

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2009-500722

(P2009-500722A)

(43) 公表日 平成21年1月8日(2009.1.8)

(51) Int.Cl.

G06F 11/34 (2006.01)
G06F 3/16 (2006.01)

F 1

G06F 11/34
G06F 3/16B
320H

テーマコード(参考)

5 B 0 4 2

審査請求 未請求 予備審査請求 未請求 (全 28 頁)

(21) 出願番号 特願2008-519325 (P2008-519325)
 (86) (22) 出願日 平成18年6月7日 (2006.6.7)
 (85) 翻訳文提出日 平成19年12月21日 (2007.12.21)
 (86) 國際出願番号 PCT/US2006/022137
 (87) 國際公開番号 WO2007/005185
 (87) 國際公開日 平成19年1月11日 (2007.1.11)
 (31) 優先権主張番号 11/170,808
 (32) 優先日 平成17年6月30日 (2005.6.30)
 (33) 優先権主張国 米国(US)

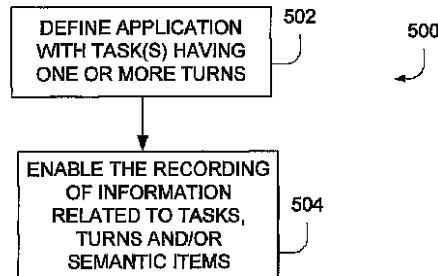
(71) 出願人 500046438
 マイクロソフト コーポレーション
 アメリカ合衆国 ワシントン州 9805
 2-6399 レッドモンド ワン マイ
 クロソフト ウェイ
 (74) 代理人 100077481
 弁理士 谷 義一
 (74) 代理人 100088915
 弁理士 阿部 和夫
 (72) 発明者 ステファン エフ. ポッター
 アメリカ合衆国 98052 ワシントン
 州 レッドモンド ワン マイクロソフト
 ウェイ マイクロソフト コーポラシ
 ョン内

最終頁に続く

(54) 【発明の名称】音声アプリケーションの測定およびロギング

(57) 【要約】

音声対応アプリケーションは、タスクに関して定義される。タスクの完了を示す情報および/またはターンデータに関連する情報は、音声対応アプリケーションが実行される時に、タスクに対して相対的に記録可能である。



【特許請求の範囲】**【請求項 1】**

コンピュータシステムで実行される音声対応アプリケーションでユーザ対話データを口ギングするコンピュータ実施される方法(520)であって、

前記コンピュータシステム上のタスクに関して定義される音声対応アプリケーションを実行すること(522)であって、タスクは、1つまたは複数のターンを含み、ターンは、前記音声対応アプリケーションによってユーザに提供されるプロンプトと、前記音声対応アプリケーションによって前記ユーザに提供されるプロンプトおよびそれに続く前記ユーザからの応答を含むプロンプト／応答交換とのうちの少なくとも1つを含む、実行すること(522)と、

(a) 前記アプリケーションで実行されるタスクの完了と、(b)めいめいのタスクに対する相対的な対応するターンの目的と、(c)前記ユーザからの応答の認識に関して変化する前記アプリケーション内で使用される値の表示とのうちの少なくとも2つを示す情報を記録すること(524)と

を含むことを特徴とするコンピュータ実施される方法(520)。

【請求項 2】

前記音声対応アプリケーションを実行すること(522)は、前記タスク階層構造で定義される、前記音声対応アプリケーションを実行することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 3】

各ターンの前記目的を示す情報を記録すること(524)は、ターンの前記目的が、前記音声対応アプリケーションが質問をすること、回答を確認すること、前記ユーザにヘルプを提供すること、およびプロンプトを繰り返すことのうちの少なくとも1つを含むかどうかを記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 4】

前記めいめいのタスクに対する相対的な各ターンに関する情報を記録すること(524)は、前記プロンプトがどの入力フィールドに関連するかに関する情報を記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 5】

前記めいめいのタスクに対する相対的な各ターンに関する情報を記録すること(524)は、前記応答がどの入力フィールドに関連するかに関する情報を記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 6】

各ターンの前記目的を示す情報を記録すること(524)は、ターンの前記目的が、ユーザがコマンドを提供すること、回答を提供すること、確認を受け入れること、および確認を拒否することのうちの少なくとも1つを含むかどうかを記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 7】

前記めいめいのタスクに対する相対的な各ターンに関する情報を記録すること(524)は、前記音声対応アプリケーションによって提供されるプロンプト、前記プロンプトに応答して前記ユーザによって提供される応答、および前記応答に関する音声リコグナイザによる認識結果に関する情報を記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 8】

タスクの完了を示す情報を記録すること(524)は、成功、失敗、または未知の完了状況値のうちの1つを示す情報を記録することを含むことを特徴とする請求項1に記載のコンピュータ実施される方法(520)。

【請求項 9】

タスクの完了を示す情報を記録すること(524)は、前記タスクに関係するダイアロ

10

20

30

40

50

グの完了の理由を示す情報を記録することを含むことを特徴とする請求項 1 に記載のコンピュータ実施される方法（520）。

【請求項 10】

音声対応アプリケーションを作成する命令を有するコンピュータ可読媒体であって、前記命令は、

コンピュータシステム上の階層構造内のタスクに関して音声対応アプリケーションを定義すること（502）と、

前記階層構造に対して相対的に前記アプリケーション内で実行されるタスクの完了を示す情報の記録を可能にすること（504）と

を含むことを特徴とするコンピュータ可読媒体。

10

【請求項 11】

定義すること（502）は、1つまたは複数のターンを使用するタスクを定義することを含み、ターンは、前記音声対応アプリケーションによってユーザに提供されるプロンプトと、前記音声対応アプリケーションによって前記ユーザに提供されるプロンプトおよびそれに続く前記ユーザからの応答を含むプロンプト／応答交換とのうちの少なくとも1つを含み、情報の前記記録を可能にすることは、対応するタスクに対して相対的に1つまたは複数のターンを示す情報の記録を可能にすることを含むことを特徴とする請求項 10 に記載のコンピュータ可読媒体。

【請求項 12】

めいめいのタスクに対する相対的な各ターンに関する情報の前記記録を可能にすること（504）は、各ターンの目的を示す情報の記録を可能にすることを含むことを特徴とする請求項 10 に記載のコンピュータ可読媒体。

20

【請求項 13】

各ターンの前記目的を示す情報の前記記録を可能にすること（504）は、ターンの前記目的が、前記音声対応アプリケーションが質問をすること、回答を確認すること、前記ユーザにヘルプを提供すること、およびプロンプトを繰り返すことのうちの少なくとも1つを含むかどうかを記録することを含むことを特徴とする請求項 12 に記載のコンピュータ可読媒体。

【請求項 14】

各ターンの前記目的を示す情報の前記記録を可能にすること（504）は、ターンの前記目的が、ユーザがコマンドを提供すること、回答を提供すること、確認を受け入れること、および確認を拒否することのうちの少なくとも1つを含むかどうかの記録を可能にすることを含むことを特徴とする請求項 12 に記載のコンピュータ可読媒体。

30

【請求項 15】

各ターンに関する情報の前記記録を可能にすること（504）は、前記音声対応アプリケーションによって提供されるプロンプト、前記プロンプトに応答して前記ユーザによって提供される応答、および前記応答に関する音声リコグナイザによる認識結果に関する情報の記録を可能にすることを含むことを特徴とする請求項 12 に記載のコンピュータ可読媒体。

【請求項 16】

前記めいめいのタスクに対する相対的な各ターンに関する情報の前記記録を可能にすること（504）は、前記プロンプトがどの入力フィールドに関連するかに関する情報の記録を可能にすることを含むことを特徴とする請求項 12 に記載のコンピュータ可読媒体。

40

【請求項 17】

前記めいめいのタスクに対する相対的な各ターンに関する情報の前記記録を可能にすること（504）は、前記応答がどの入力フィールドに関連するかに関する情報の記録を可能にすることを含むことを特徴とする請求項 12 に記載のコンピュータ可読媒体。

【請求項 18】

音声対応アプリケーションを作成する命令を有するコンピュータ可読媒体であって、前記命令は、

50

コンピュータシステム上のタスクに関して音声対応アプリケーションを定義すること(502)であって、タスクは、1つまたは複数のターンを含み、ターンは、前記音声対応アプリケーションによってユーザに提供されるプロンプトと、前記音声対応アプリケーションによって前記ユーザに提供されるプロンプトおよびそれに続く前記ユーザからの応答を含むプロンプト/応答交換とのうちの少なくとも1つを含む、定義すること(502)と、

前記音声対応アプリケーションの実行中に、前記1つまたは複数のターンのそれぞれに関するユーザおよびシステムの目的を示し、(a)前記アプリケーション内で実行されるタスクの完了および(b)前記ユーザからの応答の認識に関して変化する前記アプリケーション内で使用される値の表示のうちの少なくとも1つに関連する情報の記録を可能にすること(504)と

を含むことを特徴とするコンピュータ可読媒体。

【請求項19】

タスクの完了を示す情報の前記記録を可能にすること(504)は、成功、失敗、または未知の完了状況値のうちの1つを示す情報の記録を可能にすることを含むことを特徴とする請求項18に記載のコンピュータ可読媒体。

【請求項20】

情報の前記記録を可能にすること(504)は、プロンプトがどの入力フィールドに関連するかに関する情報の記録および応答がどの入力フィールドに関連するかに関する情報の記録を可能にすることを含むことを特徴とする請求項19に記載のコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

以下の議論は、単に、全般的背景情報として提供され、請求される主題の範囲を判定する際の助けとして使用されることは意図されていない。

【背景技術】

【0002】

携帯情報端末(PDA)、デバイス、および携帯電話機などの小型コンピューティング、デバイスは、人々によって、彼らの毎日の活動でますます高まる頻度で使用されている。現在これらのデバイスを動作させるのに使用されるマイクロプロセッサに使用可能な処理能力の向上に伴って、これらのデバイスの機能性は、高まりつつあり、いくつかの場合に、合併されつつある。たとえば、多数の携帯電話機は、現在、インターネットにアクセスし、ブラウズするのに使用することができ、住所、電話番号、および類似物などの個人情報を記憶するのに使用することができる。

【0003】

したがって、これらのコンピューティングデバイスが高まる頻度で使用されつつあることを考慮すると、ユーザが情報をこれらのコンピューティングデバイスに入力する簡単なインターフェースを提供する必要がある。残念ながら、これらのデバイスを簡単に持ち運べるようにするためにこれらのデバイスをできる限り小さくすることの望みに起因して、分離されたボタンとしてアルファベットのすべての文字を有する従来のキーボードは、通常、これらのコンピューティングデバイスのハウジングで使用可能な限られた表面積に起因して、可能ではない。小型コンピューティングデバイスの例を超えてさえ、すべてのタイプのコンピューティングデバイス用のより便利なインターフェースを提供することへの関心がある。

【0004】

この問題に対処するために、コンピューティングデバイス上でローカルに、ローカルネットワークを介して、または、インターネットなどの広域ネットワークを介してのいずれであれ、情報にアクセスするのにボイスまたは音声を使用することへの高まる関心および採用があった。音声認識を用いると、ダイアログ対話は、一般に、ユーザとコンピューテ

