannt sind, beinhalten. Eine solche elektronische Vorrichtung kann tragbar oder nicht tragbar sein. Beispiele von elektronischen Vorrichtungen, die zum Implementieren der Erfindung verwendet werden können, beinhalten: ein Mobiltelefon, einen persönlichen digitalen Assistenten, Smartphone, Kiosk, Desktop-Computer, Laptop-Computer, Tablet-Computer, elektronische Vorrichtung von Konsumenten, Vorrichtung der Unterhaltungselektronik, Musikspieler, Kamera, Fernseher, Set-Topbox, elektronische Spieleinheit oder Ähnliches. Eine elektronische Vorrichtung zum Implementieren der vorliegenden Erfindung kann jegliches Betriebssystem verwenden, wie beispielsweise iOS oder MacOS, verfügbar von Apple Inc. aus Cupertino, Kalifornien oder jegliches anderes Betriebssystem, das zur Benutzung auf der Vorrichtung angepasst ist. Während die Erfindung bezüglich einer begrenzten Anzahl von Ausführungsformen beschrieben wurde, werden Fachleute, die von der obigen Beschreibung profitieren, erkennen, dass andere Ausführungsformen entwickelt werden können, ohne von dem Bereich der hierin beschriebenen vorliegenden Erfindung abzuweichen. Darüber hinaus sollte beachtet werden, dass die Sprache, die in der Spezifikation verwendet wurde, vorwiegend aus Gründen der Lesbarkeit und für lehrende Zwecke verwendet wurde, und nicht verwendet wurde, um den erforderlichen Gegenstand abzugrenzen oder zu umschreiben. Dementsprechend ist die Offenbarung der vorliegenden Erfindung beabsichtigt, den Bereich der Erfindung, wie hier in den Ansprüchen dargelegt, zu erläutern aber nicht einzuschränken.

### Patentansprüche

1. Computerprogrammprodukt zum Interpretieren einer Benutzereingabe, um einen Task auf einer mindestens einen Prozessor aufweisenden Rechenvorrichtung auszuführen, aufweisend:

ein nicht-flüchtiges computerlesbares Speichermedium; und  
Computerprogrammcode, der auf dem Medium kodiert ist, eingerichtet, um zumindest einen Prozessor zu veranlassen, ein Verfahren auszuführen zum:  
Empfangen von Kontextinformation von einer Kontextquelle;  
Veranlassen einer Ausgabevorrichtung, einen Benutzer zu einer Eingabe aufzufordern;  
Empfangen einer Benutzereingabe über eine Eingabevorrichtung;  
Interpretieren der empfangenen Benutzereingabe, um eine Darstellung der Absicht des Benutzers abzuleiten,  
Identifizieren zumindest eines Tasks und zumindest eines Parameters für den Task, basierend zumindest teilweise auf der abgeleiteten Darstellung der Absicht des Benutzers;  
Ausführen des zumindest einen Tasks unter Verwendung des zumindest einen Parameters, um ein Ergebnis abzuleiten;  
Generieren einer Dialogantwort basierend auf dem abgeleiteten Ergebnis; und  
Veranlassen der Ausgabevorrichtung, die generierte Dialogantwort auszugeben;  
wobei der Computerprogrammcode eingerichtet ist, zumindest einen Prozessor zu veranlassen, zumindest eines der folgenden unter Verwendung der empfangenen Kontextinformation auszuführen:  
Veranlassen der Ausgabevorrichtung den Benutzer zu einer Eingabe aufzufordern, Interpretieren der empfangenen Benutzereingabe, Identifizieren des zumindest einen Tasks und des zumindest einen Parameters für den Task, und Generieren der Dialogantwort.

2. System zum Interpretieren einer Benutzereingabe, um einen Task auszuführen, aufweisend:  
eine Ausgabevorrichtung, eingerichtet um einen Benutzer zu einer Eingabe aufzufordern;  
eine Eingabevorrichtung, eingerichtet um eine Benutzereingabe zu empfangen;  
zumindest einen Prozessor, kommunikativ mit der Ausgabevorrichtung und der Eingabevorrichtung verbunden, eingerichtet die folgenden Schritte auszuführen:  
Empfangen von Kontextinformation von einer Kontextquelle;  
Interpretieren der empfangenen Benutzereingabe, um eine Darstellung der Absicht des Benutzers abzuleiten,  
Identifizieren zumindest eines Tasks und zumindest eines Parameters für den Task, basierend zumindest teilweise auf der abgeleiteten Darstellung der Absicht des Benutzers;  
Ausführen des zumindest einen Tasks unter Verwendung des zumindest einen Parameters, um ein Ergebnis abzuleiten; und  
Generieren einer Dialogantwort basierend auf dem abgeleiteten Ergebnis;

wobei die Rechenvorrichtung weiterhin eingerichtet ist, die generierte Dialogantwort auszugeben; und wobei zumindest eines der folgenden unter Verwendung der empfangenen Kontextinformation ausgeführt wird: Auffordern des Benutzers zu einer Eingabe, Interpretieren der empfangenen Benutzereingabe, Identifizieren des zumindest einen Tasks und des zumindest einen Parameters für den Task und Generieren der Dialogantwort.

3. System nach Anspruch 2, wobei die empfangene Kontextinformation zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:

Anwendungskontext;  
persönliche Daten assoziiert mit dem Benutzer;  
Daten aus einer Datenbank assoziiert mit dem Benutzer;  
Daten erhalten aus Dialoghistorie;  
Daten erhalten von zumindest einem Sensor;  
Anwendungspräferenzen;  
Anwendungsverwendungshistorie;  
Daten, die ein Ereignis beschreiben;  
Daten, die eine akustische Umgebung beschreiben, in der die gesprochene Eingabe empfangen wird; gegenwärtigen Dialogzustand;  
Eingabe, die früher von dem Benutzer bereitgestellt wurde;  
Standort;  
lokale Zeit; und  
Umgebungsbedingungen.

4. System nach Anspruch 2, wobei:  
die Ausgabevorrichtung eingerichtet ist, den Benutzer über eine Dialogschnittstelle aufzufordern; und die Eingabevorrichtung eingerichtet ist, eine Benutzereingabe über die Dialogschnittstelle zu empfangen, indem eine gesprochene Eingabe empfangen wird; und wobei der zumindest eine Prozessor eingerichtet ist, die gesprochene Eingabe in eine Textdarstellung umzuwandeln.

5. System nach Anspruch 4, wobei der zumindest eine Prozessor eingerichtet ist, die gesprochene Eingabe in eine Textdarstellung umzuwandeln, durch: Generieren einer Vielzahl von Kandidatentextinterpretationen der gesprochenen Eingabe; und Einstufen zumindest einer Teilmenge der generierten Kandidatentextinterpretationen; wobei zumindest einer der Schritte der Generierung und Einstufung unter Verwendung der empfangenen Kontextinformation ausgeführt wird.

6. System nach Anspruch 5, wobei die empfangene Kontextinformation, die zumindest bei einem von Generieren und Einstufen verwendet wird, zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:

Daten, die eine akustische Umgebung beschreiben, in der die gesprochene Eingabe empfangen wird; Daten erhalten von zumindest einem Sensor; Vokabular, das aus einer mit dem Benutzer assoziierten Datenbank erhalten wird, Vokabular, das mit den Anwendungspräferenzen assoziiert ist, Vokabular, das von der Verwendungshistorie erhalten wird; und gegenwärtiger Dialogzustand.

7. System nach Anspruch 2, wobei die Ausgabevorrichtung eingerichtet ist, den Benutzer aufzufordern, indem zumindest eine Aufforderung generiert wird, die zumindest teilweise auf der empfangenen Kontextinformation basiert.

8. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, die empfangene Benutzereingabe zu interpretieren, um eine Darstellung der Absicht des Benutzers abzuleiten, indem natürliche Sprachverarbeitung mit der empfangenen Benutzereingabe durchgeführt wird, die zumindest teilweise auf der empfangenen Kontextinformation basiert.

9. System nach Anspruch 8, wobei die empfangene Kontextinformation, die beim Interpretieren der empfangenen Benutzereingabe verwendet wird, zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:  
Daten, die ein Ereignis beschreiben;  
Anwendungskontext;  
Eingabe, die früher von dem Benutzer bereitgestellt wurde;  
bekannte Information über den Benutzer;  
Standort;  
Datum;  
Umgebungsbedingungen; und  
Historie.

10. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, den zumindest einen Task und den zumindest einen Parameter für den Task zu identifizieren, indem der zumindest eine Task und der zumindest eine Parameter für den Task zumindest teilweise basierend auf der empfangenen Kontextinformation identifiziert wird.

11. System nach Anspruch 10, wobei die empfangene Kontextinformation, die beim Identifizieren des zumindest einen Tasks und des zumindest einen Parameters für den Task verwendet wird, zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:  
Daten, die ein Ereignis beschreiben;  
Daten aus einer Datenbank assoziiert mit dem Benutzer;  
Daten erhalten von zumindest einem Sensor;  
Anwendungskontext;

Eingabe, die früher von dem Benutzer bereitgestellt wurde;  
bekannte Information über den Benutzer;  
Standort;  
Datum;  
Umgebungsbedingungen; und  
Historie.

12. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, eine Dialogantwort zu generieren, durch Generieren einer Dialogantwort basierend zumindest teilweise auf der empfangenen Kontextinformation.

13. System nach Anspruch 12, wobei die empfangene Kontextinformation, die beim Generieren einer Dialogantwort verwendet wird, zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:  
Daten aus einer Datenbank assoziiert mit dem Benutzer;  
Anwendungskontext;  
Eingabe, die früher von dem Benutzer bereitgestellt wurde;  
bekannte Information über den Benutzer;  
Standort;  
Datum;  
Umgebungsbedingungen; und  
Historie.

14. System nach Anspruch 2, wobei die empfangene Kontextinformation zumindest eine Information aufweist, die aus der Gruppe ausgewählt ist, welche besteht aus:  
Kontextinformation, die auf einem Server gespeichert wird; und  
Kontextinformation, die auf einem Client gespeichert wird.

15. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, Kontextinformation von einer Kontextquelle zu erhalten, durch:  
Anfragen der Kontextinformation von einer Kontextquelle; und  
Empfangen der Kontextinformation in Reaktion auf die Anfrage.

16. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, Kontextinformation von einer Kontextquelle zu erhalten, durch:  
Empfangen von zumindest einem Teil der Kontextinformation vor dem Empfang einer Benutzereingabe.

17. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, Kontextinformation von einer Kontextquelle zu erhalten, durch: Empfangen von zumindest einem Teil der Kontextinformation nach dem Empfang einer Benutzereingabe.

18. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, Kontextinformation von einer Kontextquelle zu erhalten, durch:  
Empfangen von statischer Kontextinformation als Teil eines Initialisierungsschrittes; und  
Empfangen von zusätzlicher Kontextinformation nach dem Empfang einer Benutzereingabe.

19. System nach Anspruch 2, wobei der zumindest eine Prozessor eingerichtet ist, Kontextinformation von einer Kontextquelle zu erhalten, durch:  
Empfangen einer Push-Benachrichtigung einer Änderung in der Kontextinformation; und  
reagierend auf die Push-Benachrichtigung, Aktualisieren der lokal gespeicherten Kontextinformation.

20. System nach Anspruch 2, wobei die Ausgabevorrichtung, die Eingabevorrichtung und der zumindest eine Prozessor als Komponenten implementiert sind von zumindest einer Vorrichtung, die aus der Gruppe ausgewählt ist, welche besteht aus:  
ein Telefon;  
ein Smartphone;  
ein Tablet-Computer;  
ein Laptop-Computer;  
ein persönlich digitaler Assistent;  
ein Desktop-Computer;  
ein Kiosksystem;  
eine elektronische Vorrichtung für Konsumenten;  
eine elektronische Unterhaltungsvorrichtung für Konsumenten;  
ein Musikspieler;  
eine Kamera;  
ein Fernseher;  
eine elektronische Spieleinheit; und  
eine Set-Top-Box.

