

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4294876号
(P4294876)

(45) 発行日 平成21年7月15日 (2009. 7. 15)

(24) 登録日 平成21年4月17日 (2009. 4. 17)

(51) Int. Cl.

F I

G 0 6 F 17/30 (2006. 01)

G 0 6 Q 50/00 (2006. 01)

G 0 6 F 17/30 1 8 O Z

G 0 6 F 17/30 1 1 O G

G 0 6 F 17/30 3 4 O A

G 0 6 F 17/60 1 3 2

請求項の数 16 (全 90 頁)

(21) 出願番号 特願2000-604334 (P2000-604334)
 (86) (22) 出願日 平成12年3月3日 (2000. 3. 3)
 (65) 公表番号 特表2002-539532 (P2002-539532A)
 (43) 公表日 平成14年11月19日 (2002. 11. 19)
 (86) 国際出願番号 PCT/IB2000/000377
 (87) 国際公開番号 W02000/054178
 (87) 国際公開日 平成12年9月14日 (2000. 9. 14)
 審査請求日 平成17年5月24日 (2005. 5. 24)
 (31) 優先権主張番号 09/263, 252
 (32) 優先日 平成11年3月5日 (1999. 3. 5)
 (33) 優先権主張国 米国 (US)

早期審査対象出願

(73) 特許権者 501351313
 アクセンチュア・リミテッド・ライアビリ
 ティ・パートナーシップ
 ACCENTURE LLP
 アメリカ合衆国 カリフォルニア州943
 04 パロ・アルト, ペイジ・ミル・ロー
 ド, 1661
 (74) 代理人 110000028
 特許業務法人明成国際特許事務所
 (72) 発明者 ジャーシマン・アナトレ・ビタリー
 アメリカ合衆国 イリノイ州60657
 シカゴ, ウェスト・バリー・アベニュー,
 522

最終頁に続く

(54) 【発明の名称】 インターフェースサポートフレームワークを利用したモバイル通信のためのシステム、方法、および製品

(57) 【特許請求の範囲】

【請求項 1】

情報ネットワークと通信可能なシンククライアント コンピュータ を用いて パーソナライズ された情報を獲得するための方法であって、

前記シンククライアント コンピュータ が、ユーザから供給された情報を含み前記ユーザの複数のペルソナを表す個人パターンテンプレートを、前記シンククライアント コンピュータ 内のメモリに確立する動作を備え、

前記複数のペルソナは、前記複数のペルソナが適用される異なるコンテキストに対応した複数のペルソナセットを含む、

前記複数のペルソナセットは、ペルソナ情報を含む1つ以上のペルソナレコードと、各ペルソナレコードに対応する1つ以上のプロフィールレコードと、を含む階層的データレコードを設定することによって生成され、

各プロフィールレコードは、ユーザ名とユーザ住所とを含む詳細ユーザプロフィール情報を含む、

前記複数のペルソナセットは、各プロフィールレコードに含まれている前記詳細ユーザプロフィール情報と、各ペルソナレコードに含まれている前記ペルソナ情報とを、前記階層的データレコードが格納されるデータモデル内に集合させることによって生成されており、

前記方法は、さらに、

前記シンククライアント コンピュータ が、ユーザ入力に基づいて照会を作成する

10

20

動作と、

前記シンククライアントコンピュータが、情報ネットワークに情報の照会を送信する動作と、

前記シンククライアントコンピュータが、前記情報ネットワークからの照会に対する応答を受信する動作と、

前記シンククライアントコンピュータが、前記シンククライアントコンピュータ上のアプリケーションツールを利用して前記応答内の情報を処理してフィルタ済み応答を得る動作と、
を備え、

前記応答内の情報の処理は、

10

前記複数のペルソナの中から、ユーザの選択に基づいて1つのペルソナを選択する動作と、

選択されたペルソナのペルソナ情報、及び、詳細ユーザプロフィール情報に基づいて、前記アプリケーションツールによって前記応答内の情報をフィルタリングする動作と、
を含み、

前記方法は、さらに、

前記シンククライアントコンピュータが、前記選択されたペルソナに基づいて前記応答内でフィルタリングされた情報を前記ユーザに対して表示する動作と
を備える方法。

【請求項2】

20

請求項1記載の方法であって、

前記個人パターンテンプレートは、複数のワードパターンを含む、方法。

【請求項3】

請求項1記載の方法であって、

前記シンククライアントコンピュータは、前記情報ネットワークに存在するモバイルポータルサーバと通信可能である、方法。

【請求項4】

請求項1記載の方法であって、

前記シンククライアントコンピュータはインテリジェントエージェントプロセッサを含む、方法。

30

【請求項5】

請求項1記載の方法であって、

前記シンククライアントコンピュータは、カスタマデータを照会する手段を含む、方法。

【請求項6】

請求項1記載の方法であって、

前記シンククライアントコンピュータは、情報の暗号化と解読を行うためのセキュリティ手段を含む、方法。

【請求項7】

請求項1記載の方法であって、

前記シンククライアントコンピュータは、第三者のコンテンツを獲得するためのインターフェースを含む前記情報ネットワークと通信可能である、方法。

40

【請求項8】

請求項1記載の方法であって、

前記シンククライアントコンピュータはモバイルクライアントのサポートを含む前記情報ネットワークと通信可能である、方法。

【請求項9】

情報ネットワークと通信してパーソナライズされた情報を獲得するシンククライアントコンピュータであって、

ユーザから供給された情報を含み前記ユーザの複数のペルソナを表す個人パターンテンプレートを格納するメモリを備え、

50

前記複数のペルソナは、前記複数のペルソナが適用される異なるコンテキストに対応した複数のペルソナセットを含み、

前記複数のペルソナセットは、ペルソナ情報を含む1つ以上のペルソナレコードと、各ペルソナレコードに対応する1つ以上のプロフィールレコードと、を含む階層的データレコードを設定することによって生成され、

各プロフィールレコードは、ユーザ名とユーザ住所とを含む詳細ユーザプロフィール情報を含み、

前記複数のペルソナセットは、各プロフィールレコードに含まれている前記詳細ユーザプロフィール情報と、各ペルソナレコードに含まれている前記ペルソナ情報とを、前記階層的データレコードが格納されるデータモデル内に集合させることによって生成されており、

10

前記シンクライアントコンピュータは、さらに、

前記シンクライアントコンピュータ上で、ユーザ入力に部分的に基づいて照会を作成するロジックと、

前記情報ネットワークに情報の照会を送信するロジックと、

前記照会に対する応答を前記情報ネットワークから受信するロジックと、

前記シンクライアントコンピュータ上のアプリケーションツールを利用して前記応答内の情報を処理してフィルタ済み応答を得るロジックであって、前記複数のペルソナの中から、ユーザの選択に基づいて1つのペルソナを選択し、選択されたペルソナのペルソナ情報、及び、詳細ユーザプロフィール情報に基づいて、前記アプリケーションツールによっ

20

て前記応答内の情報をフィルタリングするロジックと、
前記選択されたペルソナに基づいて前記応答内でフィルタリングされた情報を前記ユーザに対して表示するロジックと
を備える装置。