イングデバイスとの間で行われる。ユーザは、通常は可聴および／または可視に情報を受け取り、プロンプトに可聴に応答するか、コマンドを発行する。しかし、しばしば、アプリケーションの開発中または展開後に、そのアプリケーションの性能を確かめることが望ましい。具体的に言うと、アプリケーションに関するユーザの使用法および／または成功率を確かめることが望ましい。そのような情報があれば、開発者は、アプリケーションのユーザの必要をよりよく満足するために、アプリケーションを「チューニングする」（すなわち、調整を行う）ことができる可能性がある。たとえば、アプリケーションとユーザとの間のダイアログのうちで、問題に出会う可能性が最も高い部分を識別することが、有用である場合がある。この形で、ダイアログのこれらの部分を、混乱を軽減するために調整することができる。

10

【0005】

【特許文献1】米国特許出願第2004/0113908号明細書

【特許文献2】米国特許出願第2004/0230637A1号明細書

【発明の開示】

【発明が解決しようとする課題】

【0006】

アプリケーションとユーザとの間の対話データの記録またはロギングが、アプリケーションの性能を測定するために行われる。しかし、一般に、アプリケーション対話データのロギングは、たとえば（1）データを生成するのが面倒である、すなわち、アプリケーション開発者が、分析およびチューニング用の正しいデータを収集するために、コード内にさまざまな位置でアプリケーションを計測する（すなわち、システムデータのロギングに使用されるメッセージのセットを定義し、実装する）世話をしなければならず、（2）計測プロセスが、通常はアプリケーション固有の形で行われ、異なるアプリケーションにまたがってポータブルではなく、（3）ユーザの意図に関するより豊富な情報を用いてデータに注釈を付ける手動トランск립ションプロセス（および／または他の明示的な人間による介入）が適用されない限り、対話ログデータが、限られた価値を有する、という短所のうちのいずれか1つまたはその組合せから損害を受ける場合がある。

20

【課題を解決するための手段】

【0007】

この「発明の開示」は、下の「発明を実施するための最良の形態」でさらに説明されるいくつかの概念を単純化された形で紹介するために提供される。この「発明の開示」では、請求される主題の主要な特徴または本質的特徴を識別することは意図されておらず、請求される主題の範囲を判定する際の助けとして使用されることも意図されていない。

30

【0008】

音声対応アプリケーションは、タスクに関して定義される。タスクの完了を示す情報および／またはターンデータ（turn data）に関する情報が、音声対応アプリケーションが実行される時にタスクに対して相対的に記録可能である。

【0009】

タスクの完了を示す情報を、Dialogデータと称する。このデータは、タスクの完了の成功または失敗を定量化する。さらに、Dialogデータには、タスクが不成功であるまたは失敗する場合に理由を含めることができ、あるいは、適用可能な場合に、成功について複数の理由が可能である場合に成功の理由を含めることができる。追加データには、ユーザが応答を提供しなかった場合または音声リコグナイザが発話を認識できなかった場合を示す進行状況データを含めることができる。変化した入力フィールド値またはその状況のリストをも記録することができる。

40

【0010】

Turnデータは、アプリケーションとの直接対話を含み、アプリケーションによって提供されるプロンプト（応答が期待されない時）またはユーザ応答もしくはその不在に相關されたアプリケーションプロンプト、言い換えると、プロンプト／応答交換に基づいて編成される。したがって、記録できるデータの3つの領域は、プロンプトの目的を含む、

50

アプリケーションによって提供されるプロンプトと、応答の目的を含む、ユーザによって提供される応答と、システムによって判定された認識結果とに関する情報を含む。

【発明を実施するための最良の形態】

【0011】

音声アプリケーションの計測およびログ記録ならびにこれを実施する方法を説明する前に、音声アプリケーションで使用できるコンピューティングデバイスを全般的に説明することが有用である可能性がある。ここで図1を参照すると、データ管理デバイス（PIM、PDA、または類似物）の例示的な形が、30に示されている。しかし、本明細書で説明する概念を、下で述べる他のコンピューティングデバイス、具体的には入力ボタンまたは類似物のための限られた表面積を有するコンピューティングデバイスを使用して実践することもできることが、企図されている。たとえば、電話機および／またはデータ管理デバイスも、本明細書で説明される概念から利益を得る。そのようなデバイスは、既存のポータブルパーソナル情報管理デバイスおよび他のポータブル電子デバイスと比較して高められたユーティリティを有し、そのようなデバイスの機能およびコンパクトなサイズは、ユーザがそのデバイスを常時持ち運ぶことを促進する可能性がより高い。したがって、本明細書で説明される応用例の範囲が、本明細書で示される例示的なデータ管理デバイスもしくはPIMデバイス、電話機、またはコンピュータの開示に限定されることは、意図されていない。

10

【0012】

データ管理モバイルデバイス30の例示的な形が、図1に示されている。モバイルデバイス30は、ハウジング32を含み、ディスプレイ34を含むユーザインターフェースを有し、ディスプレイ34は、スタイラス33と共に接触感知型ディスプレイスクリーンを使用する。スタイラス33は、指定された座標でディスプレイ34を押すかこれに接触して、フィールドを選択し、カーソルの開始位置を選択的に移動し、またはジェスチャもしくは手書きを介するなど、他の形でコマンド情報を提供するのに使用される。その代わりにまたはそれに加えて、1つまたは複数のボタン35を、ナビゲーションのためにデバイス30に含めることができる。さらに、回転可能なホイール、ローラー、または類似物などの他の入力機構を設けることもできる。しかし、本発明がこれらの形の入力機構によって限定されることが意図されていないことに留意されたい。たとえば、別の形の入力に、コンピュータビジョンを介するなどのビジュアル入力を含めることができる。

20

【0013】

ここで図2を参照すると、ブロック図に、モバイルデバイス30を含む機能コンポーネントが示されている。中央処理装置（CPU）50が、ソフトウェア制御機能を実装する。CPU50は、ディスプレイ34に結合され、その結果、制御ソフトウェアに従って生成されたテキストおよびグラフィックアイコンが、ディスプレイ34に表示されるようになる。スピーカ43を、通常はデジタル・アナログコンバータ59と共にCPU50に結合して、可聴出力を提供することができる。モバイルデバイス30にダウンロードされるかユーザによって入力されたデータは、CPU50に両方向で結合された不揮発性読み取り／書き込みランダムアクセスメモリストア54に格納される。ランダムアクセスメモリ（RAM）54は、CPU50によって実行される命令の揮発性ストレージと、レジスタ値などの一時的データのストレージとをもたらす。構成オプションおよび他の変数のデフォルト値は、読み取り専用メモリ（ROM）58に格納される。ROM58は、モバイル30の基本機能性および他のオペレーティングシステムカーネル機能（たとえば、RAM54へのソフトウェアコンポーネントのロード）を制御する、デバイスのオペレーティングシステムソフトウェアを格納するのにも使用することができる。

30

【0014】

RAM54は、アプリケーションプログラムを格納するのに使用されるPC上のハードドライブの機能に似た形で、コードのストレージとしても働く。不揮発性メモリが、コードの格納に使用されているが、その代わりに、コードの実行には使用されない揮発性メモリにコードを格納することができることに留意されたい。

40

50

【0015】

無線信号を、CPU 50 に結合された無線トランシーバ 52 を介してモバイルデバイスによって送信 / 受信することができる。オプションの通信インターフェース 60 を、望まれる場合に、コンピュータ（たとえば、デスクトップコンピュータ）または有線ネットワークから直接にデータをダウンロードするために設けることもできる。したがって、インターフェース 60 は、さまざまな形の通信デバイス、たとえば、赤外線リンク、モデム、ネットワークカード、または類似物を含むことができる。

【0016】

モバイルデバイス 30 は、マイクロホン 29、アナログ - デジタル (A / D) コンバータ 37、およびストア 54 に格納されたオプションの認識プログラム（音声、DTMF、手書き、ジェスチャー、またはコンピュータビジョン）を含む。たとえば、デバイス 30 のユーザからの可聴の情報、命令、またはコマンドに応答して、マイクロホン 29 は、音声信号を供給し、この信号が A / D コンバータ 37 によってデジタル化される。音声認識プログラムは、デジタル化された音声信号に対して正規化機能および / または特徴抽出機能を実行して、中間音声認識結果を得ることができる。無線トランシーバ 52 または通信インターフェース 60 を使用して、音声データを、下で述べる、図 4 のアーキテクチャに示されたリモート認識サーバ 204 に送信することができる。その後、認識結果を、モバイルデバイス 30 でのレンダリング（たとえば、ビジュアルおよび / または可聴）およびウェブサーバ 202（図 4）への最終的な送信のためにモバイルデバイス 30 に返すことができ、ここで、ウェブサーバ 202 およびモバイルデバイス 30 は、クライアント / サーバ関係で動作する。類似する処理を、他の形の入力について使用することができる。たとえば、手書き入力を、デバイス 30 での前処理を伴ってまたはこれを伴わずにデジタル化することができる。音声データと同様に、この形の入力を、認識のために認識サーバ 204 に送信することができ、ここで、認識結果は、その後、デバイス 30 および / またはウェブサーバ 202 のうちの少なくとも 1 つに返される。同様に、DTMF データ、ジェスチャーデータ、およびビジュアルデータを、似た形で処理することができる。入力の形に応じて、デバイス 30（および上で述べた他の形のクライアント）は、ビジュアル入力用のカメラなど、必要なハードウェアを含むはずである。

10

20

30

【0017】

上で説明したポータブルコンピューティングデバイスまたはモバイルコンピューティングデバイスに加えて、本明細書で説明する概念を、一般的なデスクトップコンピュータなどの多数の他のコンピューティングデバイスと共に使用できることを理解されたい。たとえば、限られた身体能力を有するユーザは、フル英数字キーボードなどの他の従来の入力装置の操作が難し過ぎる場合に、コンピュータまたは他のコンピューティングデバイスにテキストを入力することができる。

40

【0018】

本発明は、多数の他の汎用のまたは特殊目的のコンピューティングシステム、コンピューティング環境、またはコンピューティング構成と共に動作する。本発明と共に使用するのに適切である可能性がある周知のコンピューティングシステム、コンピューティング環境、および / またはコンピューティング構成の例は、無線電話機またはセルラ電話機、通常の電話機（スクリーンを伴わない）、パーソナルコンピュータ、サーバコンピュータ、ハンドヘルドデバイスまたはラップトップデバイス、マルチプロセッサシステム、マイクロプロセッサベースのシステム、セットトップボックス、プログラマブル消費者電子製品、ネットワーク PC、ミニコンピュータ、メインフレームコンピュータ、上記のシステムまたはデバイスのいずれかを含む分散コンピューティング環境、および類似物を含むが、これらに限定はされない。

50

【0019】

次は、図 3 に示された汎用コンピュータ 120 の短い説明である。しかし、コンピュータ 120 は、やはり、適切なコンピューティング環境の 1 つの例にすぎず、本発明の使用または機能性の範囲に関する限定を提案することは意図されていない。コンピュータ 12