Es folgen 32 Blatt Zeichnungen

## Anhängende Zeichnungen

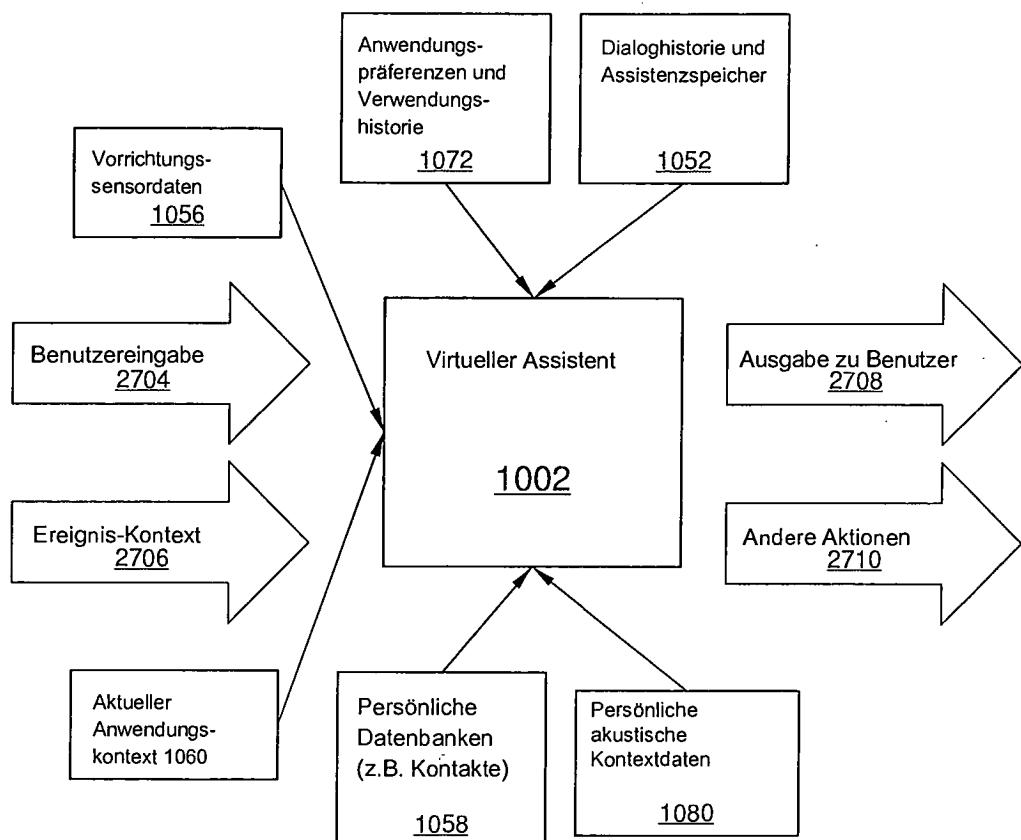
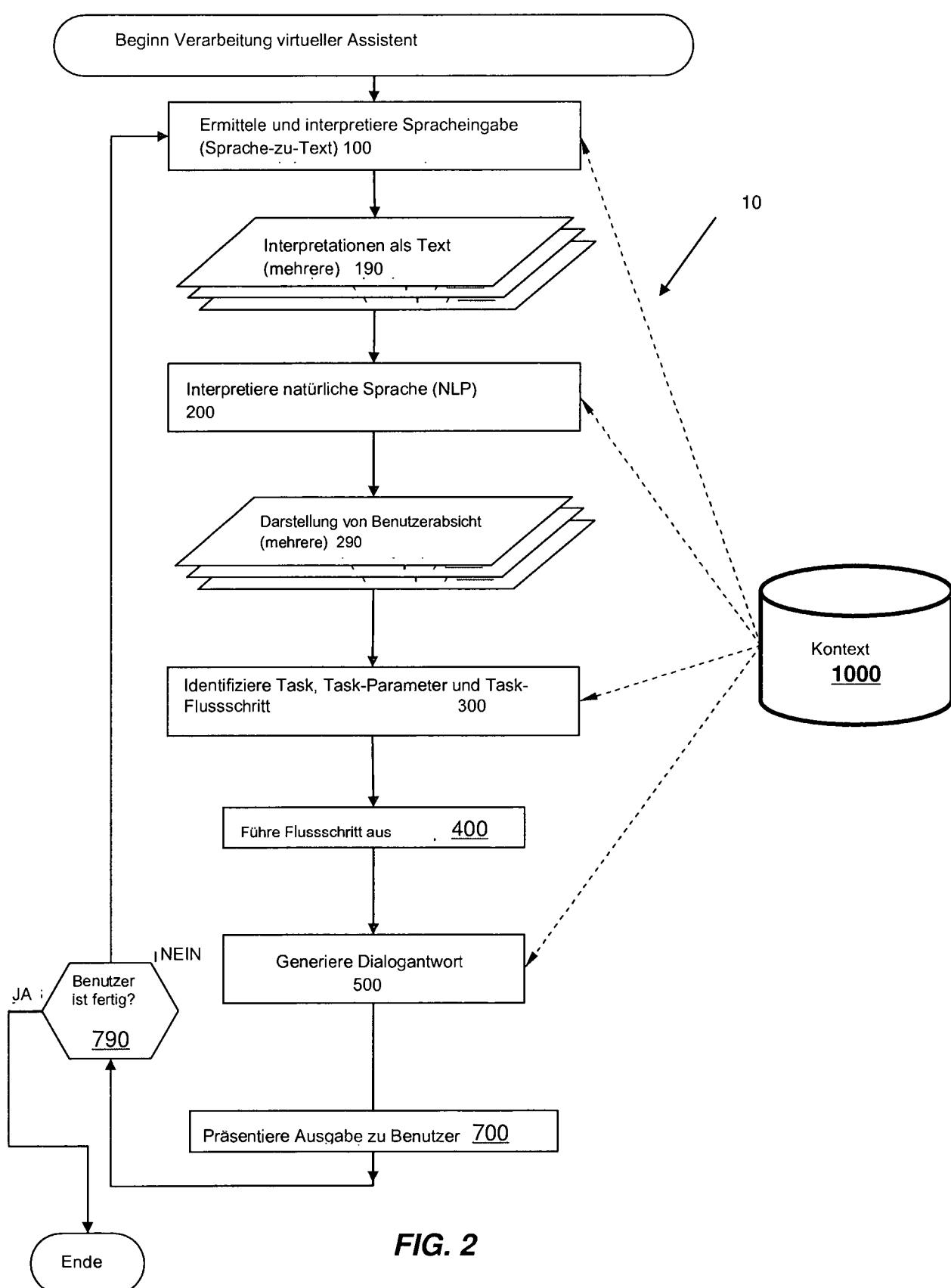


FIG. 1



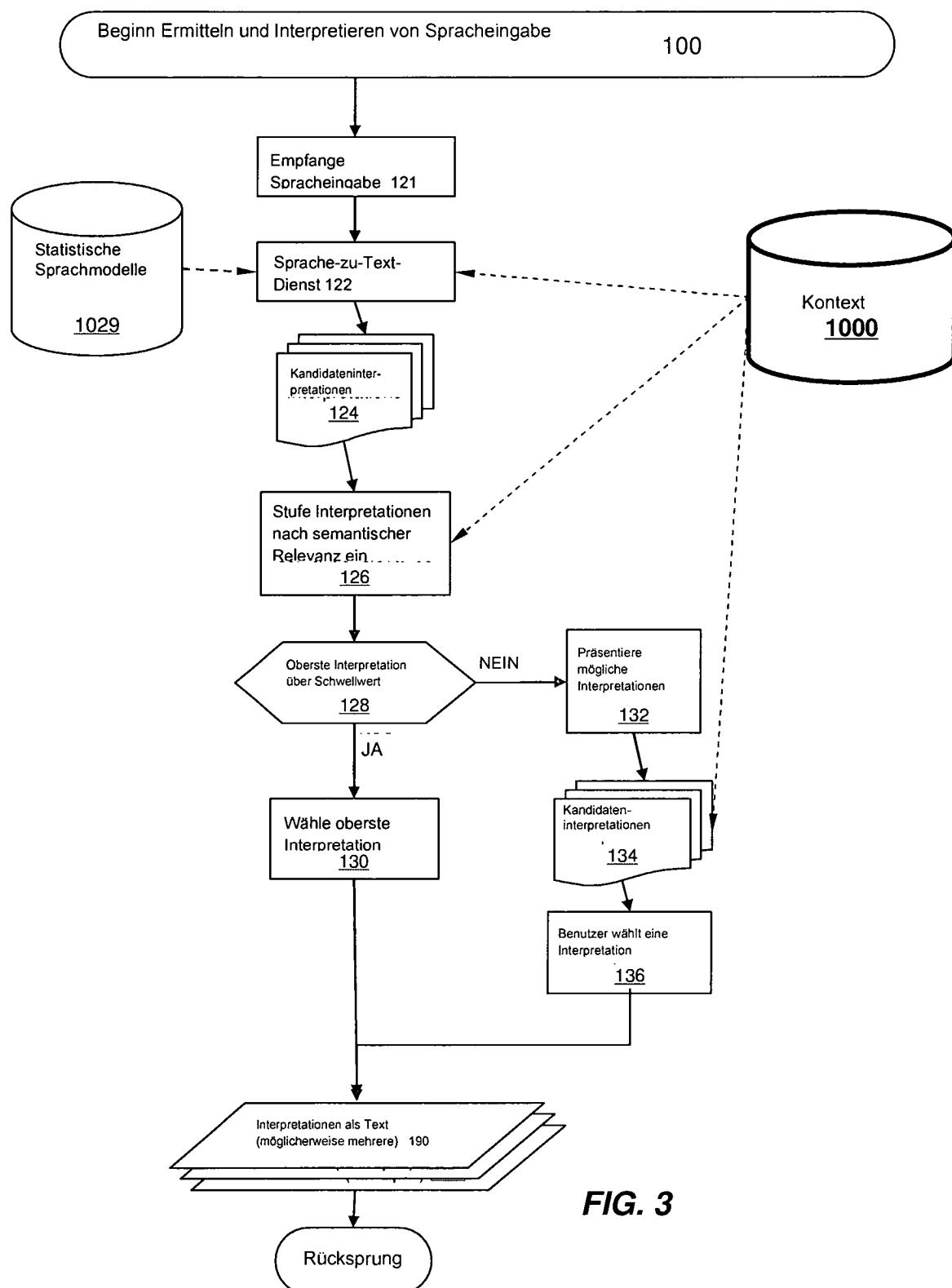
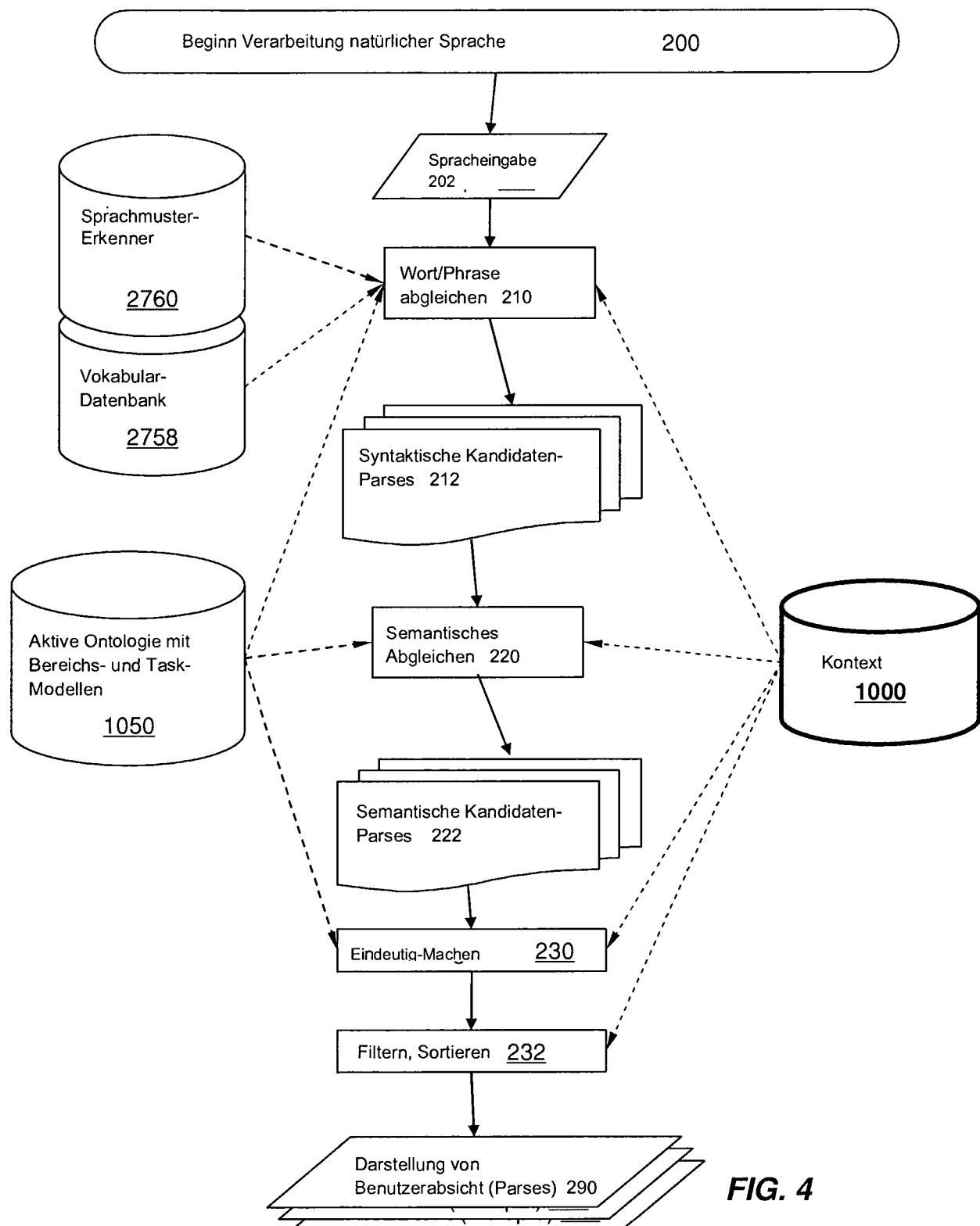