【請求項10】

パーソナライズされた情報を取得するコンピュータプログラムを格納するコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、シンクライアントコンピュータのメモリにロード可能であり、

前記コンピュータプログラムは、

ユーザから供給された情報を含み前記ユーザの複数のペルソナを表す個人パターンテンプレートを確立する命令を備え、

30

前記複数のペルソナは、前記複数のペルソナが適用される異なるコンテキストに対応した複数のペルソナセットを含み、

前記複数のペルソナセットは、ペルソナ情報を含む1つ以上のペルソナレコードと、各ペルソナレコードに対応する1つ以上のプロフィールレコードと、を含む階層的データレコードを設定することによって生成され、

各プロフィールレコードは、ユーザ名とユーザ住所とを含む詳細ユーザプロフィール情報を含み、

前記複数のペルソナセットは、各プロフィールレコードに含まれている前記詳細ユーザプロフィール情報と、各ペルソナレコードに含まれている前記ペルソナ情報とを、前記階層的データレコードが格納されるデータモデル内に集合させることによって生成されており、

40

前記プログラムは、さらに、

前記シンクライアントコンピュータ上で、ユーザ入力に部分的に基づいて照会を作成する命令と、

情報ネットワークに情報の照会を送信する命令と、

前記情報ネットワークからの照会に対する応答を受信する命令と、

前記シンクライアントコンピュータ上のアプリケーションツールを利用して前記応答内の情報を処理してフィルタ済み応答を得る命令であって、前記複数のペルソナの中から、

50

ユーザの選択に基づいて1つのペルソナを選択し、選択されたペルソナのペルソナ情報、及び、詳細ユーザプロフィール情報に基づいて、前記アプリケーションツールによって前記応答内の情報をフィルタリングする命令と、

前記選択されたペルソナに基づいて前記応答内でフィルタリングされた情報を前記ユーザに対して表示する命令と

を備えるコンピュータ読み取り可能な媒体。

【請求項11】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記個人パターンテンプレートは、複数のワードパターンを含む、コンピュータ読み取り可能な媒体。

10

【請求項12】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、さらに、モバイルポータルサーバと通信するための命令を備える、コンピュータ読み取り可能な媒体。

【請求項13】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、さらに、前記情報ネットワークと通信してカスタマデータを照会するための命令を備える、コンピュータ読み取り可能な媒体。

【請求項14】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、さらに、情報の暗号化と解読を行うための命令を備える、コンピュータ読み取り可能な媒体。

20

【請求項15】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、さらに、前記情報ネットワークと通信して第三者のコンテンツを獲得するためのインターフェースを提供する命令を備える、コンピュータ読み取り可能な媒体。

【請求項16】

請求項10記載のコンピュータ読み取り可能な媒体であって、

前記コンピュータプログラムは、さらに、前記情報ネットワークと通信してモバイルクライアントのサポートを提供する命令を備える、コンピュータ読み取り可能な媒体。

30

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、エージェントベースのシステムに関し、特に、インターフェースサポートフレームワークを介し、インターネットにアクセスしてユーザが必要とする商品情報を獲得するモバイルコンピューティング環境に関する。

【0002】

【従来の技術及び発明が解決しようとする課題】

あらゆる環境におけるコンピュータ支援は、コンピュータ技術が社会に組み込まれる度合いが高まるにつれ、不可欠となりつつある。モバイルコンピューティング技術では、個人があらゆる時にあらゆる環境でコンピュータ関連情報にアクセスできるようにすることで、この問題に取り組んでいる。

40

【0003】

モバイルコンピュータ技術における第一の大きな進歩の1つは、携帯情報端末(PDA)である。PDAは、ユーザがコンピュータ関連情報にアクセスすることを可能にし、更に手のひらのサイズである。PDAを使用することで、ユーザは、身辺雑事を整理し、メモを取り、方程式を計算し、住所録に連絡先の電話番号を記録できる。加えて、PDAは通常、デスクトップコンピュータとの通信が可能であり、これは一般的には有線接続を介して行われる。この接続により、PDAは、デスクトップコンピュータとの間での情報のダ

50

ウンロード及び情報のアップロードが可能となる。その後の開発によって、PDAに無線機能が与えられた。無線機能により、PDAは、そのPDAに物理的に接続されていない他のコンピュータと対話することが可能になった。

【0004】

無線PDAは、ワールドワイドウェブに接続されたコンピュータと通信することが可能になり、これは間もなく、Webブラウジングが可能なPDAをもたらした。PDAのWebブラウジング機能を開発した第一の企業の1つはインターコムだった。インターコムのFalcon Mobile Serverにより、Web機能付きPDAは、ホストコンピュータと直接接続することができる。ソフトウェアをホストサーバにインストールするだけで、PDA端末はワールドワイドウェブを通じて情報にアクセスできるようになった。

10

【0005】

現在では、モバイルコンピューティングにおける更なる統合が望まれている。テキサス州アービングの企業であるノキアは、Nokia 9000無線ボイス電話を開発することで、この統合問題に部分的に取り組んだ。Nokia 9000には、小型キーボードと、マイクロブラウザベンダであるアンワイヤードプラネット社からの特別なWebブラウザと、小型VGAモニタが含まれる。ノキアは、エリクソン社、モトローラ社、及びアンワイヤードプラネット社と共同で、標準化されたブラウザ技術及びサーバフォーマットであるワイヤレスアプリケーションプロトコル(WAP)を確立した。WAPは、製造業者に、無線電話にデータ機能を搭載するための標準化された方法を提供し、これにより通信事業者は無線通信による管理の度合いを高めることができる。例えば、通信事業者が新しいデータサービスの実地試験を行いたい場合、この通信事業者はサーバ上でサービスを実施し、マイクロブラウザを通じて、これを電話に伝達し、満足できないサービスであることが分かった場合には、このサービスを調整できる。

20

【0006】

従来技術の図1Aは、Webポータルネットワークに基づく従来技術のモバイルコンピューティングソリューションの図である。従来技術において、ユーザ10は、ネットワークのそれぞれの参加者に個別に対処する必要がある。従来技術のモバイルコンピューティングソリューションにおいて、ユーザ10は、インターネットサービスプロバイダ(ISP)12を利用して、Webポータル14へのアクセスを得る。Webポータル14は、ユーザ10に直接情報を提供するサードパーティサービス16にアクセスする。しかしながら、インターネットサービスプロバイダ12と取引することに加え、ユーザ10は装置製造業者又は小売業者18から無線装置を購入する必要がある。ほとんどの場合では、ユーザ10は、ブラウザプロバイダ20からブラウザを購入する必要もある。一般に、ユーザは、無線通信費用を支払う必要があり、これによりユーザが電話会社22と取引する必要性が生じる。最後に、Webでの何らかの購入により、ユーザ10がクレジットカード会社24と取引する必要性が生じる。統一されたパッケージサービスが理想的なモバイルコンピューティングソリューションであることは明白である。更には、エージェントを使用する統一されたパッケージサービスが非常に望ましい。