50

0を、その中に示されたコンポーネントのいずれかまたはその組合せに関する依存性または要件を有するものと解釈してもならない。

【0020】

下の説明を、コンピュータによって実行される、プログラムモジュールなどのコンピュータ実行可能命令の全体的な文脈で提供することができる。一般に、プログラムモジュールは、特定のタスクを実行するか特定の抽象データ型を実施するルーチン、プログラム、オブジェクト、コンポーネント、データ構造体などを含む。本明細書で説明される例示的実施形態は、通信ネットワークを介してリンクされるリモート処理デバイスによってタスクが実行される分散コンピューティング環境で実践することもできる。分散コンピューティング環境では、プログラムモジュールを、メモリストレージデバイスを含む、ローカルコンピュータ記憶媒体とリモートコンピュータ記憶媒体との両方に配置することができる。プログラムおよびモジュールによって実行されるタスクを、下で、図面の助けを得て説明する。当業者は、この説明および図面を、プロセッサ実行可能命令として実装することができ、このプロセッサ実行可能命令は、任意の形のコンピュータ可読媒体に書き込まれ得る。

10

【0021】

図3を参照すると、コンピュータ120のコンポーネントは、処理ユニット140と、システムメモリ150と、システムメモリを含むさまざまなシステムコンポーネントを処理ユニット140に結合するシステムバス141とを含むことができるが、これらに限定はされない。システムバス141は、メモリバスまたはメモリコントローラ、周辺バス、およびさまざまなバスアーキテクチャのいずれかを使用するローカルバスを含む複数のタイプのバス構造のいずれかとすることができます。限定ではなく例として、そのようなアーキテクチャに、Industry Standard Architecture (ISA) バス、Universal Serial Bus (USB)、マイクロチャネルアーキテクチャ (MCA) バス、Enhanced ISA (EISA) バス、Video Electronics Standards Association (VESA) ローカルバス、およびメザニンバスとも称するPeripheral Component Interconnect (PCI) バスが含まれる。コンピュータ120には、通常は、さまざまなコンピュータ可読媒体が含まれる。コンピュータ可読媒体は、コンピュータ120によってアクセスすることができるすべての使用可能な媒体とすることができます、コンピュータ可読媒体には、揮発性媒体と不揮発性媒体との両方、取外し可能媒体と取外し不能媒体との両方が含まれる。限定ではなく例として、コンピュータ可読媒体に、コンピュータ記憶媒体および通信媒体を含めることができます。コンピュータ記憶媒体には、コンピュータ可読命令、データ構造体、プログラムモジュール、または他のデータなどの情報を記憶する任意の方法またはテクノロジで実施される、揮発性と不揮発性と両方の、取外し可能および取外し不能の媒体が含まれる。コンピュータ記憶媒体には、RAM、ROM、EEPROM、フラッシュメモリ、または他のメモリテクノロジ、CD-ROM、ディジタル多用途ディスク (DVD)、または他の光学ディスクストレージ、磁気カセット、磁気テープ、磁気ディスクストレージ、または他の磁気ストレージデバイス、あるいは所望の情報を格納するのに使用することができ、コンピュータ120によってアクセスすることができるすべての他の媒体が含まれるが、これに限定はされない。

20

【0022】

通信媒体は、通常、搬送波または他のトランスポート機構などの変調されたデータ信号内でコンピュータ可読命令、データ構造体、プログラムモジュール、または他のデータを実施し、通信媒体は、すべての情報配信媒体を含む。用語「変調されたデータ信号」は、信号内で情報を符号化する形でその特性の1つまたは複数を設定または変更された信号を意味する。制限ではなく例として、通信媒体は、有線ネットワークまたは直接配線接続などの有線媒体と、音響、RF、赤外線、および他の無線媒体などの無線媒体を含む。上記のいずれかの組合せも、コンピュータ可読媒体の範囲に含まれなければならない。

30

【0023】

40

50

システムメモリ 150 には、読み取り専用メモリ (ROM) 151 およびランダムアクセスメモリ (RAM) 152 などの揮発性および / または不揮発性のメモリの形のコンピュータ記憶媒体が含まれる。起動中などにコンピュータ 120 内の要素の間での情報の転送を助ける基本ルーチンを含む基本入出力システム 153 (BIOS) が、通常は ROM 151 に格納される。RAM 152 には、通常は、即座にアクセス可能であるおよび / または処理ユニット 140 によって現在操作されているデータおよび / またはプログラムモジュールが含まれる。制限ではなく例として、図 3 に、オペレーティングシステム 154、アプリケーションプログラム 155、他のプログラムモジュール 156、およびプログラムデータ 157 を示す。

【0024】

10

コンピュータ 120 には、他の取外し可能 / 取外し不能の揮発性 / 不揮発性コンピュータ記憶媒体を含めることもできる。例としてのみ、図 3 に、取外し不能不揮発性磁気媒体から読み取るかこれに書き込むハードディスクドライブ 161、取外し可能不揮発性磁気ディスク 172 から読み取るかこれに書き込む磁気ディスクドライブ 171、および CD-ROM または他の光学媒体などの取外し可能不揮発性光ディスク 176 から読み取るかこれに書き込む光学ディスクドライブ 175 を示す。この例示的オペレーティング環境で使用することができる他の取外し可能 / 取外し不能の揮発性 / 不揮発性コンピュータ記憶媒体には、磁気テープカセット、フラッシュメモリカード、デジタル多用途ディスク、デジタルビデオテープ、ソリッドステート RAM、ソリッドステート ROM、および類似物が含まれるが、これに限定はされない。ハードディスクドライブ 161 は、通常は、インターフェース 160 などの取外し不能メモリインターフェースを介してシステムバス 141 に接続され、磁気ディスクドライブ 171 および光学ディスクドライブ 175 は、通常は、インターフェース 170 などの取外し可能メモリインターフェースによってシステムバス 141 に接続される。

20

【0025】

上で述べ、図 3 に図示したドライブおよびそれに関連するコンピュータ記憶媒体は、コンピュータ 120 のコンピュータ可読命令、データ構造体、プログラムモジュール、および他のデータのストレージを提供する。図 3 では、たとえば、ハードディスクドライブ 161 が、オペレーティングシステム 164、アプリケーションプログラム 165、他のプログラムモジュール 166、およびプログラムデータ 167 を格納するものとして図示されている。これらのコンポーネントを、オペレーティングシステム 154、アプリケーションプログラム 155、他のプログラムモジュール 156、およびプログラムデータ 157 と同一または異なるいずれかとすることができますことに留意されたい。オペレーティングシステム 164、アプリケーションプログラム 165、他のプログラムモジュール 166、およびプログラムデータ 167 は、最低限でもこれらが異なるコピーであることを示すために、本明細書では異なる符号を与えられている。

30

【0026】

40

ユーザは、キーボード 182、マイクロホン 183、およびマウス、トラックボール、またはタッチパッドなどのポインティングデバイス 181 などの入力デバイスを介してコンピュータ 120 にコマンドおよび情報を入力することができる。他の入力デバイス (図示せず) に、ジョイスティック、ゲームパッド、衛星放送受信用パラボラアンテナ、スキヤナ、または類似物を含めることができる。上記および他の入力デバイスは、しばしば、システムバスに結合されたユーザ入力インターフェース 180 を介して処理ユニット 140 に接続されるが、パラレルポート、ゲームポート、または universal serial serial bus (USB) などの、他のインターフェースおよびバス構造によって接続されることが可能である。モニタ 184 または他のタイプのディスプレイデバイスも、ビデオインターフェース 185 などのインターフェースを介してシステムバス 141 に接続される。モニタの他に、コンピュータに、スピーカ 187 およびプリンタ 186 など、出力周辺インターフェース 188 を介して接続することができる他の周辺出力デバイスをも含めることができる。

50

【0027】

コンピュータ120は、リモートコンピュータ194などの1つまたは複数のリモートコンピュータへの論理接続を使用して、ネットワーク化された環境で動作することができる。リモートコンピュータ194は、パーソナルコンピュータ、ハンドヘルドデバイス、サーバ、ルータ、ネットワークPC、ピアデバイス、または他の一般的なネットワークノードとすることができます、通常は、コンピュータ120に関して上で説明した要素の多数またはすべてを含む。図3に示された論理接続には、ローカルエリアネットワーク(LAN)191および広域ネットワーク(WAN)193が含まれるが、他のネットワークを含めることもできる。そのようなネットワーキング環境は、オフィス、会社全体のコンピュータネットワーク、インターネット、およびインターネットでありふれたものである。

10

【0028】

LANネットワーキング環境で使用される場合に、コンピュータ120は、ネットワークインターフェースまたはネットワークアダプタ190を介してLAN191に接続される。WANネットワーキング環境で使用される場合に、コンピュータ120には、通常は、インターネットなどのWAN193を介する通信を確立する、モデム192または他の手段が含まれる。モデム192は、内蔵または外付けとすることができますが、ユーザ入力インターフェース180または他の適当な機構を介してシステムバス141に接続することができる。ネットワーク化された環境では、コンピュータ120に関して示されたプログラムモジュールまたはその諸部分を、リモートメモリストレージデバイスに格納することができる。制限ではなく例として、図3に、リモートコンピュータ194に常駐するリモートアプリケーションプログラム195を示す。図示のネットワーク接続が、例示的であり、コンピュータの間の通信リンクを確立する他の手段を使用することができますことを了解されたい。

20

【0029】

例示的実施形態

図4に、本明細書で説明する概念と共に使用できるネットワークベースの認識(本明細書では広域ネットワークを用いて例示される)のアーキテクチャ200を示す。しかし、リモートコンポーネントとの相互作用が、リコグナイザを含む音声アプリケーションを、その中にすべての必要なコンポーネントまたはモジュールが存在する単一のコンピュティングデバイス上で動作可能とすることができるという点で、1つの実施形態にすぎないことを理解されたい。

30

【0030】

一般に、ウェブサーバ202に格納された情報には、モバイルデバイス30(本明細書では、入力の形に基づいて、必要に応じてディスプレイスクリーン、マイクロホン、カメラ、接触感知型パネルなどを有する他の形のコンピュティングデバイスも表す)を介して、または電話機80を介して(この場合に、情報は、可聴にまたは押されたキーに応答して電話機80によって生成されるトーンを介して要求され、ウェブサーバ202からの情報は、可聴でのみユーザに戻って供給される)アクセスすることができる。

【0031】

この例示的実施形態では、アーキテクチャ200は、情報が音声認識を使用してデバイス30または電話機80のどちらを介して入手される場合であっても、単一の認識サーバ204が動作の両方のモードをサポートできるという点で、統一されている。さらに、アーキテクチャ200は、周知のマークアップ音声(たとえば、HTML、XHTML、cHTML、XML、WML、および類似物)の拡張を使用して動作する。したがって、ウェブサーバ202に格納された情報に、これらのマークアップ音声に見られる周知のGUI法を使用してアクセスすることもできる。周知のマークアップ音声の拡張を使用することによって、ウェブサーバ202でのオーサリングが、より簡単であり、現在存在するレガシーアプリケーションを、ボイスまたは他の形の認識を含むように簡単に変更することができる。