FIG. 3

**FIG. 4**

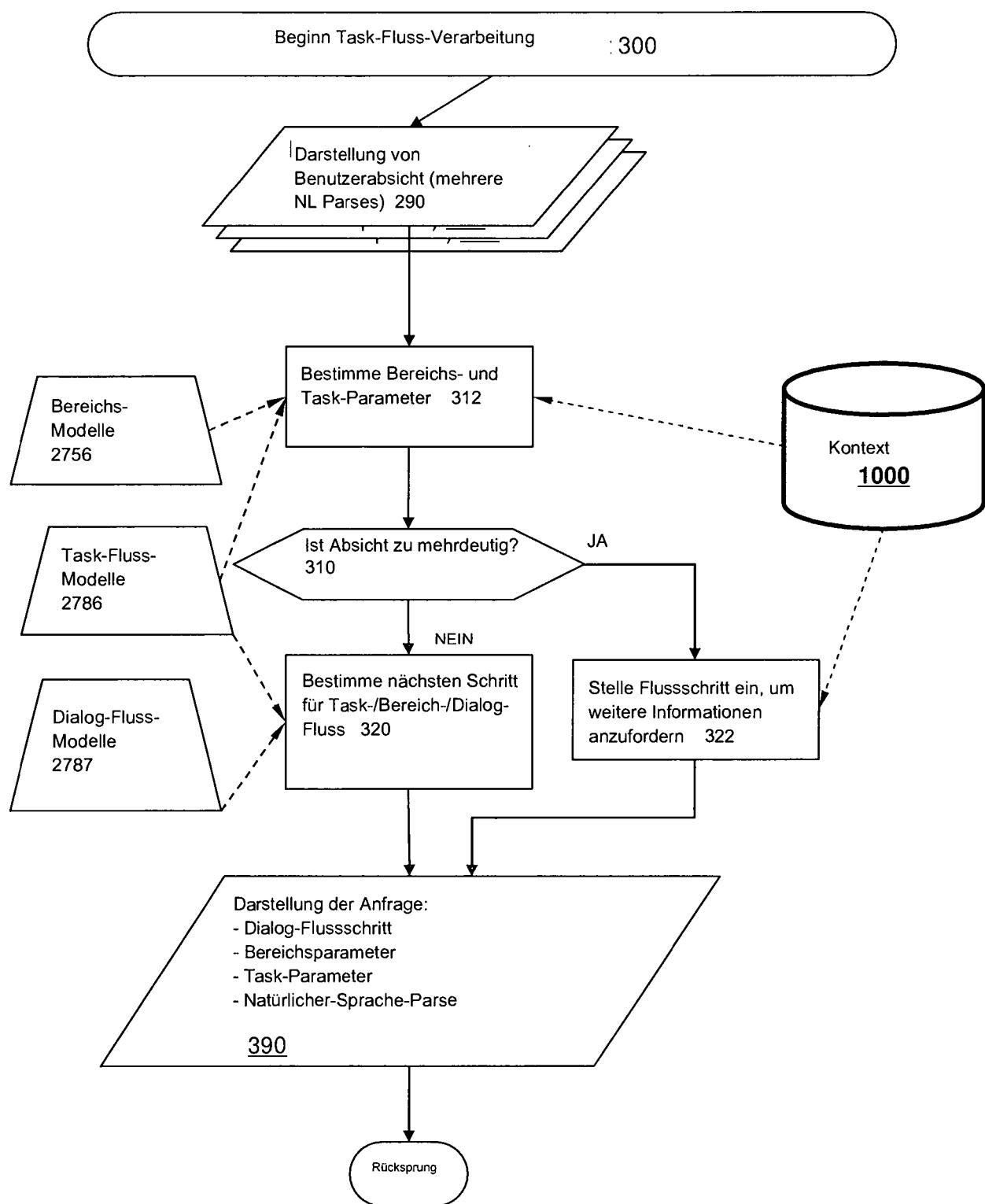
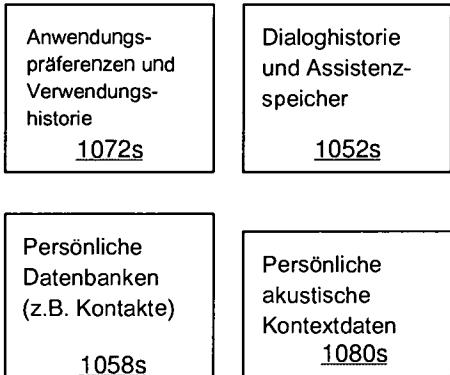


FIG. 5

# Server

1340

Server-Cache oder Master-Version von:

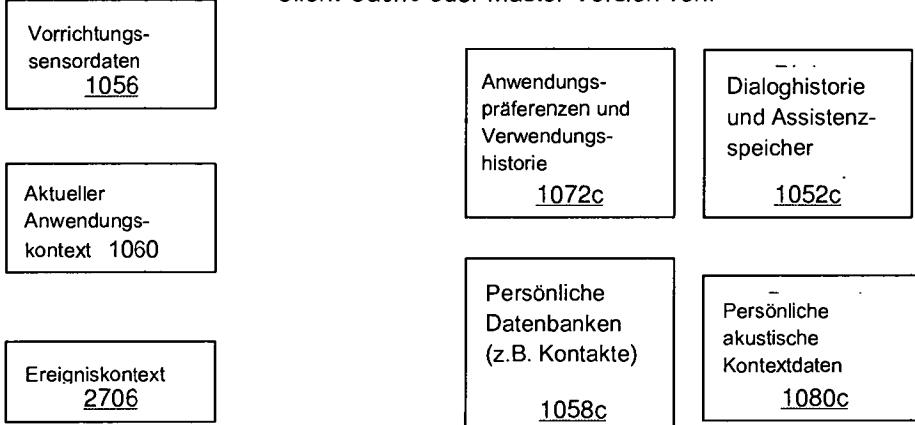


Client/Server-Kommunikation und/oder Synchronisation

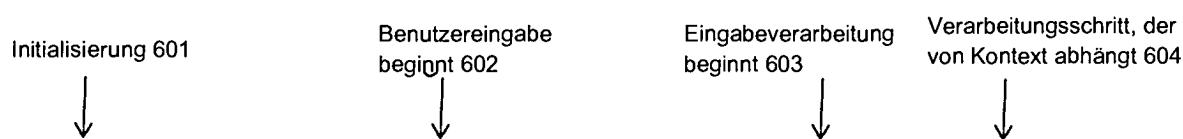
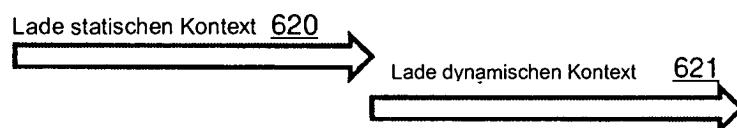
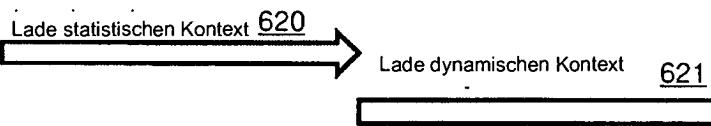
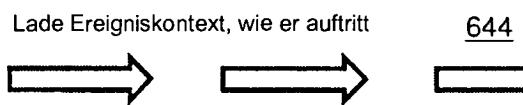
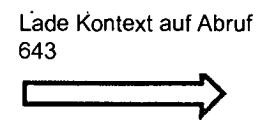
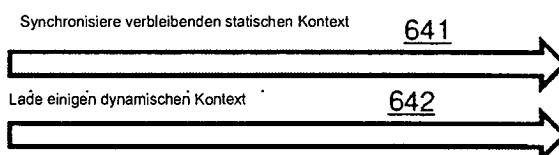
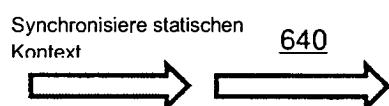
# Client / Vorrichtung

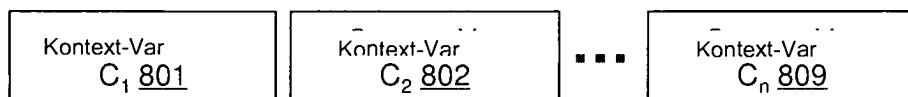
1304

Client-Cache oder Master-Version von:

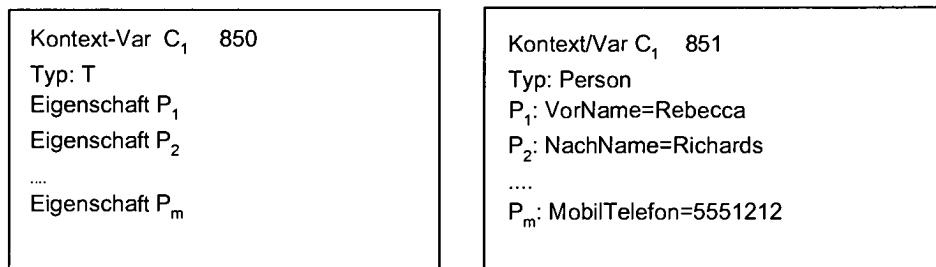
**FIG. 6**

## Kontext-Kommunikation und Koordination

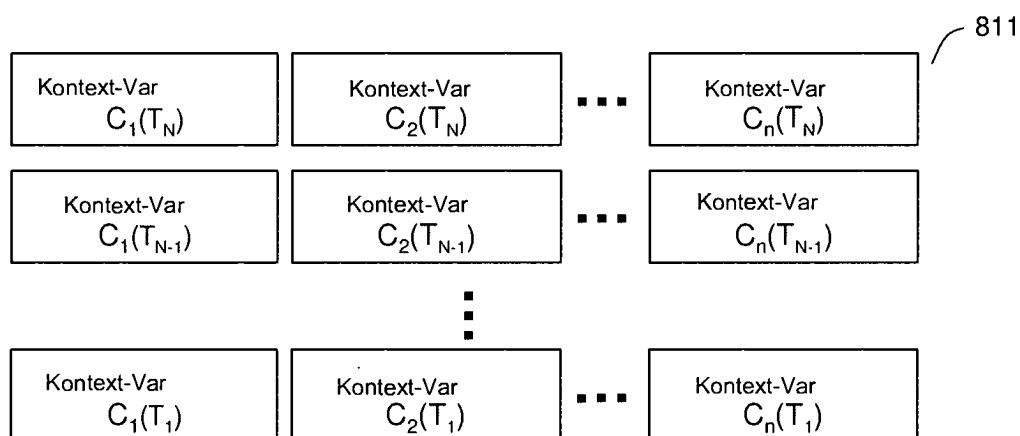
**FIG. 7a****FIG. 7b****FIG. 7c****FIG. 7d**



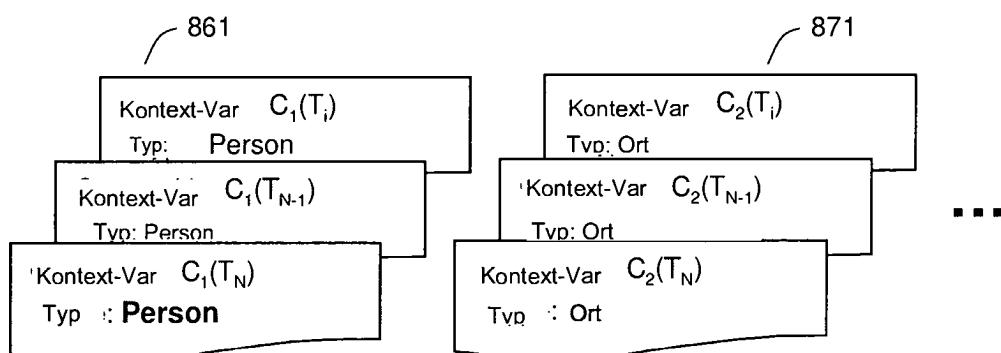
**FIG. 8a**



**FIG. 8b**



**FIG. 8c**



**FIG. 8d**

900

Kontextquelle	Sprache <u>100</u>	NLP <u>200</u>	Task <u>300</u>	Dialog <u>500</u>
Benutzername	Sync	Sync	Sync	Sync
Adressbuch - Namen	Sync	Sync	Sync	-
Adressbuch - Nummern	-	-	auf Abruf (on demand)	-
SMS-Ereignis-Kontext	-	Push	Push	-
Kalenderdatenbank	-	auf Abruf (on demand)	auf Abruf (on demand)	-