30

【0007】

エージェントベースの技術は、フォアグラウンド及びバックグラウンドモードで様々なコンピュータベースのタスクを実行するためにユーザと対話するように設計されたアプリケーションでの使用に関して、重要性が増している。エージェントソフトウェアは、ルーチンであるタスク、冗長なタスク、及び時間のかかるタスクを実行するために、ユーザの立場に立って設定されたコンピュータプログラムを備える。個々のユーザによって有効となるように、エージェントは個々のユーザの目標、習慣、及び好みに合わせてパーソナライズする必要がある。そのため、ユーザからユーザ固有の知識を効率的及び効果的に取得し、これを利用してユーザに代わってタスクを実行するために、エージェントには大きな要件が存在する。

40

【0008】

エージェンシ又はエージェントのユーザという概念は定着している。エージェントとは、

50

通常は当事者と呼ばれる別の人間により、その当事者に代わって行動することを認められた人間である。これにより、当事者はエージェントに対して、当事者が実行したくないタスク又は実行できないタスクを実行する権限を与える。例えば、保険エージェントは当事者のために保険に関する要件を取り扱うことが可能であり、或いはタレントエージェントは演奏者に代わって行動し、コンサートの日程を手配できる。

【0009】

コンピュータの出現により、エージェントを利用する新しい領域が登場した。エキスパートシステムの分野での大きな進歩により、コンピュータプログラムは、コンピュータユーザに代わって行動し、ルーチンであるタスク、冗長なタスク、及びその他の時間のかかるタスクを実行することができる。こうしたコンピュータプログラムは、「ソフトウェアエージェント」と呼ばれる。

10

【0010】

更に、最近ではコンピュータ及び通信ネットワークが急増している。こうしたネットワークにより、ユーザは、基本的には、地理的な制限をまったく受けずに、膨大な量の情報及びサービスにアクセスすることができる。したがって、ソフトウェアエージェントは、ユーザに代わって多数のタスクを実行するための豊かな環境を有している。例えば、現在では、エージェントにより、飛行機の予約を行い、チケットを購入し、そのチケットをユーザに直接送ることが可能である。同様に、エージェントは、インターネットをスキャンし、最新のスポーツ情報又はニュースから応用物理の特定の卒業論文に至るまでの情報を取得することができる。現在のソリューションでは、ユーザの来るべきイベントについて、

20

的を絞って情報を取得することにエージェント技術を応用することはできない。

【0011】

【発明の概要】

Webベース情報検索及び表示システムを促進するシステムを開示する。インターネットプロトコル機能が付いた無線電話又は同様のハンドヘルド型無線装置が、インターネットへのポータブルポータルを提供するために、他の周辺機器と統合される。この無線装置は、ユーザに、ユーザが関心を持っている情報を入力することを促す。この情報により、(Webサーバ上で実行される) サービスルーチンにクエリが伝送される。このサービスルーチンは、その後、分散通信ネットワークを利用してWebにクエリ(照会)を行い、様々なWebサプライヤから価格、納品、及び入手可能性に関する情報を探す。この情報は、フォーマットされた後、インターフェースサポートフレームワークを介してハンドヘルドデバイスのスクリーン上に表示される。こうして、ユーザは、ハンドヘルドデバイスを使用して対話式に注文することが可能になる。

30

【0012】

【発明の実施の形態】

前記及びその他の目的、態様、及び利点は、図面を参照して、以下の本発明の好適な実施形態の詳細な説明からより良く理解される。

【0013】

本発明によるシステムの好適な実施形態は、好ましくは、IBM互換パーソナルコンピュータ、Appleマッキントッシュコンピュータ、又はUNIXベースワークステーションといったパーソナルコンピュータのコンテキストにおいて実施される。代表的なハードウェア環境は図1に表されており、ここでは、マイクロプロセッサ等の中央演算処理装置110と、システムバス112によって相互接続される他の多数のユニットとを有する好適な実施形態によるワークステーションの典型的なハードウェア構成を表している。図1に示すワークステーションは、ランダムアクセスメモリ(RAM)114と、読み取り専用メモリ(ROM)116と、ディスク記憶ユニット120等の周辺機器をバス112に接続するための入出力アダプタ118と、キーボード124、マウス126、スピーカ128、マイクロフォン132、及び又はタッチスクリーン(表示なし)等のその他のユーザインタフェース装置をバス112に接続するためのユーザインタフェースアダプタ122と、ワークステーションを通信ネットワーク(例えばデータ処理ネットワーク)に接続

40

50

するための通信アダプタ134と、バス112をディスプレイ装置138に接続するためのディスプレイアダプタ136とを含む。このワークステーションには通常、Microsoft Windows NT又はWindows/95オペレーティングシステム(OS)、IBM OS/2オペレーティングシステム、MAC OS、又はUNIXオペレーティングシステム等のオペレーティングシステムが常駐する。当業者には、こうした前記のものとは異なるプラットフォーム及びオペレーティングシステム上でも、本発明が実施可能であることが理解される。

【0014】

好適な実施形態は、Java(登録商標)、C、及びC++言語を使用し、オブジェクト指向プログラミング方法論を利用して書かれる。オブジェクト指向プログラミング(OOP)は、複雑なアプリケーションの開発に使用されることが増えている。OOPがソフトウェアの設計及び開発の主流になるにつれ、様々なソフトウェアソリューションでは、OOPの恩恵を利用するための適応が求められる。メッセージングインタフェースのためのOOPクラス及びオブジェクトのセットを提供できるように、こうしたOOPの原理を電子メッセージングシステムのメッセージングインタフェースに適応させる必要性が存在する。

10

【0015】

OOPはオブジェクトを使用するコンピュータソフトウェアを開発するプロセスであり、問題を分析するステップと、システムを設計するステップと、プログラムを構築するステップとを含む。オブジェクトとは、データと関連構造及び手順の集合とを両方とも包含するソフトウェアパッケージである。データと関連構造及び手順の集合とを両方とも包含するため、特定のタスクを実行するために他の追加構造、手順、又はデータを必要としない自給自足型コンポーネントとしてイメージできる。そのため、OOPでは、コンピュータプログラムを、ほぼ自立したコンポーネントの集合としてとらえ、これはオブジェクトと呼ばれ、それぞれが特定のタスクに関する責任を有する。こうしたデータ、構造、及び手順を1コンポーネント又はモジュールにパッケージングする概念は、カプセル化と呼ばれる。

20

【0016】

一般に、OOPコンポーネントは再使用可能なソフトウェアモジュールであり、オブジェクトモデルと一致するインタフェースを提示し、コンポーネント統合アーキテクチャを通じてランタイム時にアクセスされる。コンポーネント統合アーキテクチャは、異なる処理空間にあるソフトウェアモジュールが、互いの能力又は機能を利用することを可能にする一連のアーキテクチャメカニズムである。これは一般に、このアーキテクチャを構築する共通コンポーネントオブジェクトモデルを仮定することで実行される。

30

【0017】

この点において、オブジェクトとオブジェクトのクラスとを区別することには価値がある。オブジェクトは、オブジェクトのクラスの単一のインスタンスであり、単にクラスと呼ばれる場合が多い。オブジェクトのクラスは、青写真と考えることが可能であり、ここから多くのオブジェクトを形成することができる。