40

【0032】

50

一般に、デバイス30は、ウェブサーバ202によって供給されるH T M L +スクリプトまたは類似物を実行する。ボイス認識が必要な時には、たとえば、上で述べたようにオーディオ信号がデバイス30によって前処理されている、ディジタル化されたオーディオ信号または音声特徴とすることのできる音声データが、音声認識中に使用すべき文法または言語モデルの表示と共に認識サーバ204に供給される。認識サーバ204の実施態様は、多数の形をとることができ、その1つが図示されているが、一般に、リコグナイザ211を含む。認識の結果を、望まれる場合または適当な場合にローカルレンダリングのためにデバイス30に戻して供給することができる。認識および使用される場合のすべてのグラフィカルユーザインターフェースを介する情報のコンパイルの際に、デバイス30は、さらなる処理および必要な場合のさらなるH T M Lスクリプトの受け取りのために、情報をウェブサーバ202に送信する。

10

【0033】

図4に示されているように、デバイス30、ウェブサーバ202、および認識サーバ204は、一般に、ネットワーク205、ここではインターネットなどの広域ネットワークを介して接続され、別々にアドレッシング可能である。したがって、これらのデバイスのいずれかが、物理的に互いに隣接して配置されることは、必要ではない。具体的に言うと、ウェブサーバ202が認識サーバ204を含むことは、必要ではない。この形で、ウェブサーバ202でのオーサリングは、作成者が認識サーバ204の込み入った事情を知る必要なしに、それが意図されたアプリケーションに焦点を合わせることができる。作成者が認識サーバ204の込み入った事情を知る必要があるのではなく、認識サーバ204を、独立に設計し、ネットワーク205に接続することができ、これによって、ウェブサーバ202でのさらなる変更を必要とせずに、認識サーバ204を更新し、改善することができる。下で述べるように、ウェブサーバ202には、クライアントサイドマークアップおよびスクリプトを動的に生成できるオーサリング機構を含めることもできる。さらなる実施形態で、ウェブサーバ202、認識サーバ204、およびデバイス30を、実装するマシンの機能に応じて組み合わせることができる。たとえば、クライアントが、汎用コンピュータ、たとえばパーソナルコンピュータを含む場合に、クライアントに、認識サーバ204を含めることができる。同様に、望まれる場合に、ウェブサーバ202および認識サーバ204を、単一のマシンに組み込むことができる。

20

【0034】

電話機80を介するウェブサーバ202へのアクセスは、有線または無線の電話網208への電話機80の接続を含み、この接続は、電話機80をサードパーティゲートウェイ210に接続する。ゲートウェイ210は、電話機80をテレフォニボイスブラウザ212に接続する。テレfonボイスブラウザ212は、テレfonインターフェースおよびボイスブラウザ216を提供するメディアサーバ214を含む。デバイス30と同様に、テレfonボイスブラウザ212は、ウェブサーバ202からH T M Lスクリプトまたは類似物を受信する。一実施形態で、H T M Lスクリプトは、デバイス30に供給されるH T M Lスクリプトに類似する形を有する。この形で、ウェブサーバ202は、デバイス30および電話機80を別々にサポートする必要がなく、標準G U Iクライアントを別々にサポートする必要さえない。そうではなく、共通のマークアップ言語を使用することができる。さらに、デバイス30と同様に、電話機80によって送信される可聴信号からのボイス認識は、ボイスブラウザ216から認識サーバ204へ、たとえばT C P / I Pを使用して、ネットワーク205を介してまたは専用回線207を介してのいずれかで供給される。ウェブサーバ202、認識サーバ204、およびテレfonボイスブラウザ212は、図3に示された汎用デスクトップコンピュータなどの任意の適切なコンピューティング環境で実施することができる。

30

【0035】

しかし、D T M F認識が使用される場合に、この形の認識が、一般に、認識サーバ204ではなくメディアサーバ214で実行されることに留意されたい。言い換えると、D T M F文法は、メディアサーバ214によって使用されるはずである。

50

【 0 0 3 6 】

図4に戻ると、ウェブサーバ202には、サーバサイドプラグインオーサリングツールまたはモジュール209（たとえば、Microsoft CorporationによるASP、ASP+、およびASP.NET、JSP、Java（登録商標）beans、または類似物）を含めることができる。サーバサイドプラグインモジュール209は、クライアントサイドマークアップを動的に生成でき、ウェブサーバ202にアクセスするクライアントのタイプ用のマークアップの特定の形すら動的に生成することができる。クライアント情報は、クライアント／サーバ関係の最初の確立時にウェブサーバ202に供給することができ、あるいは、ウェブサーバ202に、クライアントデバイスの機能を検出するモジュールまたはルーチンを含めることができる。この形で、サーバサイドプラグインモジュール209は、ボイス認識シナリオごとに、すなわち電話機80を介してボイスのみまたはデバイス30についてマルチモーダルで、クライアントサイドマークアップを生成することができる。一貫するクライアントサイドモデルを使用することによって、多数の異なるクライアント用のアプリケーションオーサリングが、大幅により単純になる。

10

【 0 0 3 7 】

クライアントサイドマークアップの動的生成に加えて、下で述べる高水準ダイアログモジュールを、アプリケーションオーサリングでの開発者による使用のためにストア211に格納されるサーバサイドコントロールとして実装することができる。一般に、高水準ダイアログモジュール211は、開発者によって指定されるパラメータに基づいて、ボイスのみのシナリオとマルチモーダルシナリオとの両方でクライアントサイドマークアップおよびスクリプトを動的に生成する。高水準ダイアログモジュール211には、開発者の必要に合わせるために、クライアントサイドマークアップを生成するためのパラメータを含めることができる。

20

【 0 0 3 8 】**クライアントサイドマークアップの生成**

上で示したように、サーバサイドプラグインモジュール209は、クライアントデバイス30から要求が行われた時に、クライアントサイドマークアップを出力する。手短に言うと、サーバサイドプラグインモジュール209は、ウェブサイトを、したがってアプリケーションおよびそのアプリケーションによって提供されるサービスを、定義しました構築することを可能にする。サーバサイドプラグインモジュール209内の命令は、コンパイルされたコードから作られる。このコードは、ウェブ要求がウェブサーバ202に達する時に実行される。次に、サーバサイドプラグインモジュール209が、新しいクライアントサイドマークアップページを出力し、このクライアントサイドマークアップページが、クライアントデバイス30に送信される。周知の通り、このプロセスを、一般にレンダリングと称する。サーバサイドプラグインモジュール209は、マークアップ言語を抽象化し、カプセル化する「コントロール」を操作し、したがってクライアントサイドマークアップページのコードを操作する。マークアップ言語を抽象化し、カプセル化し、ウェブサーバ202上で動作するそのようなコントロールは、たとえば、「サーブレット」または「サーバサイドプラグイン」を含み、またはこれらと同等である。

30

【 0 0 3 9 】

既知の通り、従来技術のサーバサイドプラグインモジュールは、ビジュアルレンダリングおよびクライアントデバイス30との相互作用に関するクライアントサイドマークアップを生成することができる。特許出願書に、認識拡張および可聴プロンプト拡張を含めるためのサーバサイドプラグインモジュール209の拡張に関する3つの異なる手法が詳細に記載されている（たとえば、特許文献1および特許文献2参照）。本発明の諸態様は、これらの手法のすべてと共に使用することができるが、1つの手法の短い説明を、例示的実施形態を説明するために下で提供する。

40

【 0 0 4 0 】

図5を参照すると、認識／可聴プロンプトコントロール306は、ビジュアルコントロ

50

ール302と別々であるが、下で述べるように選択的にこれに関連する。この形で、コントロール306は、ビジュアルコントロール302に直接に基づいて構築されるのではなく、ビジュアルコントロール302を書き直す必要なしに認識／可聴プロンプト使用可能化を提供する。コントロール306は、コントロール302と同様に、ライブラリ300を使用する。この実施形態で、ライブラリ300は、ビジュアルプロンプトマークアップ情報と認識／可聴プロンプトマークアップ情報との両方を含む。

【0041】

この手法には、大きい利益がある。第1に、ビジュアルコントロール302は、内容において変更される必要がない。第2に、コントロール306は、一貫性があり、音声対応コントロール302の性質に従って変更される必要がない単一のモジュールを形成することができる。第3に、音声使用可能化のプロセスすなわち、ビジュアルコントロール302とのコントロール306の明示的な関連付けは、明示的かつ選択的なプロセスなので、設計時に完全に開発者の制御の下にある。これは、ビジュアルコントロールのマークアップ言語が、コントロール306によって生成されたマークアップ言語によって提供される認識を介してまたはキー・ボードなどの従来の入力デバイスを介してなど、複数のソースからの入力値を受け取ることを可能にもする。短く言うと、コントロール306を、サーバサイド・プラグイン・モジュール209のビジュアルオーサリング・ページの既存アプリケーションオーサリング・ページに追加することができる。コントロール306は、クライアント・デバイス30のユーザに対話の新しいモーダリティ（すなわち、認識および／または可聴プロンプト）を提供すると同時に、ビジュアルコントロールのアプリケーション・ロジックおよびビジュアル入出力機能を再利用する。コントロール306を、アプリケーション・ロジックをコーディングできるビジュアルコントロール302に関連付けることができることを考慮して、コントロール306を、以下では「コンパニオン・コントロール306」と称し、ビジュアルコントロール302を、「プライマリ・コントロール302」と称する。これらの言及が、コントロール302および306を区別するために提供され、制限的であることは意図されていないことに留意されたい。たとえば、コンパニオン・コントロール306を使用して、ボイスのみウェブサイトなどのビジュアルレンダリングを含まないウェブサイトを開発し、またはオーサリングすることができる。その場合に、ある種のアプリケーション・ロジックを、コンパニオン・コントロール・ロジックで実施することができる。

【0042】

コンパニオン・コントロール400の例示的セットを、図6に示す。この実施形態では、コンパニオン・コントロール400は、一般に、QA・コントロール402、Command・コントロール404、CompareValidator・コントロール406、Custom Validator・コントロール408、およびsemantic map410を含む。semantic map410は、概略的に示されており、semantic item412を含み、semantic item412は、ビジュアルドメイン・プライマリ・コントロール402（たとえば、コンパニオン・コントロール400のHTMLおよび非ビジュアル認識ドメインの間のレイヤ）を形成する、入力フィールドと考えることができる。

【0043】

QA・コントロール402は、出力・コントロールの機能を実行するPromptオブジェクトを参照する、すなわち、ヒューマンダイアログ用の「プロンプトする」クライアント・サイド・マークアップを提供するPromptプロパティを含み、このクライアント・サイド・マークアップは、通常、直接にマークアップに含まれるかURLを介して参照されるデータである事前に録音されたオーディオファイルまたはテキスト・音声変換用のテキストを再生することを含む。同様に、入力・コントロールは、QA・コントロール402およびCommand・コントロール404として実施され、ヒューマンダイアログにも従い、Promptプロパティ（Promptオブジェクトを参照する）と少なくとも1つのAnswerオブジェクトを参照するAnswerプロパティとを含む。QA・コントロール402とCommand・コントロール404との両方が、クライアント・デバイス30のユーザか

10

20

30

40

50

らの期待される入力または可能な入力に文法を関連付ける。

この点で、コントロールのそれぞれの短い説明を提供することが役に立つ可能性がある。

【0044】

Q A コントロール

一般に、Q A コントロール 402 は、図示のプロパティを介して、たとえば、出力可聴プロンプトの提供、入力データの収集、入力結果の信頼性妥当性検査の実行、入力データの確認の可能化、およびウェブサイトでのダイアログフローの制御の援助のうちの 1 つまたは複数を実行することができる。言い換えると、Q A コントロール 402 は、特定のトピックのコントロールとして機能するプロパティを含む。