FIG. 9

950

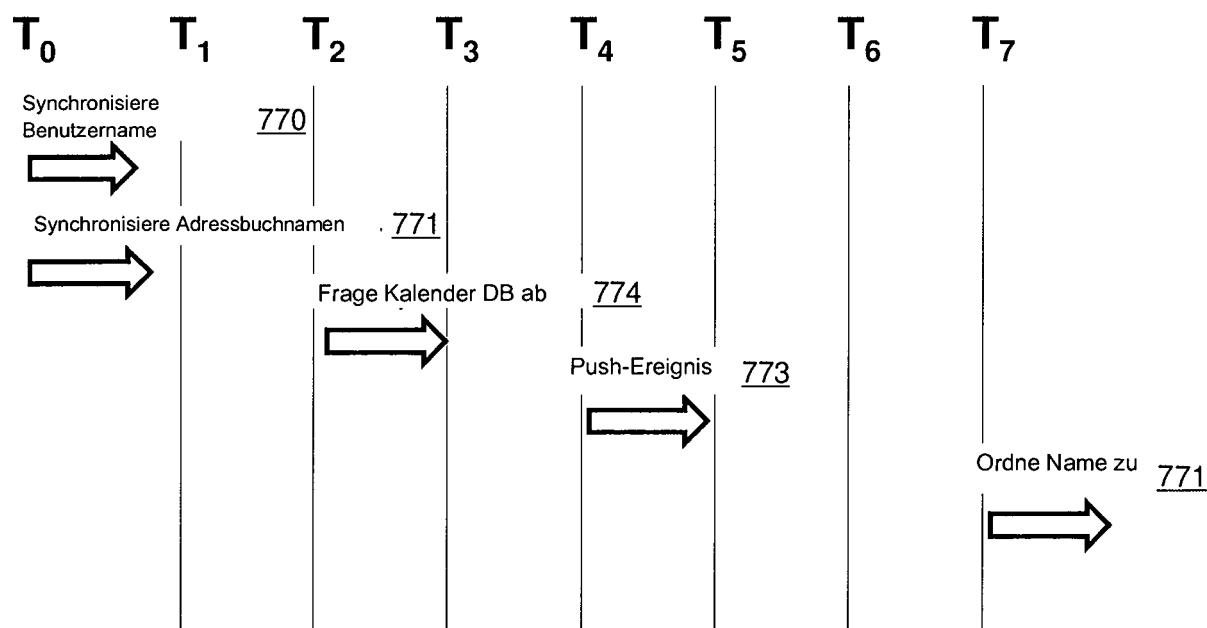
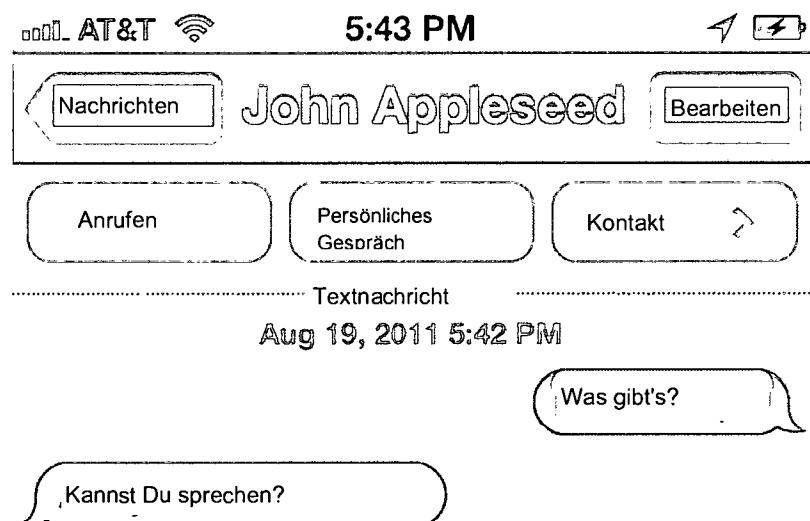