【0018】

OOPによって、プログラマは、別のオブジェクトの一部であるオブジェクトを作成することができる。例えば、ピストンエンジンを表すオブジェクトは、ピストンを表すオブジェクトと複合関係を有すると言える。実際には、ピストンエンジンは、ピストンと、バルブと、他の多くのコンポーネントとを備え、ピストンがピストンエンジンの要素であるという事実は、OOPにおいて、2つのオブジェクトによって、論理的及び意味的に表すことができる。

40

【0019】

OOPでは、他のオブジェクトに「ぶら下がる」オブジェクトを作成することもできる。2つのオブジェクトがあり、一方はピストンエンジンを表し、他方はピストンがセラミック製であるピストンエンジンを表す場合、2つのオブジェクトの間の関係は複合関係では

50

ない。セラミックピストンエンジンは、ピストンエンジンを構成していない。むしろ単に、ピストンエンジンよりも1つ多くの制限、つまりピストンがセラミック製であるという制限を有するピストンエンジンの一種である。この場合、セラミックピストンエンジンを表すオブジェクトは派生オブジェクトと呼ばれ、ピストンエンジンを表すオブジェクトのあらゆる態様を継承し、更なる制限又は詳細が加わっている。セラミックピストンエンジンを表すオブジェクトは、ピストンエンジンを表すオブジェクトに「ぶら下がって」いる。こうしたオブジェクトの間の関係は継承と呼ばれる。

【0020】

セラミックピストンエンジンを表すオブジェクト又はクラスが、ピストンエンジンを表すオブジェクトのあらゆる態様を継承する時、これはピストンエンジンクラスにおいて定められ標準的なピストンの熱特性を継承する。しかしながら、セラミックピストンエンジンオブジェクトは、金属製ピストンに関連するものとは通常異なる、セラミック固有熱特性をオーバーライドする。オリジナルをスキップして、セラミックピストンに関連する新しい関数を使用する。異なる種類のピストンエンジンは異なる特性を有するが、ピストンエンジンに関連する同じ基本関数を有する可能性がある（例えば、エンジン内のピストンの数、点火順序、潤滑等）。任意のピストンエンジンオブジェクトにおいて、こうしたそれぞれの関数にアクセスするために、プログラマは同じ関数を同じ名前呼び出すが、それぞれのタイプのピストンエンジンは、同じ名前の陰に異なる／オーバーライドされる関数の実施を有する可能性がある。異なる関数の実施を同じ名前の陰に隠すこうした能力は多形性と呼ばれ、これによりオブジェクト間の通信は大幅に簡素化される。

【0021】

複合関係、カプセル化、継承、及び多形性の概念により、オブジェクトは実世界のほとんどあらゆるものを表すことが可能である。実際には、私達が有する現実の論理的な認識は、オブジェクト指向ソフトウェアにおいてオブジェクトとなることが可能なものの種類を決定することに関する唯一の制限となる。いくつかの代表的なカテゴリは次の通りである、

- オブジェクトは、交通量シミュレーションの自動車、回路設計プログラムの電気コンポーネント、経済モデルの国、又は航空管制プログラムの飛行機といった物理的な物を表すことができる。

- オブジェクトは、ウィンドウ、メニュー、又はグラフィックオブジェクトといったコンピュータユーザ環境の要素を表すことができる。

- オブジェクトは、個人ファイル又は都市の緯度及び経度の表といった明細表を表すことができる。

- オブジェクトは、時間、角度、及び複素数、又は平面の点といったユーザ定義データタイプを表すことができる。

【0022】

ほぼすべての論理的に分離可能な内容を表すオブジェクトの非常に大きな能力を有するため、OOPによって、ソフトウェア開発者は、現実の一部の態様のモデルであるコンピュータプログラムを設計及び実施することが可能となり、その現実は物理的な実体、プロセス、システム、又は合成物であっても構わない。オブジェクトはいかなるもので表すことが可能であるため、ソフトウェア開発者は、将来の大きなソフトウェアプロジェクトにおいてコンポーネントとして使用することが可能なオブジェクトを作成できる。

【0023】

新しいソフトウェアプログラムの90%が、以前から存在する再使用可能なオブジェクトから作られた、証明済みで既存のコンポーネントから構成される場合、新しいソフトウェアプロジェクトの残りの10%のみを最初から書き、テストする必要がある。90%は、広範にテストされた再使用可能オブジェクトの評価に由来するため、エラーが発生する可能性のある領域はプログラムの10%である。その結果、OOPにより、ソフトウェア開発者は、他の既に構築されたオブジェクトからオブジェクトを構築することができる。

【0024】

このプロセスは、アセンブリ及びサブアセンブリから構築される複雑な機械と良く似ている。そのため、OOP技術では、オブジェクトとして開発者が利用可能な既存のコンポーネントからソフトウェアが構築される点において、ソフトウェアエンジニアはハードウェアエンジニアに近い存在となる。こうしたすべてが1つにまとまり、ソフトウェアの品質改善、及び開発速度の増加につながる。

【0025】

プログラミング言語は、カプセル化、継承、多形性、及び複合関係といったOOPの原理を完全にサポートし始めている。C++言語の登場により、多くの市販ソフトウェア開発者はOOPを採用した。C++は、高速な機械実行可能コードを提供するOOP言語である。更に、C++は、市販アプリケーションとシステムプログラミングプロジェクトとの両方に適している。今のところ、C++は多くのOOPプログラマの中で最も人気の高い選択肢であるが、Smalltalk、コモンリスプオブジェクトシステム(CLOS)、及びEiffelといったその他の多数のOOP言語が存在する。加えて、OOP機能は、Pascal等の更に伝統的で人気の高いプログラミング言語にも追加されている。

【0026】

オブジェクトクラスの利点は以下のようにまとめることができる。

- オブジェクトとそれに対応するクラスとは、複雑なプログラミング問題を多くの小さく単純な問題に分割する。
- カプセル化は、互いに通信可能な小さく独立したオブジェクトへのデータの編成を通じて、データ抽象化を強制する。カプセル化は、オブジェクト内のデータを予想外のダメージから保護するが、オブジェクトのメンバ関数及び構造を呼び出すことで、他のオブジェクトがそのデータと対話することを可能にする。
- サブクラシング及び継承により、システム内で利用可能な標準クラスから新しい種類のオブジェクトを派生させることを通じて、オブジェクトの拡張及び修正が可能となる。したがって、最初から始める必要なく、新しい機能が作成される。
- 多形性及び多重継承により、複数のプログラマは、多くの異なるクラスの特性を混合及び一致させ、依然として予測可能な形で関連するオブジェクトと協調することが可能な特別なオブジェクトを作成できる。
- クラス継承及びコンテナメント継承が、実世界のオブジェクトとそれらの間の関係とのモデリングに関する柔軟なメカニズムを提供する。
- 再使用可能クラスのライブラリは、多くの状況で有効であるが、いくつかの制限も有する。例えば、以下の通りである。
- 複雑性。複雑なシステムにおいて、関連するクラスのクラス継承は、数十又は数百のクラスにより、極めて紛らわしくなる可能性がある。
- 制御フロー。クラスライブラリを利用して書かれたプログラムは、依然として制御フローに責任を有する（つまり、特定のライブラリから作成されたすべてのオブジェクトの間の対話を制御する必要がある）。プログラマは、どの種類のオブジェクトに関して、どのタイミングで、どの関数を呼び出すかを判断する必要がある。
- 作業の重複。クラスライブラリによって、プログラマは多くのコードの小さな断片を使用及び再使用できるが、それぞれのプログラマはこうした断片を異なる方法で組み合わせる。二人のプログラマは、同じクラスライブラリのセットを使用して、正確に同じことを行う2つのプログラムを書くことができるが、これらの内部構造（つまり設計）はまったく異なる可能性があり、これはそれぞれのプログラマが途中で行った多数の小さな判断によって決まる。必然的に、同様のコードの断片は、わずかに異なる方法で同様の事柄を行うことになり、行うべき協調動作を行わない。