10

【0045】

Q A コントロール 402 は、他のコントロールと同様に、ウェブサーバ 202 上で実行され、これは、Q A コントロール 402 が、サーバサイドマークアップフォーマリズム (A S P、J S P、または類似物) を使用してウェブサーバ上で保持されるアプリケーション開発ウェブページで定義されるが、異なる形のマークアップとしてクライアントデバイス 30 に出力されることを意味する。Q A コントロールがプロパティ Prompt、Reco、Answers、ExtraAnswers、およびConfirms のすべてから形成されるように見える図 6 に図示されてはいるが、これらが、単にオプションであり、これらのうちの 1 つまたは複数を Q A コントロールのために含めることができることを理解されたい。

20

【0046】

この点で、アプリケーションシナリオについて Q A コントロール 402 の使用を説明することが役に立つ可能性がある。図 6 を参照すると、ボイスのみアプリケーションにおいて、Q A コントロール 402 は、ダイアログ内の質問および回答として機能することができる。質問は、Prompt オブジェクトによって提供され、文法は、入力データの認識およびその入力に対する関連する処理のための文法オブジェクトを介して定義される。Answers プロパティは、Answer オブジェクトを使用して、認識された結果を Semantic Map 410 内の SemanticItem 412 に関連付け、Answer オブジェクトは、認識結果をどのように処理するかに関する情報を含む。線 414 は、Semantic Map 410 およびその中の SemanticItem 412 との Q A コントロール 402 の関連付けを示す。多数の SemanticItem 412 が、線 418 によって表されるようにビジュアルコントロールまたはプライマリコントロール 302 に個別に関連するが、1 つまたは複数の SemanticItem 412 を、ビジュアルコントロールに関連せず、内部的にのみ使用されるものとすることができる。クライアントデバイス 30 のユーザが、たとえば「TapEvent」を用いてビジュアルテキストボックスに触れることができるマルチモーダルシナリオでは、可聴プロンプトが、必要でない場合がある。たとえば、クライアントデバイスのユーザが対応するフィールドに何を入力しなければならないかの表示を形成するビジュアルテキストを有するテキストボックスを含むプライマリコントロールについて、対応する Q A コントロール 402 は、オーディオ再生またはテキスト - 音声変換などの対応するプロンプトを有しても有しなくてよいが、認識に関する期待される値に対応する文法と、入力を処理するか、音声が検出されない、音声が認識されない、またはタイムアウト時に発火されるイベントなどの他のリコグナイザイベントを処理するイベントハンドラとを有する。

30

【0047】

さらなる実施形態では、認識結果が、その認識結果が正しかったことの信頼性のレベルを示す信頼性レベル測定値を含む。確認閾値を、Answer オブジェクト内で、たとえば 0.7 と等しい ConfirmThreshold として指定することもできる。確認レベルが、関連する閾値を超える場合に、その結果を、確認されたと考えることができる。

40

【0048】

50

それに加えてまたはその代わりに、音声認識の文法を指定するために、QAコントロールおよび/またはCommandコントロールが、Dtmf（デュアルトーン変調周波数）文法を指定して、プロンプトまたは質問に応答する電話機キーアクティブ化を認識できることにも留意されたい。

【0049】

この点で、Semantic Map 410のSemantic Item 412が、たとえば認識を介して音声またはDtmfを充たされた時に、複数のアクションを行えることに留意されたい。第1に、値が「変更された」ことを示すイベントを、発行するか発火させることができる。確認レベルが満足されたかどうかに応じて、発行または発火できるもう1つのイベントに、対応するセマンティックアイテムが確認されたことを示す「確認」イベントが含まれる。これらのイベントは、ダイアログを制御するのに使用される。

10

【0050】

Confirmsプロパティには、Semantic Item 412に関連し、望まれる場合にConfirmThresholdを含むことができるという点で、Answersプロパティに関して上で説明したものに類似する構造を有する回答オブジェクトを含めることもできる。Confirmsプロパティは、認識結果自体を含むことが意図されているのではなく、既に含まれる結果を確認し、含まれる結果が正しいかどうかをユーザから確かめることができている。Confirmsプロパティは、前に得られた結果の値が正しかったかどうかをアサートするのに使用されるAnswerオブジェクトのコレクションである。含むQAのPromptオブジェクトは、これらのアイテムについて照会し、関連するSemantic Item 412から認識結果入手し、これを「Did you say Seattle?（シアトルと言いましたか？）」などの質問に形成する。ユーザが、「Yes（はい）」などの肯定で応答する場合に、確認済みイベントが発火される。ユーザが、「No（いいえ）」などの否定で応答する場合に、関連するSemantic Item 412がクリアされる。

20

【0051】

Confirmsプロパティは、確認プロンプトがユーザに提供された後に、訂正を受け入れることもできる。たとえば、確認プロンプト「Did you say Seattle？」に応答して、ユーザが、「San Francisco（サンフランシスコ）」、または「No, San Francisco（いいえ、サンフランシスコです）」と応答することができ、この場合に、QAコントロールは、訂正を受け取っている。Answerオブジェクトを介してどのSemantic Itemが確認されているかに関する情報を有するので、Semantic Item内の値を、訂正された値に置換することができる。望まれる場合に、「When did you want to go to Seattle?（いつシアトルに行きたいですか？）」などの情報に関して、さらなるプロンプトに確認を含めることができ、ここで、システムによるこのプロンプトは、「Seattle」に関する確認と、出発の日付に関するさらなるプロンプトとを含むことに留意されたい。目的地の場所に対する訂正を提供するユーザによる応答は、Confirmsプロパティをアクティブ化して、関連するセマンティックアイテムを訂正し、出発日だけを伴う応答は、目的地の暗黙の確認を提供する。

30

【0052】

ExtraAnswersプロパティは、ユーザが既に行われたプロンプトまたは照会に加えて提供できるAnswerオブジェクトを、アプリケーション作成者が指定することを可能にする。たとえば、旅行指向システムが、目的地の都市に関してユーザにプロンプトを出すが、ユーザが「Seattle tomorrow（シアトル、明日）」を示すことによって応答する場合に、当初にそのユーザにプロンプトを出したAnswersプロパティは、目的地都市「Seattle」を取り出し、したがってこれを適当なSemantic Itemにバインドするが、ExtraAnswersプロパティは、次に続く日として「Tomorrow」を処理することができ（システムが現在の日付を知っ

40

50

ていると仮定して)、これによって、この結果を Semantic Map 内の適当な Semantic Item にバインドすることができる。Extra Answers プロパティは、ユーザがそれについても述べる可能性がある可能な余分な情報について定義された 1 つまたは複数の Answer オブジェクトを含む。上で提供した例では、出発日に関しても情報を取り出したので、確認レベルが対応する ConfirmThreshold を超えると仮定すると、このシステムは、この情報についてユーザに再びプロンプトを出す必要がない。確認レベルが対応する閾値を超えた場合には、適当な Confirms プロパティがアクティブ化されるはずである。

【0053】

Command コントロール

Command コントロール 404 は、通常は問われた質問に関する意味的インポートをほとんど有しておらず、むしろ、たとえばヘルプ、キャンセル、繰り返しなど、援助を求めるかナビゲーションをもたらす、ボイスのみダイアログで一般的なユーザ発話である。Command コントロール 404 には、プロンプトオブジェクトを指定する Prompt プロパティを含めることができる。さらに、Command コントロール 404 は、文法 (Grammar プロパティを介する) および認識に対する関連する処理 (どちらかといえば、Semantic Item への結果のバインディングなしの Answer オブジェクトに似る) だけではなく、コンテキストの「スコープ」およびタイプを指定するにも使用することができる。これは、クライアントサイドマークアップでのグローバル拳動とコンテキスト感応型拳動との両方のオーサリングを可能にする。Command コントロール 404 は、「ヘルプ」コマンドまたは、クライアントデバイスのユーザがウェブサイトの他の選択されたエリアへナビゲートすることを可能にするコマンドなど、追加のタイプの入力を可能にする。

【0054】

CompareValidator コントロール

CompareValidator コントロールは、演算子に従って 2 つの値を比較し、適当なアクションを行う。比較される値は、整数、テキストのストリングなど、任意の形とすることができます。CompareValidator は、妥当性検査される Semantic Item を示すプロパティ SemanticItemToValidate を含む。妥当性検査される Semantic Item を、定数または別の Semantic Item と比較することができ、ここで、定数または他の Semantic Item は、それぞれプロパティ ValueToCompare および SemanticItemToCompare によって供給される。CompareValidator に関する他のパラメータまたはプロパティには、行われる比較を定義する Operator と、たとえば整数またはセマンティックアイテムのストリングなど、値のタイプを定義する Type とが含まれる。

【0055】

CompareValidator コントロールに関する妥当性検査が不合格の場合に、Prompt プロパティは、得られた結果が正しくなかったことをユーザに指示する、再生できる Prompt オブジェクトを指定することができる。比較の際に妥当性検査が不合格になる場合には、SemanticItemToValidate によって定義された関連する Semantic Item は、システムが正しい値についてユーザに再びプロンプトを出すために、空であるものとして示される。しかし、正しくない値が、正しくない値を繰り返すユーザへのプロンプトに使用される場合に、Semantic Map 内の関連する Semantic Item の正しくない値をクリアしないことが役に立つ場合がある。アプリケーション作成者の望みに応じて、CompareValidator コントロールを、関連する Semantic Item の値が値を変化させる時またはその値が確認された時のいずれかにトリガすることができます。

【0056】

CustomValidator コントロール

10

20

30

40

50

`CustomValidator` コントロールは、`CompareValidator` コントロールに似ている。プロパティ `SemanticItem` を示し、プロパティ `ClientValidationFunction` は、関連する関数またはスクリプトを介してカスタム妥当性検査ルーチンを指定する。この関数は、妥当性検査が不合格であったか否かのブール値「`yes`」もしくは「`no`」またはその同等物を提供する。`Prompt` プロパティは、妥当性検査のエラーまたは不合格の表示を提供する `Prompt` オブジェクトを指定することができる。アプリケーション作成者の望みに応じて、`CustomValidator` コントロールを、関連する `SemanticItem` の値が値を変化させる時またはその値が確認された時のいずれかにトリガることができる。

10

【0057】

コントロール実行アルゴリズム

クライアントサイドスクリプトまたはクライアントサイドモジュール（本明細書では「`RunSpeech`」と称する）が、図6のコントロールに関してクライアントデバイスに提供される。このスクリプトの目的は、ロジックを介してダイアログフローを実行するためであり、このロジックは、スクリプトがクライアントデバイス30で実行される時、すなわち、コントロールに関するマークアップがクライアントに含まれる値に起因してクライアントでの実行のためにアクティビ化される時に、スクリプトで指定される。このスクリプトは、ページ要求の間での複数のダイアログターンを可能にし、したがって、テレフォニプラウザ216を介するなど、ボイスのみダイアログのコントロールに特に役立つ。クライアントサイドスクリプト `RunSpeech` は、完成したフォームがサブミットされるか、新しいページが他の形でクライアントデバイス30から要求されるまで、クライアントデバイス30上でループの形で実行される。