FIG. 10

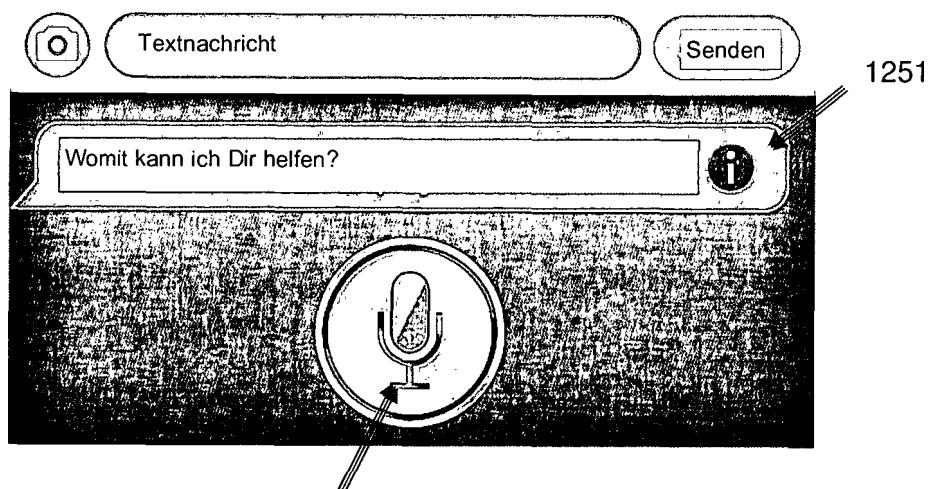


1150

**FIG. 11**

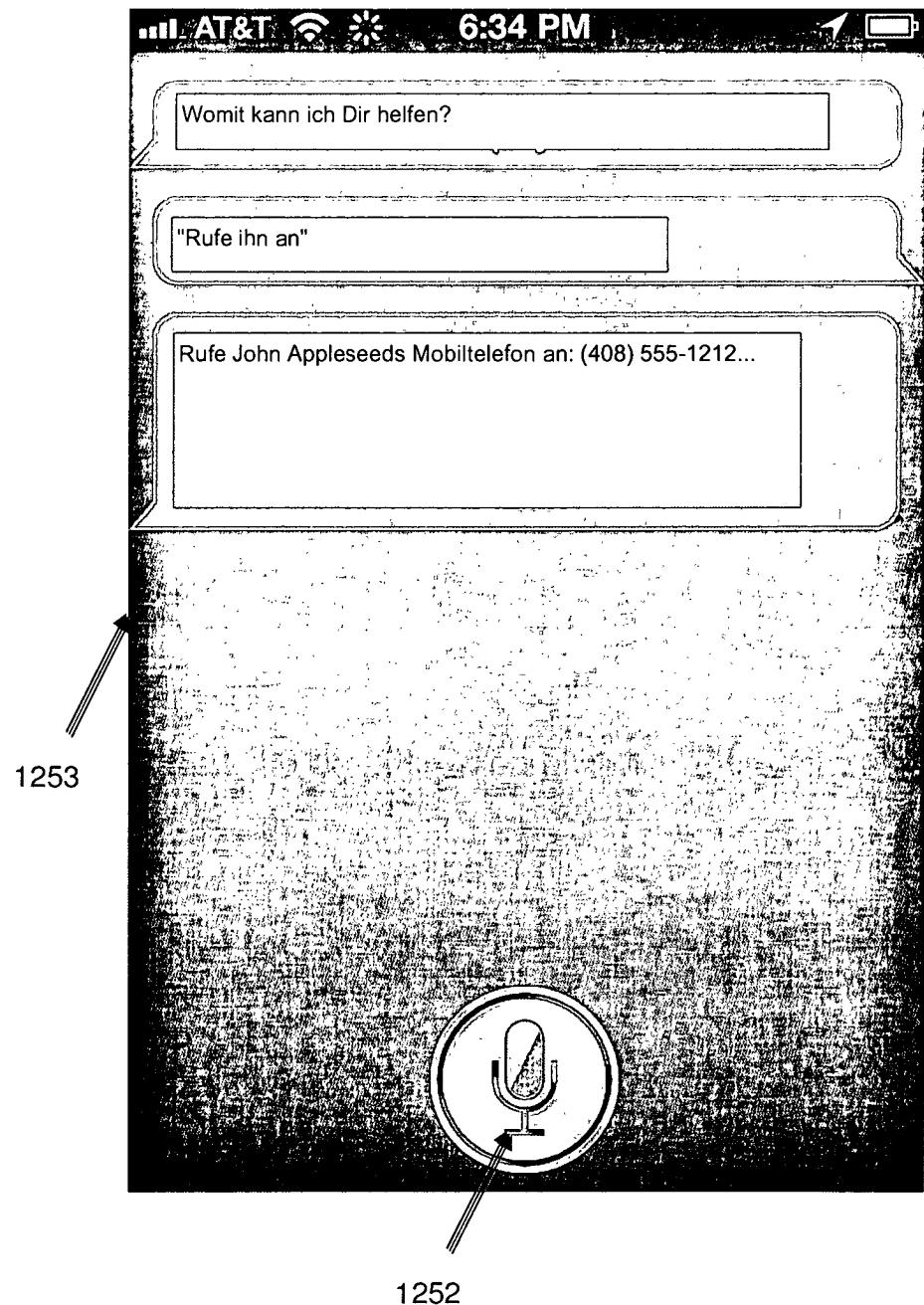


1250



1252

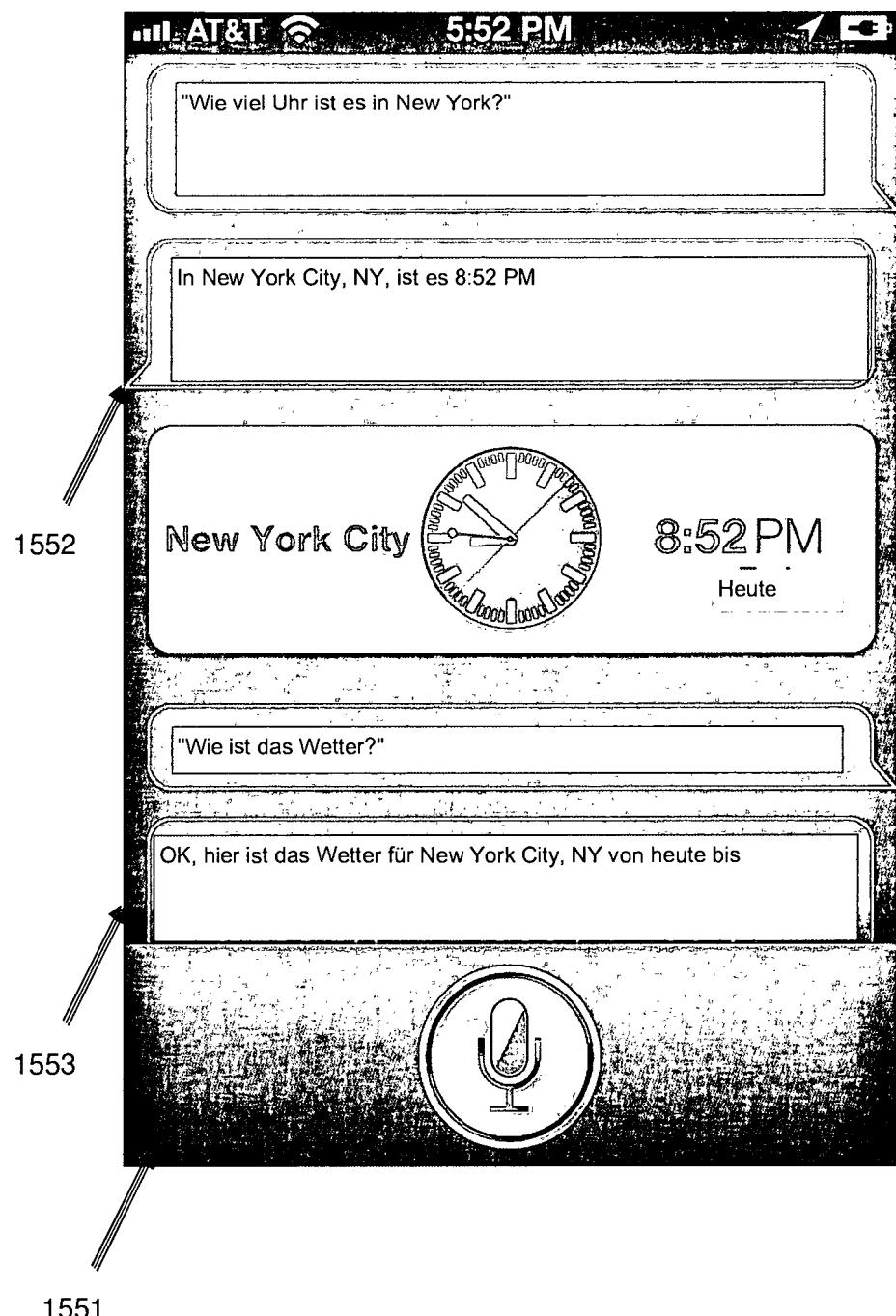
**FIG. 12**



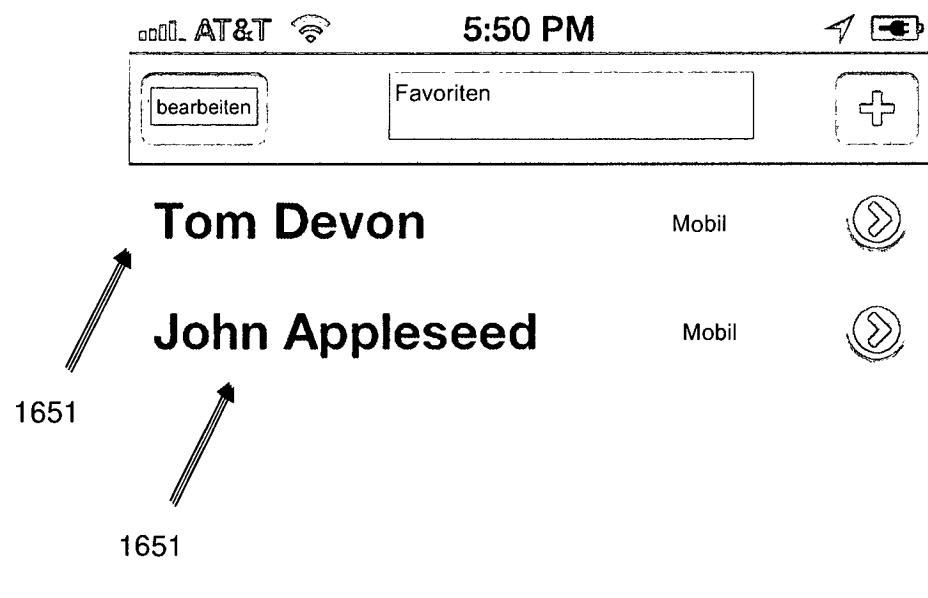
**FIG. 13**



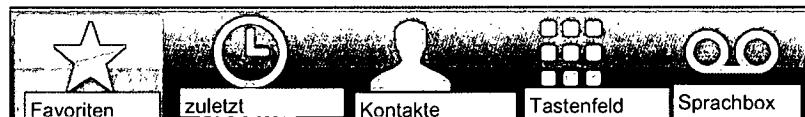
**FIG. 14**



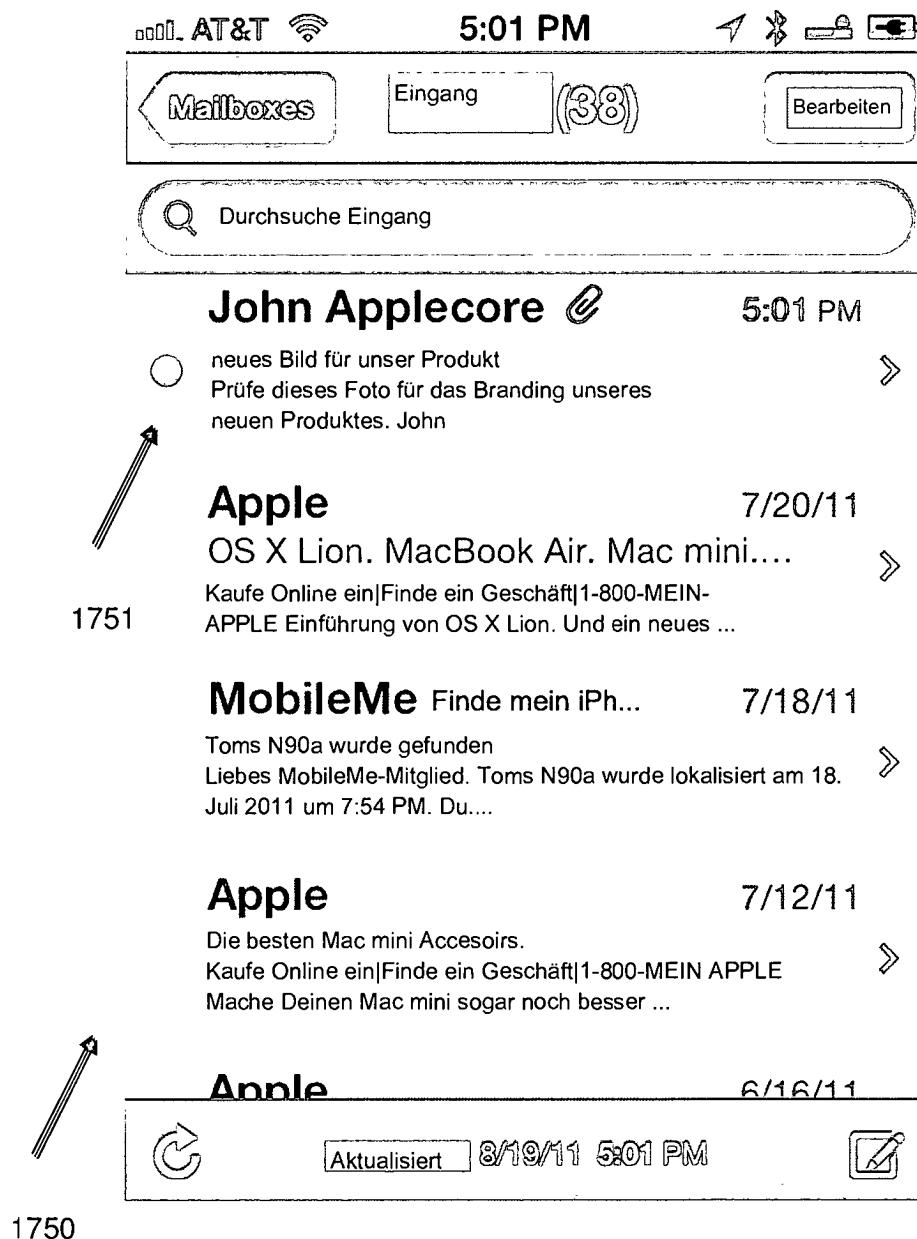
**FIG. 15**

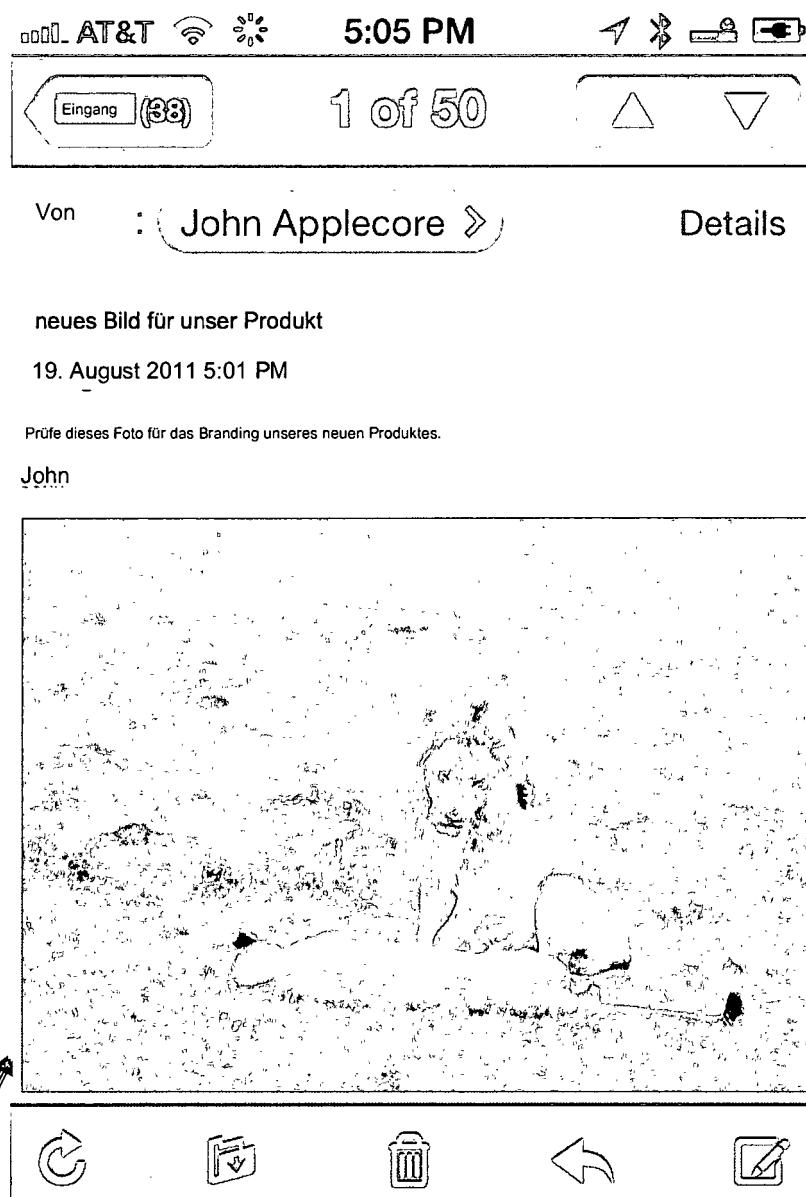