【0027】

クラスライブラリは非常に柔軟である。プログラムが複雑になるにつれ、更に多くのプログラマは基本的な問題に対する基本的なソリューションを何度も再発明することを強いられる。比較的新しいクラスライブラリ概念の拡張は、クラスライブラリのフレームワークを有することである。このフレームワークは更に複雑で、小規模なパターンと、特定の

10

20

30

40

50

アプリケーション領域での共通要件及び設計を実施する大きなメカニズムとを記録した共同クラスの大きな集合で構成される。これは最初、メニュー、ウィンドウ、ダイアログボックス、及びその他のパーソナルコンピュータ用の標準ユーザインタフェース要素を表示することに伴う雑用から、フリーアプリケーションのプログラマによって開発された。

【0028】

フレームワークは更に、自分の書いたコードと他人の書いたコードとの対話に関するプログラマの考え方の変化を表す。手続き型プログラミングの初期に、プログラマは、特定のタスクを実行するために、オペレーティングシステムが提供するライブラリを呼び出したが、基本的には、プログラムは最初から最後まで実行され、プログラマのみが制御フローに責任を有した。これは、給与小切手を印刷すること、数表を計算すること、又は1つの形のみで実行されるプログラムのその他の問題を解決することには適していた。

10

【0029】

グラフィカルユーザインタフェースの開発は、こうした手続き的プログラミングの仕組みを完全に覆し始めた。こうしたインタフェースにより、ユーザは、プログラムロジックではなく、プログラムを駆動し、一定の行動が実行されるべき時期を判断できるようになる。現在、ほとんどのパーソナルコンピュータソフトウェアでは、マウス、キーボード、及びその他の外部イベントのソースをモニタするイベントループによって、これを達成し、ユーザが実行した行動に従ってプログラマのコードの該当する部分を読み出す。プログラマは、もはやイベントが発生する順番を決定する必要はない。代わりに、プログラムは別個の断片に区分され、これは予測できない時期に予測できない順序で呼び出される。このようにして制御をユーザに委ねることで、開発者は、より使いやすいプログラムを作成する。にもかかわらず、開発者によって書かれたプログラムの個々の断片は、依然として特定のタスクを達成するためにオペレーティングシステムが提供するライブラリを読み出し、プログラマは依然として、イベントループに呼び出された後のそれぞれの断片内での制御フローを決定する必要がある。アプリケーションコードは依然としてシステムの「頂上に居座って」いる。

20

【0030】

イベントループプログラムであっても、プログラマは、すべてのアプリケーション毎に個別に書く必要がない多数のコードを書く必要がある。アプリケーションフレームワークの概念は、イベントループの概念を更に進める。基本メニュー、ウィンドウ、及びダイアログボックスを構築し、その後これらをすべて一緒に動作させるためのあらゆるナット及びボルトに対処する代わりに、アプリケーションフレームワークを使用するプログラマは最初に、アプリケーションコードと基本ユーザインタフェースを適所に配置する。その後、フレームワークの一般的な能力を、意図するアプリケーションの特定の能力と置き換えることで、そこから構築を行っていく。

30

【0031】

アプリケーションフレームワークは、プログラマが最初から書く必要のあるコードの総量を減少させる。しかしながら、フレームワークは実際のところ、ウィンドウの表示、コピー及びペーストのサポート、その他を行う一般的なアプリケーションであるため、プログラマはイベントループプログラムで許されるよりも大きな度合いの制御を放棄することもできる。フレームワークコードは、ほとんどすべてのイベントハンドリング及び制御フローを管理し、プログラマのコードは、フレームワークが必要とする時のみ（例えば、独自のデータ構造を作成又は操作するために）呼び出される。

40

【0032】

フレームワークプログラムを書くプログラマは、制御をユーザに委ねる（イベントループプログラムの場合も同じ）だけでなく、プログラム内の詳細な制御フローもフレームワークに委ねる。このアプローチにより、興味深い方法で共に動作する更に複雑なシステムの作成が可能となり、同様の問題のために何度も繰り返し作成されたカスタムコードを有する個別のプログラムとは対照的である。

【0033】

50

したがって、上で説明したように、フレームワークは基本的には、一定の問題領域に関する再使用可能な設計ソリューションを形作る協同クラスの集合である。通常は、（例えばメニュー及びウィンドウに関する）デフォルトの行動を提供するオブジェクトを含み、プログラマは、デフォルト行動の一部を継承させ、フレームワークが適切な時期にアプリケーションコードを呼び出すように、その他の行動をオーバーライドすることで、これを使用する。

【0034】

フレームワークとクラスライブラリとの間には主に3種類の相違点がある。

●行動対プロトコル。クラスライブラリは基本的には、プログラムにおいて個々の行動が求められる時に呼び出すことが可能な行動の集合である。一方、フレームワークは、行動

10

だけでなく、行動を統合可能な方法を管理するプロトコル又は規則セットを提供し、これにはプログラマが提供するはずのものに対するフレームワークが提供するものに関する規則が含まれる。

●呼び出し対オーバーライド。クラスライブラリでは、コードによりプログラマがオブジェクトのインスタンスを作成し、メンバ関数を呼び出す。フレームワークにより、同じようにオブジェクトのインスタンスを作成し、呼び出すことは可能だが（つまり、フレームワークをクラスライブラリとして扱う）、フレームワークの再使用可能設計の大きな利点を利用し、プログラマは通常、オーバーライドし、フレームワークによって呼び出されるコードを書く。フレームワークは、オブジェクト内での制御フローを管理する。プログラムを書くことには、様々な断片がどのように動作すべきかを指定することではなく、フレーム

20

ワークによって呼び出されるソフトウェアの様々な断片の中で責任を区分することが含まれる。

●実施対設計。クラスライブラリでは、プログラマは実施のみを再使用するが、フレームワークでは設計を再使用する。フレームワークは、関連プログラムのファミリ又はソフトウェアの断片が動作する方法を具現化する。これは一定の領域の様々な特定の問題に適合可能な一般的な設計ソリューションを表す。例えば、単一のフレームワークはユーザインタフェースが動作する方法を具現化可能であり、ただし、同じフレームワークで作成された2種類のユーザインタフェースは、まったく異なるインタフェースの問題を解決する可能性がある。