20

【0058】

一般に、一実施形態で、このアルゴリズムは、音声を出力し、ユーザ入力を認識することによって、ダイアログターンを生成する。このアルゴリズムの全体的なロジックは、ボイスのみシナリオについて、次の通りである（他の形で上で述べられていないプロパティまたはパラメータについては、特許出願（たとえば、特許文献1参照）を参照されたい）。

1. 音声インデックス順で、第1のアクティブな（下で定義する）QAコントロール、`CompareValidator` コントロール、または`CustomValidator` コントロールを見つける。

30

2. アクティブコントロールがない場合には、ページをサブミットする。

3. そうでない場合には、コントロールを実行する。

【0059】

QAは、次の場合に限ってアクティブと考えられる。

1. QAの `clientActivationFunction` が、存在しないか `true` を返すかのいずれかであり、かつ、

40

2. `Answers` プロパティコレクションが空でない場合に、`Answers` のセットによってポイントされるすべての `SemanticItems` の `State` が `Empty` であり、または、

3. `Answers` プロパティコレクションが空である場合に、`Confirm` アレイ内の少なくとも1つの `SemanticItem` の `State` が `NeedsConfirmation` である。

しかし、QAが、`true` の `PlayOnce` を有し、その `Prompt` が成功して実行された（`OnComplete` に達した）場合に、そのQAは、アクティビ化の候補ではない。

【0060】

QAは、次のように実行される。

1. これが、前のアクティブコントロールと異なるコントロールである場合には、プロ

50

ンプトCount値をリセットする。

2. Promptカウント値を増分する。

3. PromptSelectFunctionが指定されている場合には、その関数を呼び出し、返されたストリングをPromptのinlinePromptにセットする。

4. Recoオブジェクトが存在する場合には、それを開始する。このRecoは、既にアクティブコマンド文法を含んでいなければならない。

【0061】

Validator(CompareValidatorまたはCustomValidatorのいずれか)は、次の場合にアクティブである。10

1. SemanticItemToValidateが、このバリデータによって妥当性検査されておらず、その値が変化している。

【0062】

CompareValidatorは、次のように実行される。

1. バリデータのOperatorに従って、SemanticItemToCompareまたはValueToCompareの値とSemanticItemToValidateの値とを比較する。

2. このテストがfalseを返す場合には、SemanticItemToValidateのテキストフィールドを空にし、プロンプトを再生する。

3. このテストがtrueを返す場合には、このバリデータによって妥当性検査されたものとしてSemanticItemToValidateをマークする。20

【0063】

CustomValidatorは、次のように実行される。

1. SemanticItemToValidateの値を用いてClientValidationFunctionを呼び出す。

2. この関数がfalseを返す場合には、semanticItemをクリアし、プロンプトを再生する。そうでない場合には、このバリデータによって妥当性検査される。

【0064】

Commandは、次の場合に限ってアクティブと考えられる。

1. それがScope内にあり、かつ、30

2. スコープツリー内でより下に同一Typeの別のCommandがない。

【0065】

マルチモーダルの場合に、このロジックは、次のアルゴリズムに単純化される。

1. トリガするイベントすなわち、ユーザがコントロールをタップするのを待つ。

2. 期待される回答を収集する。

3. 入力を聞く。

4. 結果をSemanticItemにバインドする。結果がない場合には、イベントを送出する。

5. 1に戻る。

【0066】

マルチモーダル環境では、ユーザが結果のビジュアルプレゼンテーションに関連するテキストボックスまたは他の入力フィールドを訂正する場合に、このシステムが、値が確認されたことを示すために、関連するSemanticItemを更新できることに留意されたい。

【0067】

さらなる実施形態では、図6に示されているように、アプリケーション作成者がテレフォニートランザクションを処理する音声アプリケーションを作成することを可能にするcallコントロール407、ならびに共通の音声シナリオを1つのコントロールにラップする手段を提供するapplicationコントロール430が設けられる。callコントロール407およびapplicationコントロール430は、本発明の実線に40

必要ではなく、単に完全を期して言及されたものである。このそれぞれのさらなる議論は、特許出願（たとえば、特許文献1および特許文献2参照）で提供される。

【0068】

ユーザ対話データの記録

例として前述の構造を使用することによって、アプリケーション開発者は、音声対応アプリケーションを開発することができる。しかし、本明細書で説明する諸態様は、開発者がユーザ対話データを記録したまはロギングすることを可能にする。

【0069】

それでも、本明細書で説明する概念は、ダイアログモデルを提供する、上で説明したダイアログオーサリング構造に限定されるのではなく、ミドルウェア、API（アプリケーションプログラムインターフェース）または類似物として実装されるものなどであるがこれらに限定はされないダイアログモデルを生成する、下で説明する情報の一部またはすべてを記録するように構成された任意のオーサリングツールに適用できることを理解されたい。さらに、テレフォニアアプリケーションなどの音声対応アプリケーションの機能的性質およびそのボイスユーザインターフェースの詳細は、ドメインおよびアプリケーションタイプにまたがって大きく異なる可能性があり、したがって、通常可能にされる任意の自動化ロギングは、ヒューリスティックに過ぎず、決定的ではない。この理由から、これの実装は、変更不能なプロパティではなく、オーバーライド可能なデフォルトとして自動化ログイベントプロパティを実装する可能性が高い。それでも、豊富な情報のロギングを単純化し、容易にすることは、手動オーサリングおよびプログラム的オーサリングに頼るシステムに対する大きい進歩である。

10

20

20

【0070】

戻って図4を参照すると、ダイアログコントロール211に従って音声対応アプリケーションを実行するウェブサーバ202は、モバイルデバイス30を介するアクセスまたは電話機80を介するアクセスなどであるがこれらに限定されない任意のタイプのユーザのためにアプリケーションが実行される時に、ストア217にユーザ対話ログデータを記録する。

【0071】

アプリケーションは、一般に、それほど排他的にではなく、本明細書で通常は必要に応じてCommandコントロール404、Applicationコントロール430、Callコントロール407、ならびにバリデータ406および408と共にQAコントロール402によって例示される階層コントロールのセットとして定義され、または記述される。この階層は、完了されなければならない全体的タスクならびにその全体的タスクを完了するためのその全体的タスクのサブタスクを定義する。この階層内のレベルの個数は、アプリケーションの複雑さに依存する。たとえば、アプリケーションを、全体的に航空便予約に向ける（すなわち最上位タスク）ことができ、2つの主要なサブタスクは、出発情報および到着情報の入手に向けられる。同様に、さらなるサブタスクを、具体的には、出発／到着空港情報、出発／到着時刻などを入手するサブタスクを、出発情報の入手および到着情報の入手という主要なサブタスクのそれぞれについて定義することができる。これらのサブタスクは、それを含むタスク内のシーケンス内に現れるものとすることができます。

30

40

【0072】

一般に、2タイプのデータすなわち、Task / DialogデータおよびTurnデータが記録される。Task / Dialogデータから始ると、ログ内で表されるこのデータは、タスクおよびサブタスクに関するアプリケーションの階層構造またはシーケンシャル構造を取り込まなければならない。図7に、アプリケーションを作成する方法500を示す。ダイアログオーサリングツールは、ステップ502で、ネストされたTask単位またはシーケンシャルTask単位に関してダイアログのオーサリングまたは定義を可能にし、その結果、開発者が音声対応アプリケーションを記述する時に、その作成者は、通常、その音声対応アプリケーションをモジュラの形で記述するようになる。すなわち

50

、作成者は、個々の Turn を特定の Task を達成するセットにグループ化し、個々のタスクをより上位の Task を達成するセットにグループ化することを奨励される。 Task 構造および個々の Task に出入りするフローは、設計時に既知なので、ステップ 504 で、 Task に入ることまたは Task から出ることのロギング（たとえば、 Task Start イベントおよび Task Complete イベントを介する）ならびにアプリケーションによって使用される入力フィールドに関してユーザから入手される Turn データおよび値のロギング（本明細書では「セマンティックアイテム」として例示される）を可能にして、 Task 構造のシーケンスおよび / または階層の自動化ロギングを提供する。これは、ダイアログフロー、入手された値、および Task 構造を、イベントログから明示的に回復でき、構築できることを意味する。ステップ 502 および 504 が、これらのステップの特徴の一部またはすべてを異なる順序でまたは同時に実行できるという点で、例示のみを目的として別々に示されていることに留意されたい。

10

【0073】

このデータは、任意の所与のタスクまたはサブタスクの完了の成功、失敗、または他（たとえば未知）という状況をも定量化する。さらに、 Task / Dialog データは、タスクが不成功もしくは失敗の場合の理由、またはその完了状況が既知でない理由、または適用可能な場合に複数の理由が成功について可能である場合に成功の理由を含む。追加データには、ユーザが応答を供給しなかった場合または音声リコグナイザが発話を認識できなかった場合を示す進行状況データを含めることができる。プロンプトまたはユーザ応答に基づくかこれに関連する値に関してアプリケーションによって使用される入力フィールド値またはストレージロケーションのリスト、あるいは変化したその状況をも、記録することができる。

20

【0074】

図 8 に、音声対応アプリケーションを実行する方法 520 を示す。方法 520 は、ステップ 522 で、1つまたは複数の Turn を有する Task に関する定義された音声対応アプリケーションを実行することを含む。ステップ 524 は、 Task 、 Turn 、およびセマンティックアイテムに関する情報を記録することを含む。ステップ 522 および 524 が、これらのステップの特徴の一部またはすべてを異なる順序でまたは同時に実行できるという点で、例示のみを目的として別々に示されていることに留意されたい。

30

【0075】

一実施形態で、 Task / Dialog データは、次の情報の一部またはすべてを含む。

Task / Dialog データ

`name` : タスク / ダイアログの作成者定義のストリング識別子、たとえば「get CreditCardInfo」、「ConfirmTravel」など。作成者が設計時に名前を供給しない場合には、たとえば Dialog1 、 Dialog2 、 DialogN 、 ... など、デフォルト名が与えられる。

`parent` : 含むダイアログの名前（ログからダイアログ階層を再構成するために）。

40

`TaskStart` : Task / Dialog に最初に入った時のタイムスタンプ。

`TaskComplete` : Task / Dialog から出た時のタイムスタンプ。このイベントは、デフォルト値を有するアプリケーションが閉じられる時のすべての開かれているダイアログについて、ボトムアップで必ず発火されなければならない（すなわち、ログに「オープンエンド」のダイアログはない）。

`status` : タスク / ダイアログの完了状況であり、作成者によってセット可能であり、ダイアログの実行に基づいて自動的に推論され、あるいは、作成者定義の条件に基づいて半自動的にセットされる。一実施形態で、デフォルト値状況を「UNSET」とすることができ、後続の値は、次のうちの 1 つとすることができます。