1650



**FIG. 16**

**FIG. 17**

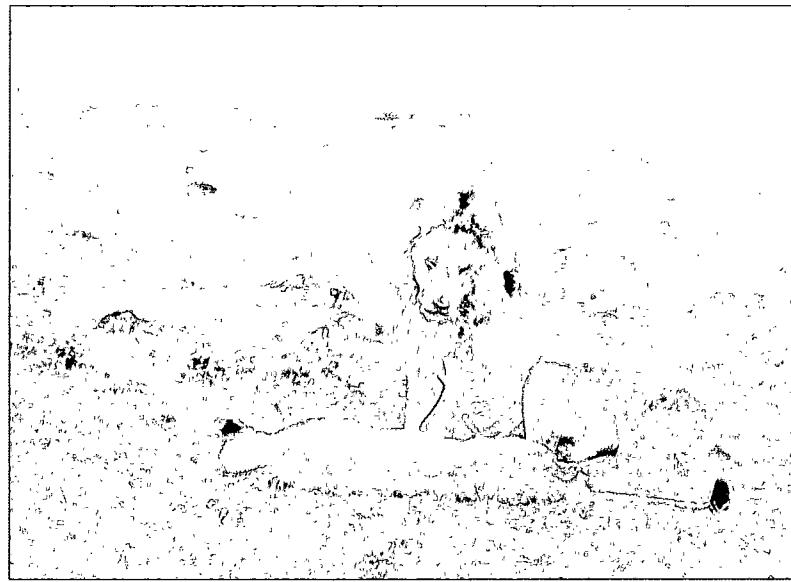


1751

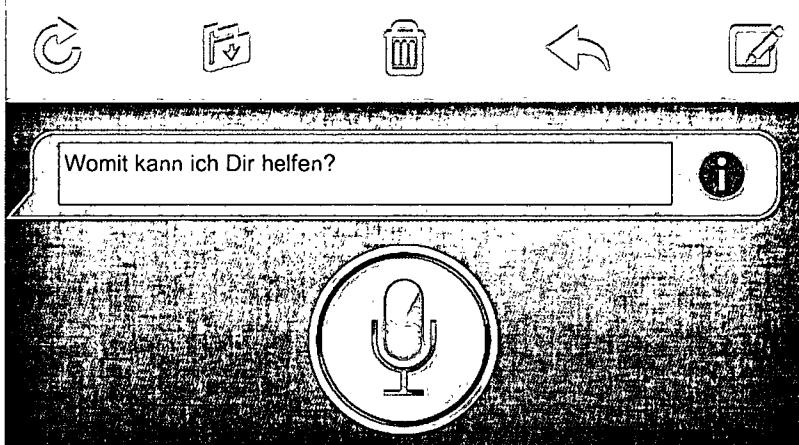
**FIG. 18**

19. August 2011 5:01 PM

Prüfe dieses Foto für das Branding unseres neuen Produktes.  
John

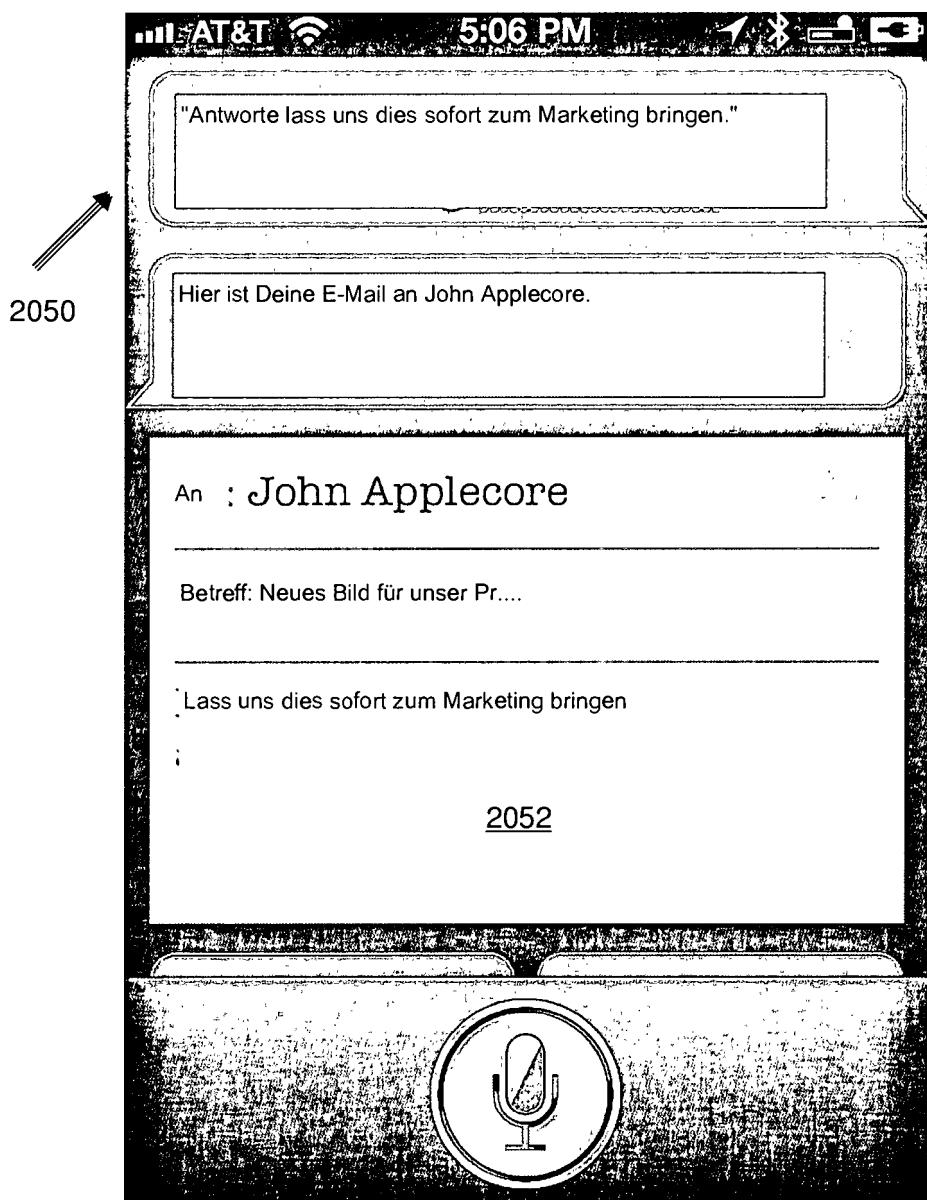


1751

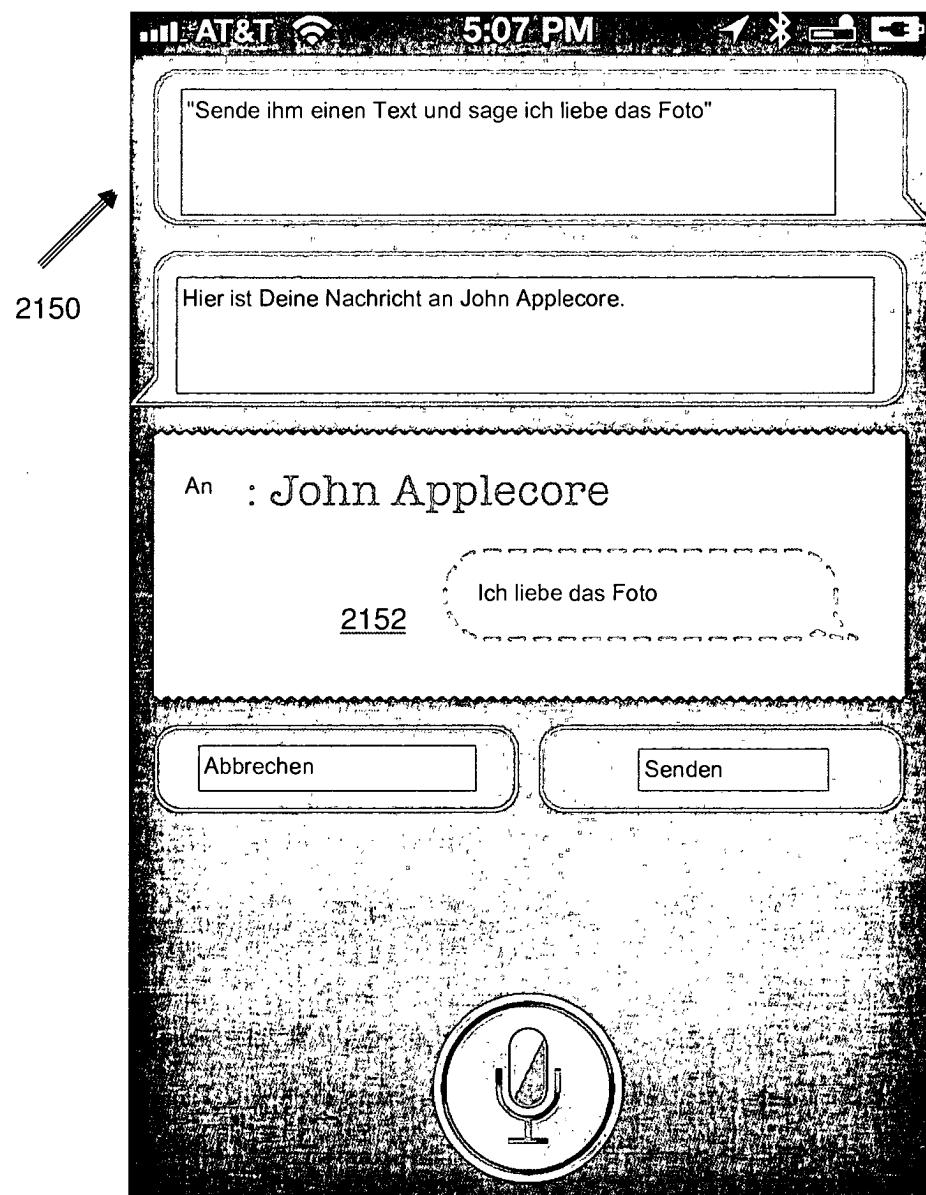


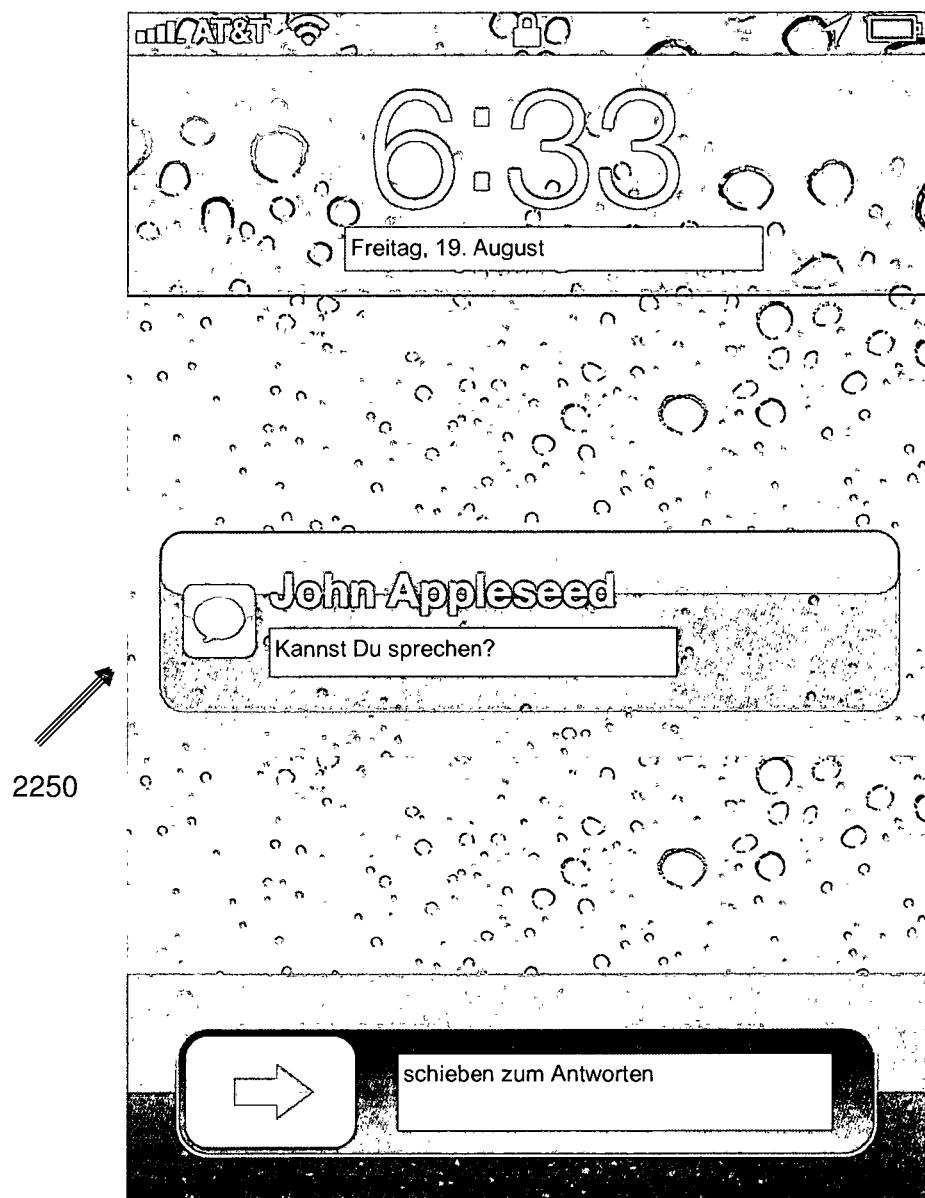