【0035】

30

したがって、様々な問題及びプログラミングタスクに対するソリューションのためのフレームワークの開発を通じて、ソフトウェアの設計及び開発作業における大きな削減が達成できる。本発明の好適な実施形態では、HyperText Markup Language (HTML) を利用して、クライアントとNewcoとの間で媒体を輸送する汎用の安全な通信プロトコルと共に、ドキュメントをインターネット上で実施する。HTTP又はその他のプロトコルは、過度の実験を行うことなく、容易にHTMLの代用とすることができる。こうした製品の情報は、T・バーナーズ＝リー、D・コノリの「RFC1866: Hypertext Markup Language-2.0」（1995年11月）、及びR・フィールディング、H・フリスティック、T・バーナーズ＝リー、J・ゲティス、及びJ・C・モグルの「Hypertext Transfer Protocol-HTTP/1.1: HTTP Working Group Internet Draft」（1996年5月2日）

40

【0036】

現在まで、Web開発ツールは、クライアントからサーバまでの範囲に渡り、既存のコンピューティングリソースと相互運用する動的Webアプリケーションを作成する能力にお

50

いて制限されている。最近まで、HTMLは、Webベースのソリューションの開発において使用される支配的な技術だった。しかしながら、HTMLは以下の領域において不適切であることが明らかにされている。

- 低い性能。
- 限られたユーザインタフェース能力。
- 静的Webページのみを作成可能。
- 既存のアプリケーション及びデータとの相互運用性の欠如。
- 基準化できないこと。

【0037】

サンマイクロシステムズのJava言語は、以下により、クライアント側の多くの問題を解決する。

- クライアント側の性能を改善すること。
- 動的なリアルタイムWebアプリケーションの作成を可能にすること。
- 幅広いユーザインタフェースコンポーネントを作成する能力を提供すること。

【0038】

Java(登録商標)によって、開発者は堅牢なユーザインタフェース(UI)コンポーネントを作成することができる。カスタム「ウィジェット」(例えば、リアルタイム株価表示機、アニメーションアイコン等)を作成可能であり、クライアント側の性能は改善される。HTMLとは違い、Java(登録商標)はクライアント側の妥当性確認の概念をサポートし、性能の改善のためにクライアントに適切な処理をオフロードする。動的なリアルタイムWebページが作成可能である。前記カスタムUIコンポーネントを使用することで、動的Webページも作成可能である。

【0039】

サンのJava言語は、「インターネットをプログラムする」ための業界で認められた言語として出現した。サンではJava(登録商標)を、「シンプルで、オブジェクト指向で、分散型で、インタープリタ型で、堅牢で、安全で、アーキテクチャニュートラルで、ポータブルで、高性能で、マルチスレッドで、動的で、専門語準拠である汎用プログラミング言語」と定義している。Java(登録商標)は、プラットフォーム独立Javaアプレットの形態において、インターネットのためのプログラミングをサポートする。Javaアプレットは、サンのJavaアプリケーションプログラミングインタフェース(API)に準拠した小さな特化したアプリケーションで、これにより開発者は、Webドキュメントに「インタラクティブコンテンツ」を追加することができる(例えば、単純なアニメーション、ページ装飾、基本的なゲーム)。アプレットは、サーバからクライアントへコードをコピーすることで、Java互換ブラウザ(例えば、Netscape Navigator)内で実行される。言語の観点からは、Java(登録商標)の中核機能セットはC++に基づいている。サンのJava(登録商標)に関する文献では、Java(登録商標)は基本的に「C++であり、更に動的なメソッドソリューションのためにObjective Cの拡張子が付いている」と記載されている。

【0040】

Java(登録商標)と同様の機能を提供する別の技術は、マイクロソフトとActiveXテクノロジーによって提供され、開発者及びWebデザイナーに、インターネット及びパーソナルコンピュータ用の動的コンテンツを構築するのに必要な手段を与える。ActiveXは、アニメーションと、3Dバーチャルリアリティと、ビデオと、その他のマルチメディアコンテンツとを開発するツールを含む。このツールはインターネット規格を使用し、複数のプラットフォームで動作し、100を超える企業によってサポートされている。このグループのビルディングブロックは、ActiveXコントロールと呼ばれる小さな高速のコンポーネントで、これにより開発者はソフトウェアの一部をHypertext Markup Language(HTML)のページに埋め込むことができる。ActiveXコントロールは、様々なプログラミング言語と共に動作し、こうした言語にはMicrosoft Visual C++、Borland Delphi、Microsoft Visual Basicプログラミングシステムが含まれ、将来的には

10

20

30

40

50

、マイクロソフトのJava用開発ツール、コードネーム「Jakarta」も含まれる。ActiveXテクノロジは更に、ActiveXサーバフレームワークを含み、これにより開発者はサーバアプリケーションを作成することが可能になる。当業者は、本発明を実施するために、過度の実験を行うことなく、ActiveXをJava(登録商標)の代用とすることができると容易に認識する。

【0041】

好適な実施形態によれば、個人が様々なソースからミーティングについての関連情報を検索するのを支援することで、来るべきミーティングのための個人の準備に責任を有するエージェントとして、バックグラウンドファインダ(BF)が実施される。BFは対象となるミーティングを示す文字の形態で入力テキストを受領する。この入力テキストは、ミーティングの時間を含むカレンダープログラムによって、好適な実施形態に従って生成される。ミーティングの時間が近づくと、カレンダープログラムは対象イベントのテキストを取得するためにクエリを受け、その情報はエージェントへの入力として利用される。その後、エージェントは、入力ミーティングテキストをパースし、タイトル、ボディ、参加者、場所、時間等の様々なコンポーネントを抽出する。このシステムは、ミーティングテキストの特定のミーティングフィールドを特定するためにパターンマッチングも実行する。この情報は、Web上の様々な情報源にクエリを行い、現在のミーティングについての関連する記事を取得してカレンダーリングシステムに返送するために利用される。例えば、個人がネットスケープとマイクロソフトとの紛争に関して話し合うために、これら2社とのミーティングに出席する場合、この個人はカレンダーリングシステムから、こうした初期情報を取得することになる。このシステムはその後テキストをパースし、ミーティングする企業が「ネットスケープ」及び「マイクロソフト」であり、トピックが「紛争」であることを認識する。その後、このシステムは、トピックについての関連情報をWebでクエリする。したがって、本発明の目的によれば、このシステムはカレンダーリングシステムを更新し、最終的には、対象となるミーティングに関するユーザの準備をするために収集できる最高の情報をユーザに与える。好適な実施形態によれば、この情報はカレンダーシステムに組み込まれたリンクからの選択により取得されたファイルに格納される。