SUCCESS

FAILURE

50

U N K N O W N

【 0 0 7 6 】

自動タスク完了状況

ある種の場合に、上で示したように、状況を、タスク出口の性質から、その状況が成功、失敗、または未知のうちの1つであったかどうかを穏当な確かさで推論することができる。たとえば、エラーまたは例外の結果として終了するタスクを、Failureの完了状況と共に自動的にロギングすることができる。同様に、キャンセルされたタスク（たとえば、Cancel（）がタスクオブジェクトに対して呼び出された場合）を、Failureの完了状況と共に自動的にロギングすることができる。同様に、ある種の「ストライクアウト」（たとえば、下で述べるMaxSilencesまたはMaxNoReco）カウントに達したの結果として終了するタスクを、Failureの完了状況と共に自動的にロギングすることができる。

10

【 0 0 7 7 】

対照的に、そのタスクで出会うまたは設計時にそのタスクに属するものとして指定されたTurnのすべてのセマンティックアイテム（すなわち、アプリケーションの入力フィールド）が根拠のある（ユーザ入力またはそれから導出された）値を有する、自然に終了する（すなわち、キャンセルされていない）タスクは、Successの完了状況と共に自動的にロギングされる。

20

【 0 0 7 8 】

半自動化されたタスク完了

タスク状況ロギングの部分的自動化も、有用である。任意の所与のタスクについて、作成者は、満足された場合に出口の任意の点でのタスクの状況を決定する、タスクの成功または失敗に関する条件のセットをステップ502で指定しまたは定義することができる。これらの条件は、プログラム的（たとえば、foo == 'bar'）とすることができ、あるいは、より役立つことに、作成者がタスクごとに1つまたは複数のセマンティックアイテム（たとえば、departureCityおよびarrivalCityについて提供される値）だけを指定する必要があるようになるように条件を単純化することができ、本システムは、これらのセマンティックアイテムが確認された値を有する時にSuccessを自動的にロギングし、オプションで、これらのセマンティックアイテムが確認された値を有しない時にFailureを自動的にロギングする。

30

【 0 0 7 9 】

この態様は、タスク状況ロギングがタスクからのすべての出口点でプログラム的にコーディングされる必要がないことを意味するので、有用な、時間を節約する機構である。その代わりに、条件は、エンドユーザがタスクを終了する時に、必ず自動的に評価され、状況が、判定され、余分な開発者コードなしでロギングされる。

reason : 作成者によってセットできる、ダイアログの完了の理由。たとえば、

Command - ダイアログの異なる部分に対して変更するためにユーザによって話されたコマンドおよびそのコマンドの性質（たとえば、「Cancel」、「Operator」、「Main Menu」など）。

userHangup - ユーザが、ハングアップするか、他の形で放棄するかあきらめた。

40

applicationError - アプリケーションエラーが発生した。

maxNoRecons - 認識に達することなく発話の最大回数に達した。

maxSilences - 無音ユーザ応答の最大回数に達した。

【 0 0 8 0 】

SemanticUpdate :

items : 新しい値および対応する状況を含む、その値 / 状況が変更されたすべてのセマンティックアイテムのリスト。通常、このデータは、各ダイアログターン（アプリケーションによるプロンプト / ユーザによる応答または応答の欠如）に関して、1つまたは複数のセマンティックアイテム値および / または状況が変化するという点で、下で述べる

50

`Turn` データに相関される。しかし、いくつかの場合に、アプリケーションは、それ自体でセマンティックアイテムを変更することができる。たとえば、アプリケーションが、クレジットカード番号などの値を妥当性検査することができない場合に、そのアプリケーションは、必ずしもダイアログターンに基づくのではなく、それ自体でその値をクリアすることができる。それでも、そのような変更は記録されるはずである。

【0081】

`Turn` データは、アプリケーションとの直接対話を含み、アプリケーションによって提供されるプロンプト（応答が期待されない場合）またはユーザ応答もしくはユーザ応答の欠如に相関されるアプリケーションプロンプト、言い換えると、プロンプト／応答交換、または必ずしもプロンプトに応答するのではなくユーザによって提供されるコマンド、または少なくともプロンプトに対する応答であると期待されない応答に基づいて編成される。したがって、記録され得るデータの3つの領域は、アプリケーションによって提供されるプロンプト、ユーザによって供給される応答（期待される応答であれ期待されない応答であれ）、およびシステムによって決定される認識結果に関する情報を含む。一実施形態で、`Turn` データは、次の情報の一部またはすべてを含む。

10

【0082】

`Turn` データ

`config`

`name` : 作成者定義のストリング識別子。作成者が設計時に名前を供給しない場合には、デフォルト名を与えることができる。しかし、同一の `Dialog / Task` 内の異なるターンの間で明瞭かつ一貫して区別する必要がある。可能な技法は、プロンプトの名前およびタイプに基づくことである。

20

`type` : 特定の `Turn` の目的の指定を、それに関連するセマンティックアイテムの性質から推論することができる。上の前述の説明の場合に、セマンティックアイテムは、`Answers`、`ExtraAnswers`、および `Confirms` の概念を介して `Turn` に関連する。

`Turn` の目的の例には、次が含まれる。

新しい情報を求める（`Turn` は `Answers` を使用可能にする）

関連情報を確認する（受け入れ／拒否、`Turn` は `Confirms` を使用可能にする）

30

情報ステートメントを与える（`Turn` は `Answers` または `Confirms` を保持しない）。

`parent` : 含む `Dialog / Task` の名前（ログからダイアログ階層を再構成するため）。

`language` : 使用される言語。

`speech grammars` : どの音声認識文法が使用されているかに関する情報

。

`D M T F grammars` : どの DTMF 認識文法が使用されているかに関する情報

。

`thresholds` : 値の拒絶および／または値の確認に関する信頼性閾値。

40

`timeouts` : プロンプトに続く最初の無音について許容される時間期間、応答の終りを判定する終了無音、およびバブルと考えられる時間期間。

【0083】

`prompt`

`name` : オプション。ターンデータ名を使用できるという点で必要ではない場合がある。

`type` : ダイアログモデルは、そのどれであってもアプリケーションによって選択することができ、それを使用することによってシステムが試みているものの記録を達成することを可能にする、複数の事前定義のプロンプトタイプすなわち `Turn` の目的を含むことができる。

50

プロンプトタイプの例には、次が含まれる。

Main Prompt - 質問をすること（またはステートメントを与えること）

Help Prompt - ヘルプを提供すること

Repeat Prompt - 情報内容を繰り返すこと

No Recognition Prompt - 「認識なし」に応答すること

Silence Prompt - 無音に応答すること

Escalated No Recognition Prompt - 複数の試行の後に「認識なし」に応答すること

Escalated Silence Prompt - 複数の試行の後に無音に応答すること

10

これらの type は、事前定義であり、任意の時の選択に使用可能なので、 type によって自動的にロギングことができ、これによって、 Turn の目的を達成するための所与のプロンプトの目的という概念を用いてログデータを自動的に豊かにする。

したがって、すべてがダイアログオーサリングモデルのプログラミングプリミティブであり、したがってアプリケーションが出会った時に自動的にロギングされる、 Turn タイプと組み合わされるプロンプトタイプは、ログ内の任意の点でのシステムの目的の豊かなビューを可能にする。

semantic items : それに関するプロンプトを出される（質問 / 確認サイクルをリンクすることなどに使用される）セマンティックアイテム。

ダイアログモデルは、ダイアログフローに関してオーサリングを単純化するために、それが値および状況を含むセマンティックアイテムという概念を使用する。すべてのセマンティックアイテムの変化する値および状況を自動的にロギングし、それをタスクおよびユーザ / システム移動情報と組み合わせることによって、ログが、さらに豊かにされる。

Answers / Extra Answers / Confirms モデルは、セマンティックアイテムを Turn に、したがって Task にリンクする。したがって、どのセマンティックアイテムがどのシステム移動およびユーザ移動に関係し、どのセマンティックアイテムがどの Task に寄与するかがわかる（自動的にロギングができる）。

プロンプトのテキスト内容：たとえば「 welcome 」。

bargein : オン / オフ / プロンプト中時間。

30

User Perceived Latency : ユーザの応答と次のプロンプトの再生との間の時間期間。システムが重い負荷の下にある時には、この時間期間がより長くなる場合があり、これは、アプリケーションが応答していないとユーザに思わせる可能性があるという点で、ユーザを混乱させる可能性がある。

TTS : True / False - テキスト - 音声がプロンプトを生成するのに使用されていたかどうか。

プロンプト完了時間：プロンプトが完成された / 中断された時間。

プロンプトwave ファイル：提供された実際のプロンプト。

【0084】

ユーザ入力：

40

mode : ユーザが DTMF / 音声のどちらを提供しているか。

type : ユーザが Command を提供しているかどうか、および、そうである場合に、どのタイプ（たとえば、 Help / Repeat / など）か、または、ユーザが Response を提供しているかどうか、および、そうである場合に、どのタイプ（たとえば、 Answer / Confirm / Deny ）か。

ダイアログモデルは、アプリケーションの文法の機能を、応答を提供する際のユーザの目的を示すユーザ応答の異なるタイプすなわち、 Answer 、 Accept 、 Deny などに分類する。これらのタイプは、ユーザが何を達成することを試みているとシステムが考えているかのインジケータとして直接にロギングすることができる。異なる応答タイプの例は、次の通りである。

50

A n s w e r - ユーザが、値を要求する質問に対する回答を提供した。

E x t r a A n s w e r - ユーザが、質問の焦点を越える回答を提供した。

A c c e p t - ユーザが、情報の1つを確認した。

D e n y - ユーザが、情報の1つに異議を唱えた。

H e l p C o m m a n d - ユーザが、ヘルプを求めた。

R e p e a t C o m m a n d - ユーザが、情報の繰り返しを要求した。

O t h e r C o m m a n d - ユーザが、ある他の形のコマンドを発行した（明示的にタイプを指定されないが、我々は、それが上のタイプのどれでもないことを知っている）。

S i l e n c e - ユーザが、何も言わなかった（これは、時々、ある形の「暗黙の受け入れ」として使用される）。 10

これらのタイプは、特定の文法に関連するので、ユーザが対応する文法に一致する何かを言う時に、必ず、これらを自動的にロギングすることができる。多くのシステムは、単一のダイアログターンに複数のタイプを含めることを許容し、たとえば、複数のアイテムを受け入れること、または単一のターンで1つのアイテムに回答し、別のアイテムを受け入れることを許容する。

S i l e n c e : 無音が検出される場合に、**M a x S i l e n c e s** に対する相対的なその個数またはカウント。

N o R e c o : 認識が発話について検出されない場合に、**M a x N o R e c o s** に対する相対的なその個数またはカウント。 20

E r r o r : エラーが発生した場合に、それがアプリケーションまたはプラットフォームによって送出されたかどうか。

【 0 0 8 5 】

r e s u l t :

R e c o g n i t i o n r e s u l t : システムによって返された認識結果。一般に、認識結果は、解釈された発話のセマンティックマークアップ言語（S M L）タグを含む。さらに、N - B e s t 代替解釈を提供することができ、適当な場合にオーディオ記録結果を提供することができる。