1950

**FIG. 19**

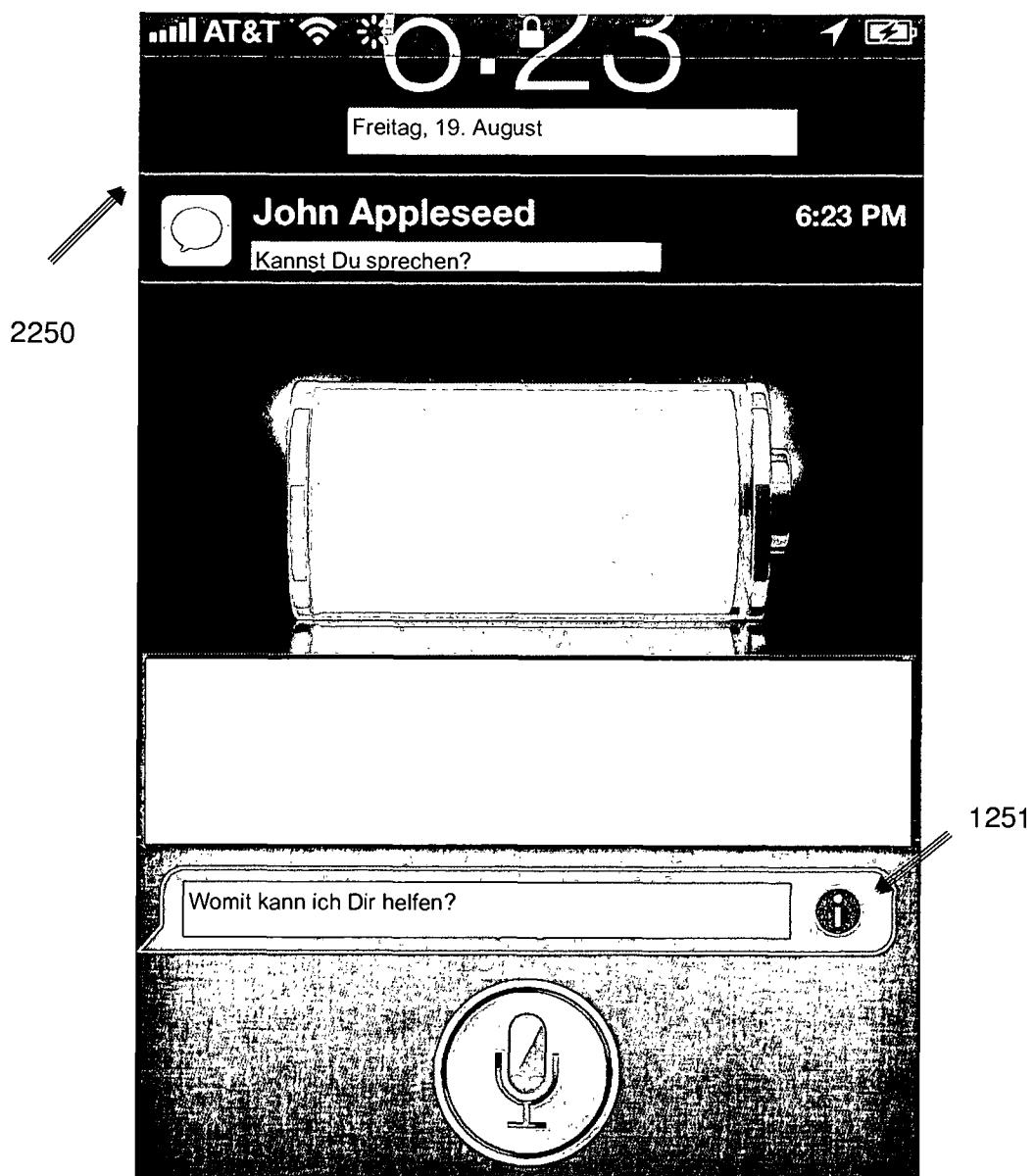


**FIG. 20**

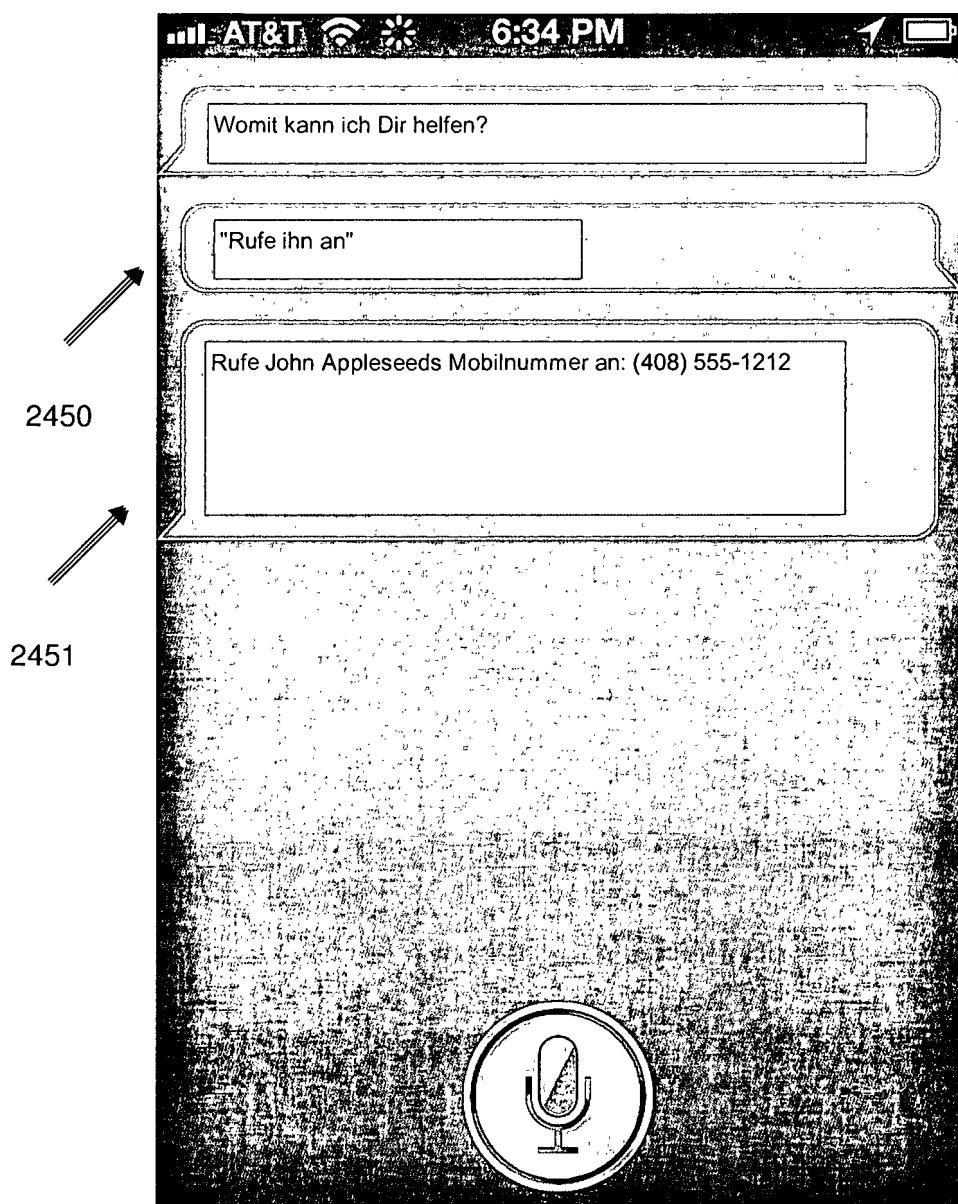




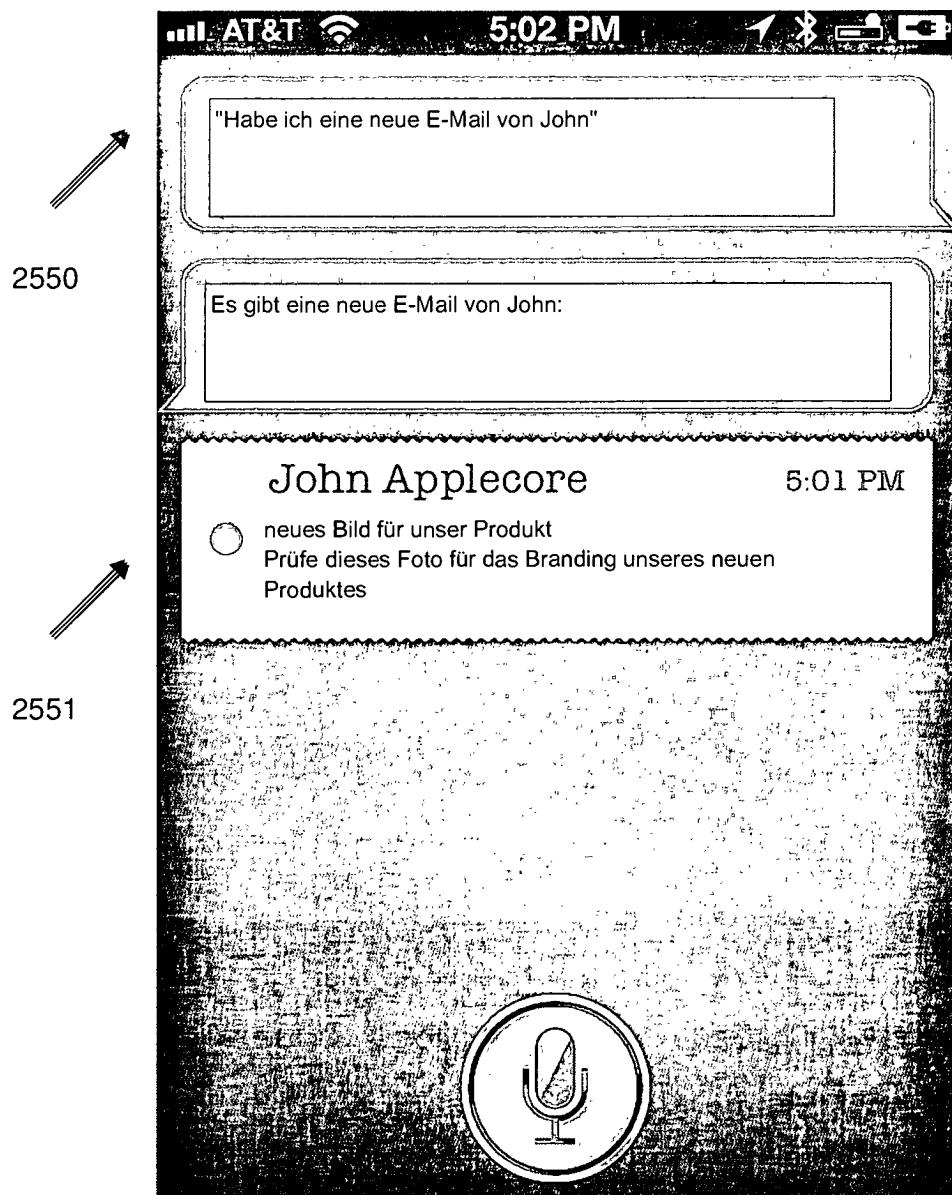
**FIG. 22**



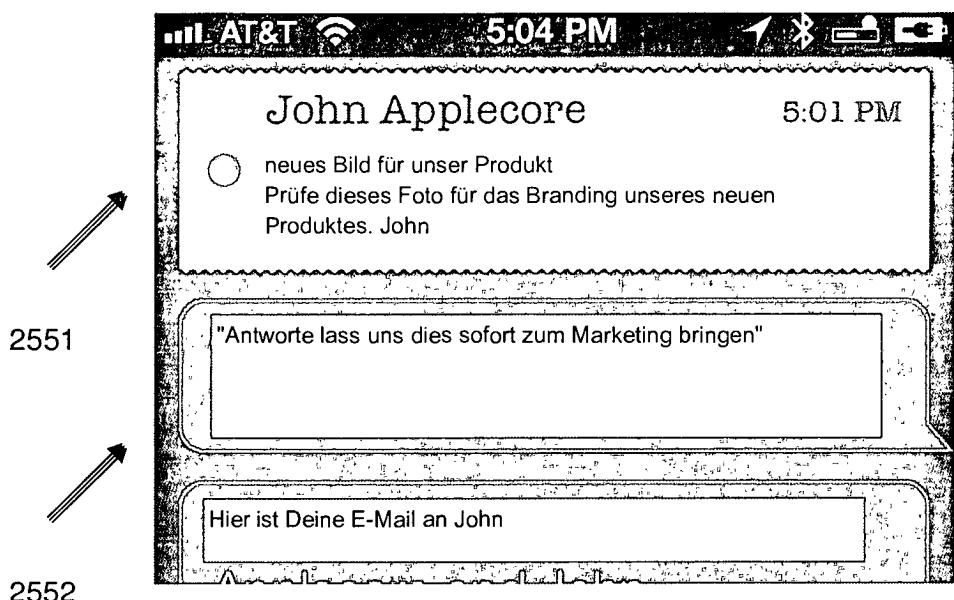
**FIG. 23**



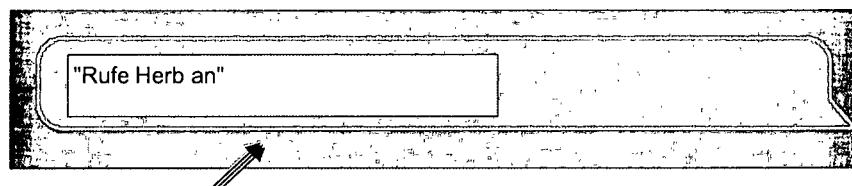
**FIG. 24**



**FIG. 25A**

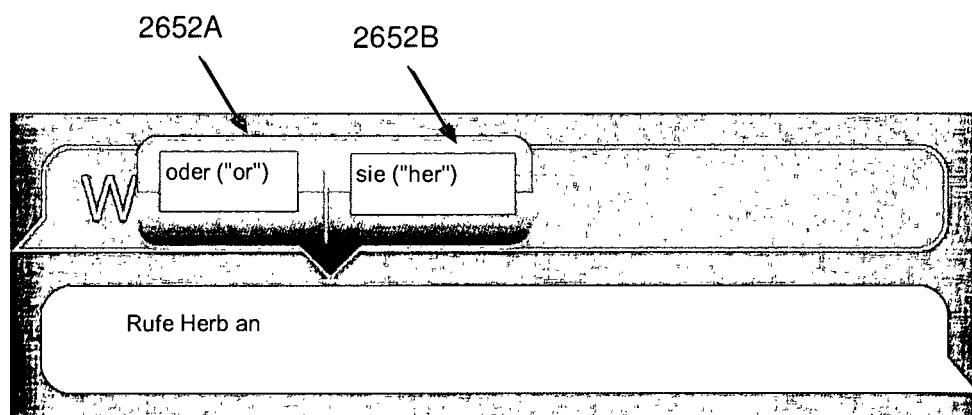