【0042】

プログラム構成：

好適な実施形態によるコンピュータプログラムは、5つの個別のモジュール、つまりBF.Main、BF.Parse、Background Finder.Error、BF.PatternMatching、及びBF.Searchによって構成される。更に、デバッグの目的でのみ使用されるユーザインタフェースを提供するfrmMainも存在する。好適な実施形態による実行可能なプログラムは、このユーザインタフェースによって決して実行されず、マイクロソフトのWinsoc kコントロールを通じてのみカレンダーリングシステムに戻されるべきである。システムの好適な実施形態は、2種類のモードで実行され、これはカレンダーリングシステムによってシステムに送られるコマンドラインの下で指定できる。このシステムは、シンプルモードで動作している時、キーワードクエリを実行し、外部のサーチエンジンに発信する。コンプレックスモードで実行される時、このシステムは、サーチエンジンに送信するクエリを形成する前にパターンマッチングを実行する。

【0043】

データ構造：

好適な実施形態によるシステムは、3つのユーザ定義構造を利用する。

1. TMeetingRecord
2. TPatternElement
3. TPatternRecord

ユーザ定義構造TMeetingRecordは、単一のミーティングに関するあらゆる関連情報を格納するために使用される。この情報には、ユーザID、ミーティングの当初の説明、ミーティングのタイトル及びボディからのキーワードの抽出リスト等が含まれる。重要な注意点として、好適な実施形態によるシステムのインスタンスにつき、1つのみのミーティング

レコードが作成される。これは、来るべきミーティングに関するサービスをおこなうためにシステムが生じる度に、1つのみのミーティングに関する情報を検索するタスクが割り当てられるためである。そのため、現在のミーティングに対応して作成されたミーティングレコードが調査される。ParseMeetingTextは、このミーティングレコードを占有させ、その後これはミーティングに関する情報を他の機能に提供するためにパースされる。GoPatternMatchが、特定のミーティングフィールドに任意の値を結びつけることが可能である場合、ミーティングレコード内の対応するエントリも更新される。括弧内で説明されるそれぞれのフィールドが付いたtMeetingRecordの構造を、好適な実施形態に従って、以下に提示する。

【0044】

10

A.1.1.1.1.1 Public Type tMeetingRecord

sUserID As String (Muninによって与えられるユーザID)

sTitleOrig As String (Muninに送り返すために側に置く必要があるオリジナルのノンストップリストッドタイトル)

sTitleKW As String (キーワードのみを有するストップリストッドタイトル)

sBodyKW As String (キーワードのみを有するストップリストッドボディ) sCompany()
) As String (パターンマッチングを通じてタイトル又はボディにおいて特定される会社)

sTopic() As String (パターンマッチングを通じてタイトル又はボディにおいて特定されるトピック)

20

sPeople() As String (パターンマッチングを通じてタイトル又はボディにおいて特定される人間)

sWhen() As String (パターンマッチングを通じてタイトル又はボディにおいて特定される時間)

sWhere() As String (パターンマッチングを通じてタイトル又はボディにおいて特定される場所)

sLocation As String (Muninによって渡される場所)

sTime As String (Muninによって渡される時間)

sParticipants() As String (Muninによって渡される従事する全参加者)

sMeetingText As String (ユーザIDのないオリジナルのミーティングテキスト)

30

End Type

【0045】

パターンマッチングにおいて利用されるそれぞれの個別パターンを保持するために作成される他の2つの構造が存在する。レコードtAPatternRecordは、パターンのすべてのコンポーネント／要素を含む配列である。タイプtAPatternElementは、パターン内の要素を表すストリングの配列である。それぞれの要素には多くの「代替物」が存在する可能性があるため、すべての代替物が何かを追跡するストリングの配列が必要となる。tAPatternElement及びtAPatternRecordの構造を、好適な実施形態に従って、以下に提示する。

【0046】

40

Public Type tAPatternElement

elementArray() As String

End Type

Public Type tAPatternRecord

patternArray() tAPatternElement

End Type

【0047】

共通ユーザ定義定数：

プログラムのそれぞれの宣言セクションでは多くの定数が定義されており、これは好適な

50

実施形態によるシステムを維持するプロセスの一部として定期的な更新が必要な場合がある。この定数は、コードを維持するための更新として発生するシステムの動的な設定を可能にするためにアクセス可能である。

【0048】

以下の表に含まれるのは、時折修正される可能性が最も高いと私が考えるそれぞれのモジュールからの定数のリストである。しかしながら、以下のリストに含まれないが、コード内で使用される他のコードも存在する。これは、こうしたリストに含まれない定数が、決して変更されないことを意味するものではない。変更される頻度が少ないことを意味するものである。

【0049】

メインモジュール (BF.Main) について：

【0050】

【表1】

定数	プリセット値	使用法
MSGTOMUNIN_TYPE	6	BF と Munin との間のメッセージを特定するのに使用されるメッセージ数を定義する。
IP_ADDRESS_MUNIN	「 10.2.100.4 8」	Munin 及び BF がUDPを通じてデータを転送できるように、これらが実行されるマシンのIPアドレスを定義する。
PORT_MUNIN	7777	運用しているリモートポートを定義する。
TIMEOUT_AV	60	inetコントロールにおいてタイムアウトを設定するための定数を定義する。
TIMEOUT_NP	60	inetコントロールにおいてタイムアウトを設定するための定数を定義する。
CMD_SEPARATOR	「\」	Munin のコマンドのどの部分が入力ミーティングテキストの始まりを表すかを伝える区切り文字を定義する。
OUTPARAM_SEPARATOR	「::」	出力の異なる部分を分離する区切り文字を定義する。この分離記号は、msgタイプと、ユーザIDと、ミーティングタイトルと、検索した実際の記事の始まりとを区別するためのものである。

【0051】

サーチモジュール (BF.Search) について：

【0052】

【表2】

定数	現在値	使用法
PAST_NDAYS	5	Alta Vistaの項目を振り返りたい日数を定義する。実際にアルタビスタにおいてニュース検索を実行してはいないため、ここでは実際には関係ない。
CONNECTOR_AV_URL	「 +AND +」	キーワードを結合する方法を定義する。すべてのキーワードをストリングに入れたいため、ここではANDを使用する。ORその他を使用したい場合は、結合子のみを変更する。
CONNECTOR_NP_URL	「 +AND +」	キーワードを結合する方法を定義する。すべてのキーワードをストリングに入れたいため、ここではANDを使用する。ORその他を使用したい場合は、結合子のみを変更する。
NUM_NP_STORIES	3	News Pageから Munin に戻される記事の数を定義する。
NUM_AV_STORIES	3	Alta Vistaから Munin に戻される記事の数を定義する。

10

20

【 0 0 5 3 】

パースモジュール (BF.Parse) について：

【 0 0 5 4 】

【表 3】

定数	現在値	使用法
PORTION_SEPARATOR	「::」	Munin によって送られるミーティングテキストの異なる部分の区切り文字を定義する。例えば、「09::Meet with Chad::about life :: Chad Denise:::」において、「::」はミーティングテキストの異なる部分の区切り文字である。
PARTICIPANT_SEPARATOR	「 」	オリジナルのミーティングテキストの参加者リスト部分におけるそれぞれの参加者の区切り文字を定義する。上の例を参照せよ。