さらに、解釈ごとに、

S M L タグなしの発話テキスト（音声が提供される場合）またはキー押下げ（D T M F が提供される場合）。 30

c o n f i d e n c e : 解釈の信頼性レベル。

s e m a n t i c m a p p i n g s : S M L 結果の諸部分とセマンティックアイテムとの間のリンク。言い換えると、S M L 結果からのどの値がどのセマンティックアイテムに置かれるか。

g r a m m a r r u l e m a t c h e d : 文法のどのルールがユーザ入力と一致したか。

c o n f i d e n c e : 全体としての発話の信頼性。

b a r g e i n : ユーザによる会話への割込みのタイミングまたはN U L L（会話への割込みが存在しない場合）。 40

認識 **w a v e** ファイル : 実際に記録されたユーザ入力またはそれへのポインタ。

【 0 0 8 6 】

要約すると、ロギングされるユーザ対話データは、ダイアログを、関心を持たれているあるフィールド（たとえば、フォームフィールドまたはスロット値）を操作するタスクの階層構造またはシーケンシャル構造とみなすことを可能にし、タスク内の各ダイアログターンは、フォームフィールドに関するシステム目的（ダイアログ移動）（たとえば、値に関する質問、値の確認、値の繰り返しなど）と、音声リコグナイザがユーザ目的と考えているもの（たとえば、値の供給、値の拒否、ヘルプの要求など）との両方をロギングする。

【 0 0 8 7 】

10

20

40

50

実用的な利益が、この構造を用いて実現される。具体的に言うと、システム性能の分析は、成功または失敗のいずれかのタスク完了が、一般に明示的であり、したがって、トランザクション的成功レート報告が、非常に単純化され、タスクを完了するのに要するダイアログステップの性質が、よりよく理解される（各ステップの背後にある目的がオーサリング時に既知なので）という点で、改善される。

【0088】

この形のデータロギングの実装は、これがダイアログオーサリングツールに組み込まれる形に起因して、簡単である。この計測の高水準の性質は、さまざまなアプリケーションタイプに一般的であり、ロギングの実際の詳細は、概念的にとロギングプリミティブに関してとの両方でこれがオーサリングツールに統合されることによって、オーサリング時に容易にされる。したがって、アプリケーション作成者は、タスク／サブタスクモデルを使用してアプリケーションを構造化し、タスクから出るどの遷移が成功の完了を示すかを示すことを奨励され、システム／ユーザ目的ロギングがダイアログターンオーサリングモデルに組み込まれるので、システム／ユーザ目的ロギングを明示的に計測する必要がなくなる。

10

【0089】

上では、主題を、特定の実施形態を参照して説明したが、当業者は、添付の特許請求の範囲の趣旨および範囲から逸脱せずに、形態および詳細において変更を行えることを認めるであろう。

20

【図面の簡単な説明】

【0090】

【図1】コンピューティングデバイスオペレーティング環境の第1実施形態を示す平面図である。

【図2】図1のコンピューティングデバイスを示すブロック図である。

【図3】汎用コンピュータを示すブロック図である。

【図4】クライアント／サーバシステムのアーキテクチャを示すブロック図である。

【図5】クライアントサイドマークアップ内で認識および可聴コンピューティングを提供する手法を示すブロック図である。

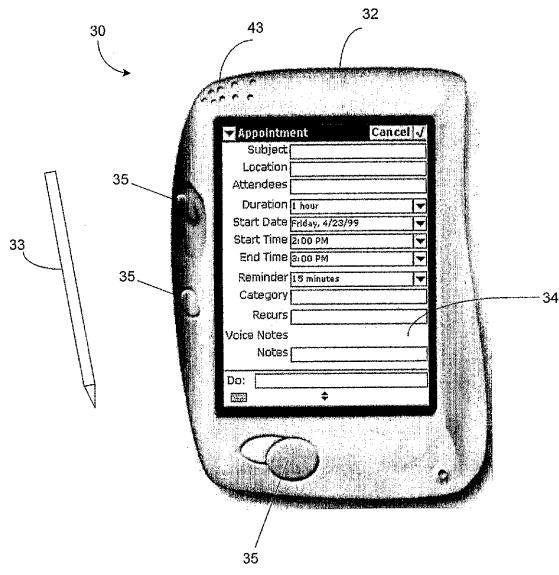
【図6】コンパニオンコントロールを示すブロック図である。

30

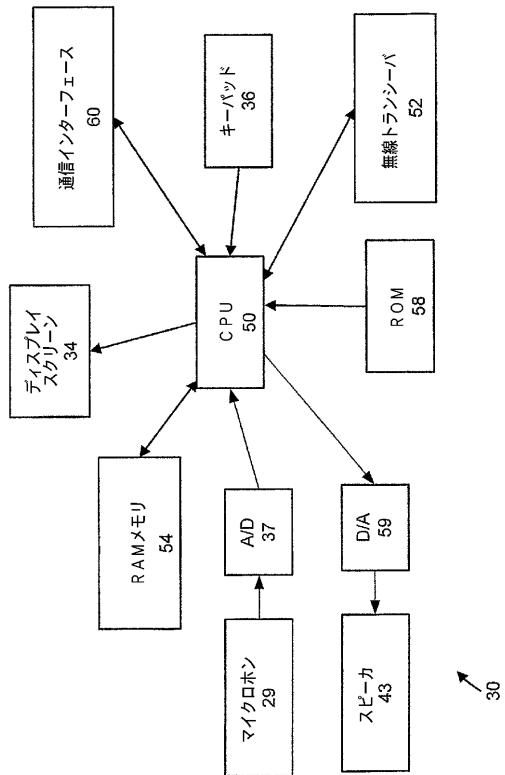
【図7】音声対応アプリケーションを作成する方法を示す流れ図である。

【図8】音声対応アプリケーションを実行する方法を示す流れ図である。

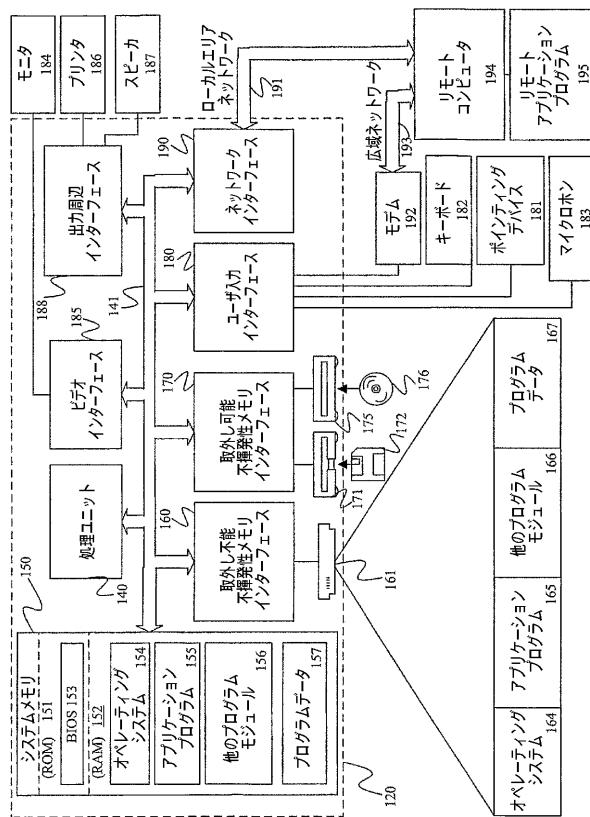
【図1】



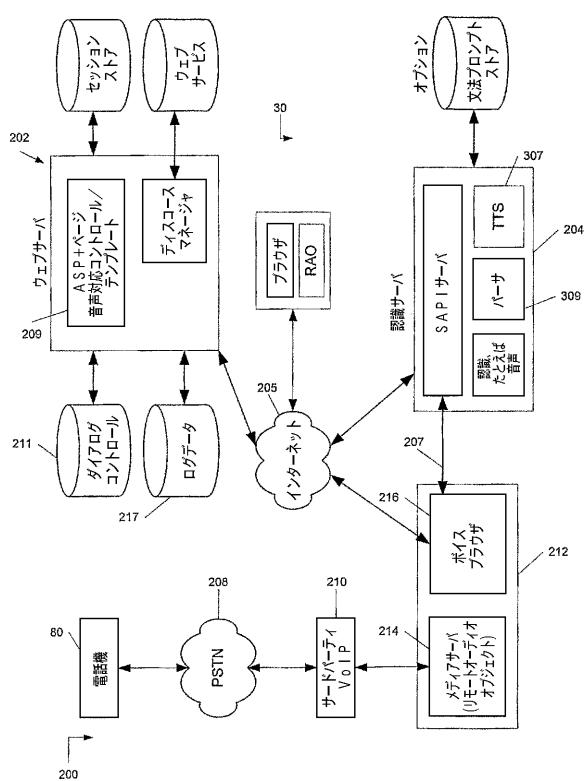
【図2】



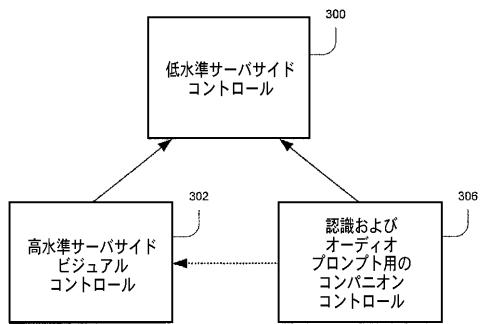
【図3】



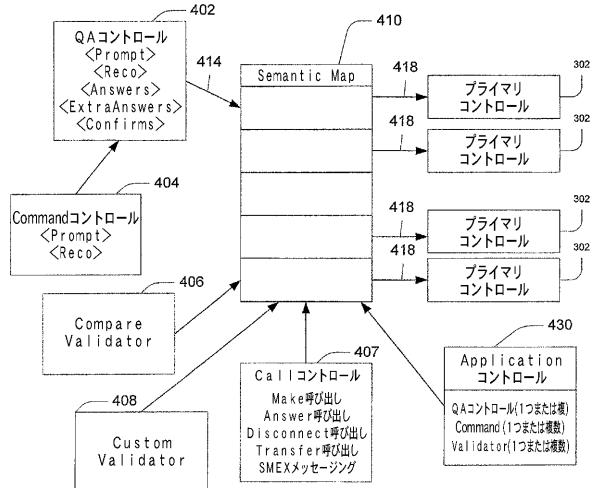
【図4】



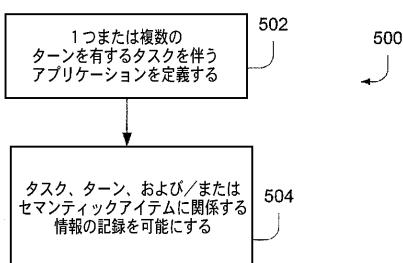
【図5】



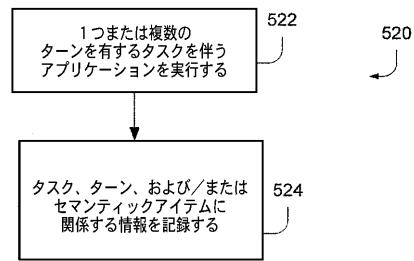
【図6】



【図7】



【図8】



フロントページの続き

(81) 指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HU,IE,IS,IT,LT,LU,LV,MC,NL,PL,PT,RO,SE,SI,SK,TR),OA(BF, BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BW,BY,BZ,CA,CH,CN,CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW

F ターム(参考) 5B042 GA21 LA23 MA08 MC36 MC37 NN42 NN45