**FIG. 25B**



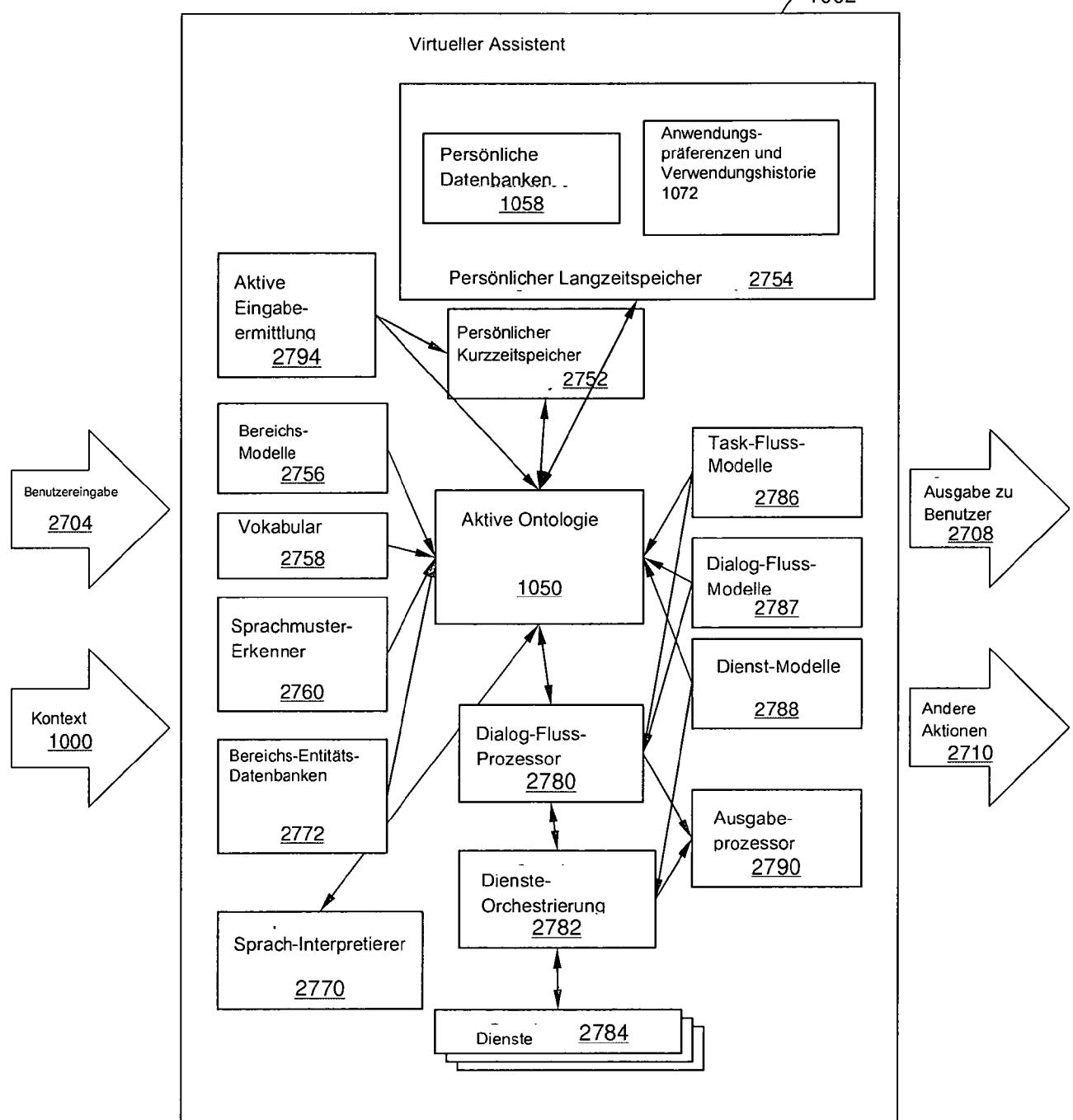
2651

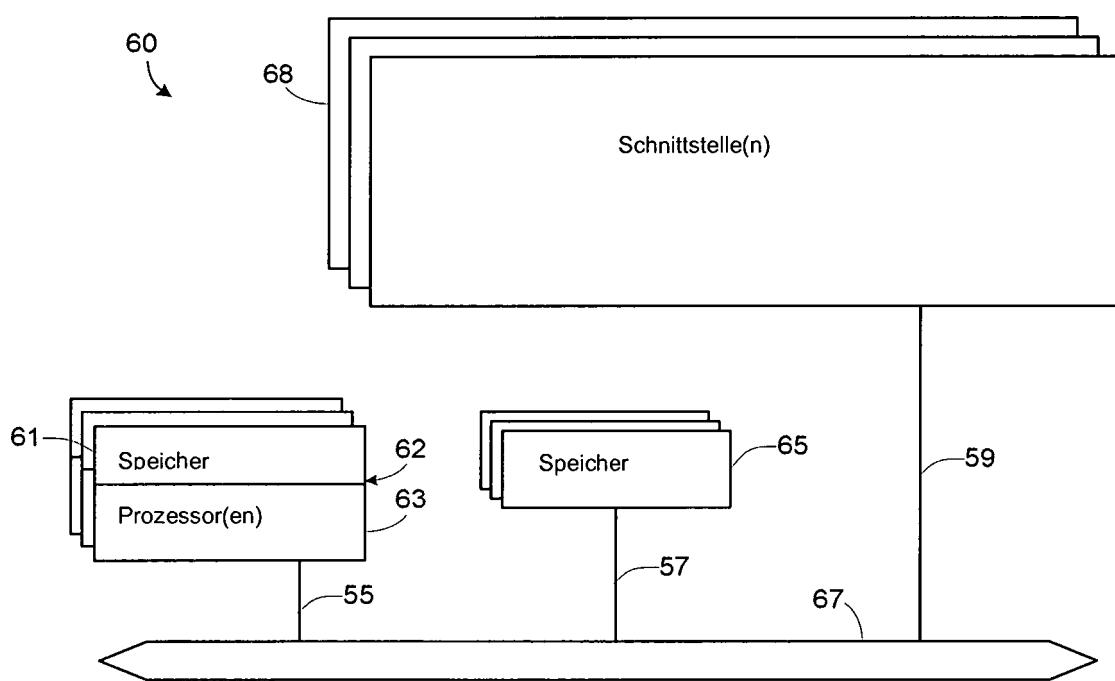
**FIG. 26A**



**FIG. 26B**

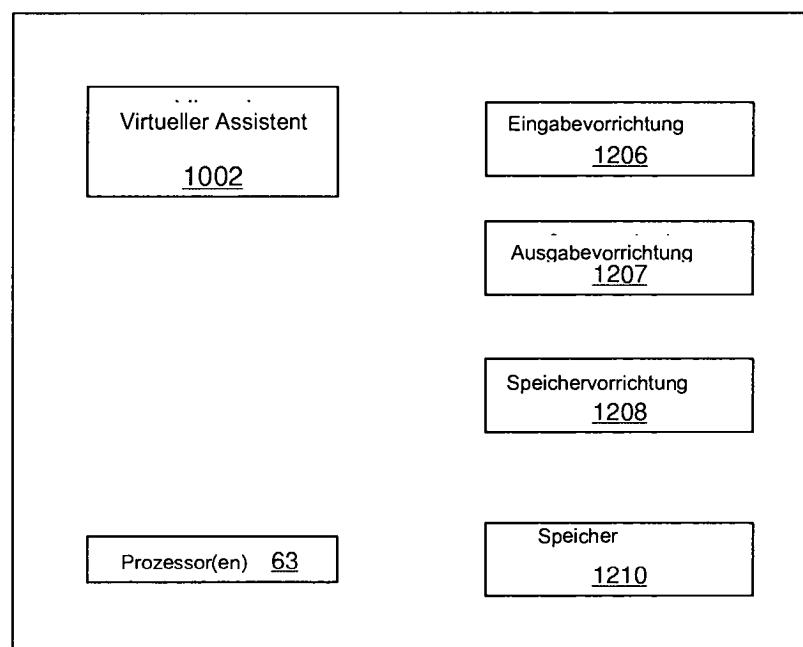
1002

**FIG. 27**

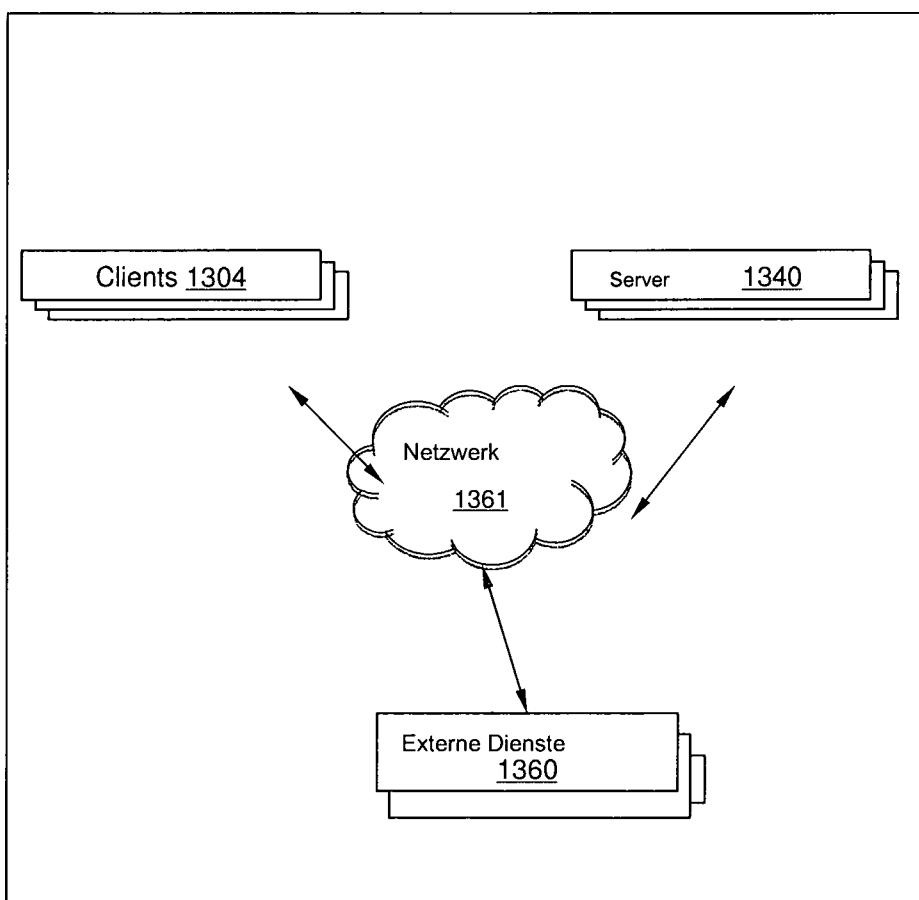


**FIG. 28**

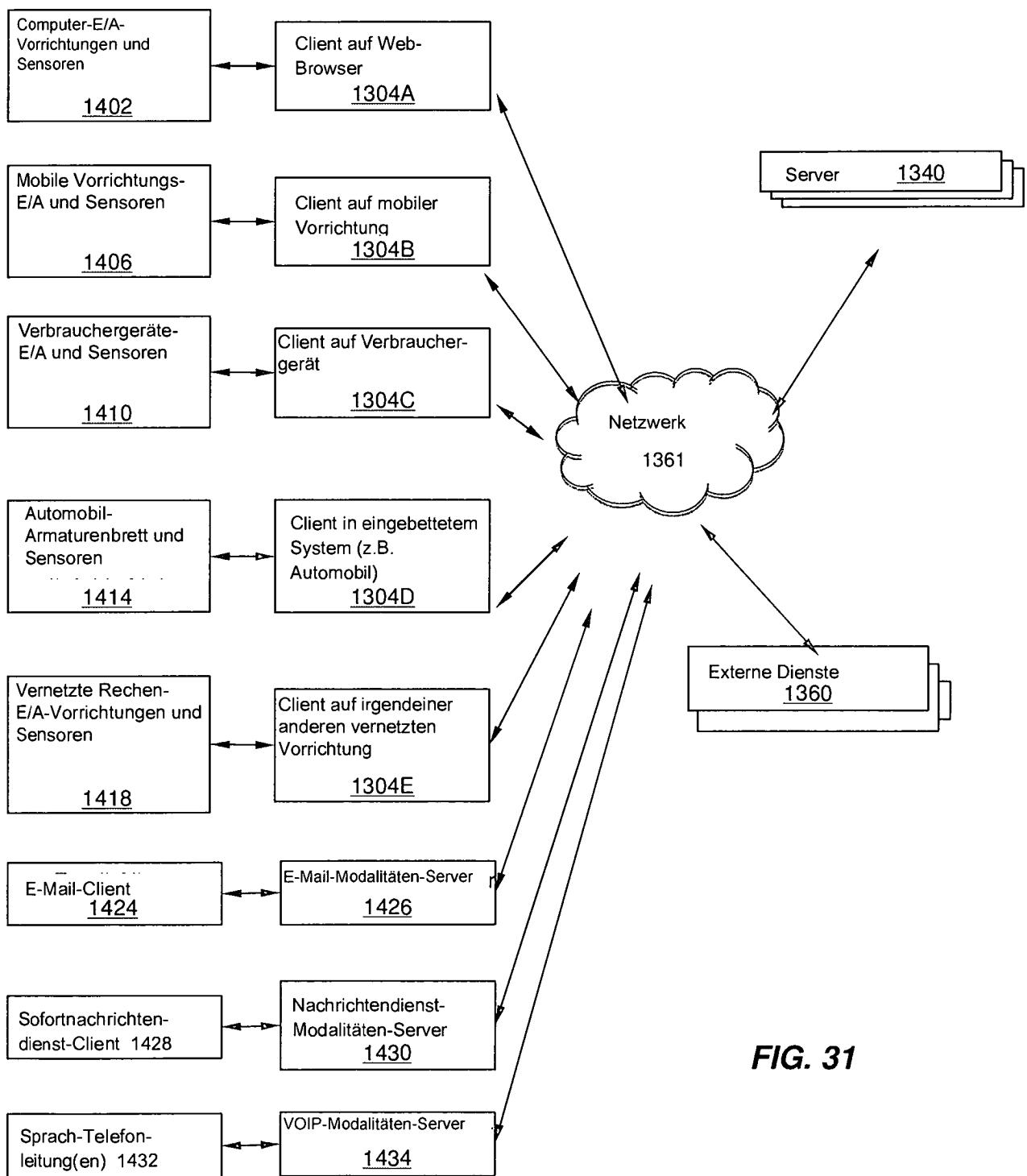
60

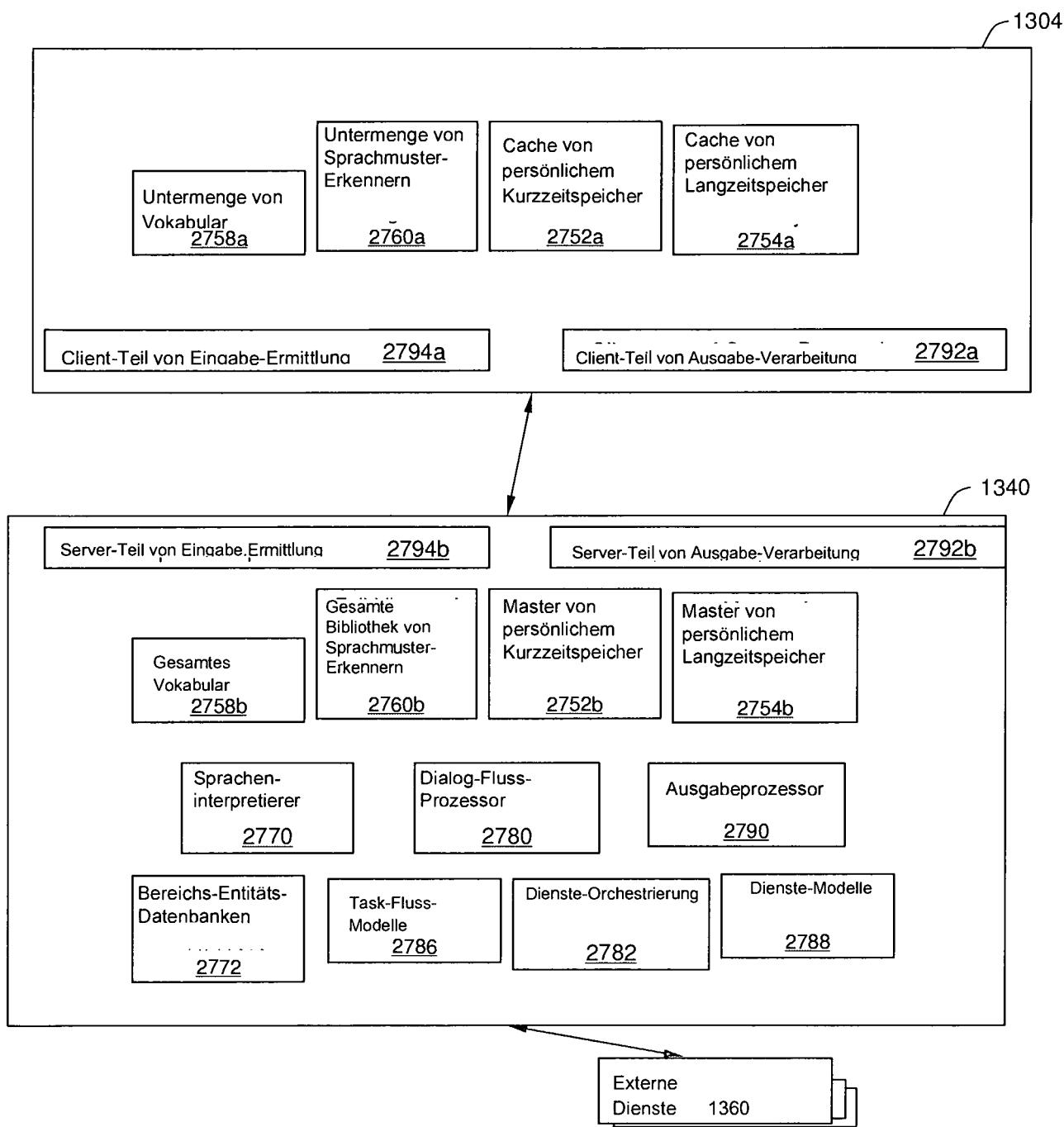


**FIG. 29**



**FIG. 30**

**FIG. 31**



**FIG. 32**