30

40

【 0 0 5 5 】

パターンマッチングモジュール (BFPatternMatch) について：このモジュールには、頻繁に更新する必要のある定数は存在しない。

【 0 0 5 6 】

一般的なプロセスフロー：

プロセスフローとそれぞれの間の関数の調整とは、図 2 乃至 6 に例示した 5 つのフローチ

50

ャートによって最も上手く表すことができる。図2は、好適な実施形態による全体的なプロセスフローを表している。処理はチャートの最上部にある関数ブロック200で開始され、これはプログラムがスタートした時に開始される。アプリケーションがスタートすると、関数ブロック210に示すように、コマンドラインがパースされ、適切なミーティングテキストが移動され、好適な実施形態によるバックグラウンドファインド動作のターゲットが開始される。関数ブロック220に示すように、ターゲットが決定された後、グローバルストップリストが生成される。その後、関数ブロック230に示すように、マッチング動作に利用されるすべてのパターンが生成される。その後、チャートをトレースすることにより、関数ブロック200がGoBF240を呼び出し、これは特定のターゲットのサーチエンジンのために正しい検索クエリ情報をラップすることに関するロジック処理に責任を有する。例えば、関数ブロック240は関数ブロック250に続き、その後、関数ブロック260に示すようにGoPatternMatchを呼び出す。GoPatternMatchのプロセスフローを見るためには、「BFのパターンマッチングユニットに関するプロセスフロー」というタイトルの図にスワップする。

10

【0057】

注意すべき重要な事柄の1つは、チャートの同じレベルで表された関数は、共通する親関数によって、左から右（上から下）に連続して呼び出されることである。例えば、Main200は、ProcessCommandLine210を呼び出し、その後CreateStopList220を呼び出し、その後CreatePattern230を呼び出し、その後GoBackgroundFinder240を呼び出す。図3乃至6はそれぞれ、プログラム全体、パーシングユニット、パターンマッチングユニット、及びサーチユニットに関するロジックを詳細に表している。図6は、BackgroundFinderを通じた重要な情報のデータフローを決定するロジックを詳細に表しており、こうした情報の作成及び処理に責任を有する関数を示している。

20

【0058】

シンプルクエリモード下での詳細なサーチアーキテクチャ：

Alta Vistaの検索：

【0059】

（図2の関数ブロック270）

Alta Vistaサーチエンジンは、図2の関数ブロック270に示すように、識別情報を利用して、現在のミーティングに関連するトピックに関する全般情報を戻す。好適な実施形態によるシステムは、オリジナルミーティングテキストのタイトル部分からすべてのキーワードを取り出し、Alta Vistaに対して送る高度なクエリを構築する。このキーワードは、クエリにおいて論理的に統合される。この結果も、同じキーワードのセットに基づいてランク付けされる。検索したい項目に関して、データ制限又は発行者の基準を促進できることを当業者は容易に理解する。上位ランクの記事のセットは、好適な実施形態によるカレンダーリングシステムに戻される。

30

【0060】

News Page：

（図2の関数ブロック275）

News Page検索システムは、ターゲットミーティングに関する最新のニューストピックを与えることに責任を有する。このシステムは、オリジナルミーティングテキストのタイトル部分からすべてのキーワードを取り出し、News Pageサーチエンジンに対して送るクエリを構築する。このキーワードは、クエリにおいて論理的に統合される。最近発行された項目のみが検索される。News Page検索システムは、ユーザの好みに応じてユーザが設定可能なデータ制限基準を提供する。上位ランクの記事はカレンダーリングシステムに戻される。

40

【0061】

図3は、好適な実施形態のユーザプロフィールデータモデルである。処理は関数ブロック300で開始され、これはメインモジュールからプログラムを呼び出すことに責任を有する。その後、関数ブロック310において、ラップ関数が呼び出され、関数ブロック32

50

0でのキーワード抽出処理が準備される。キーワードが抽出された後、処理フローは関数ブロック330に続き、区切り文字が正確に配置されているかどうか判断される。その後、関数ブロック340において、特定のストリング内のワード数が計算され、関数ブロック350において、特定のフィールドの区切り文字と、ミーティングテキストからの特定のフィールドとが検索される。その後、関数ブロック380において、ストリングの区切り文字が再度チェックされ、適切に配置されていることが確認される。最後に、関数ブロック360において、メッセージのタイトル及びボディからのそれぞれのワードの抽出が一度に1ワードずつ実行され、入力語句において次に最も近いワード区切り文字を見つける関数ブロック362と、ワードから不要なデータを取り去る関数ブロック364と、ワードがストップリストにあるかどうかを判断し、ワードがストップリストにある場合はエラーを戻す関数ブロック366とのロジックが利用される。

10

【0062】

好適な実施形態によるパターンマッチング：

シンプルサーチ方法に関連する制限には以下が含まれる。

【0063】

1. ミーティングテキストからキーワードのセットを抽出するために、不要なワードのストップリストに頼ることから、ストップリストがどのくらい包括的かによって制限される。ミーティングテキストのどの部分を捨てるべきかを考えるよりも、ミーティングテキストのどの部分が必要かに注目するべきである。

【0064】

2. 好適な実施形態によるシンプルサーチ方法では、ミーティングタイトルからのキーワードのみを使用して、Alta Vista及びNews Pageに送るクエリを形成する。これは、クエリの代替情報源、つまりミーティング通知のボディを無視している。ミーティングボディからのキーワードをクエリの形成に含めることは不可能であり、これは意味のある結果を得られないほど長く複雑なクエリが生じることが多いためである。

20

【0065】

3. それぞれのキーワードが何を意味するかを伝える方法は存在しない。例えば、2つのキーワードとして「アンディ」と「グローブ」とが抽出される可能性がある。しかしながら、シンプルサーチでは、「アンディ・グローブ」が実際は人名であることを知る方法はない。「アンディ・グローブ」が人名であることを何らかの方法で推測できる可能性を想像せよ。彼が雇用されている場所及び現在住んでいる場所といった情報。

30

【0066】

4. まとめると、不必要なワードをパースして除外するためにストップリストのみに頼っていることから、「情報過多」に悩まされている。

【0067】

好適な実施形態に従って、パターンマッチングはこうした制限を克服する：

ここでは、好適な実施形態に従って、上に対応するそれぞれの問題にパターンマッチングがどのように対処できるかを示す。

【0068】

1. パターンマッチングを行うことで、ミーティングテキストの必要な部分のみを一致させ、この部分を抽出する。

40

【0069】

2. ミーティングボディに関してパターンマッチングを実行し、ミーティングボディから必要な部分のみを抽出すること。これにより、ミーティングボディは完全に無駄なものにはならない。

【0070】

3. パターンマッチングは、指定したテンプレートのセットに基づき、人名、社名、及びその他の項目をミーティングテキストから特定することを可能にする。

【0071】

4. まとめると、パターンマッチングにより、情報過多に悩まされることがなくなる。当

50

然ながら、大きな問題は、パターンマッチングがどの程度上手く動作するかである。人工知能処理のみに頼る場合、100%のヒット率にはならない。提示された全社名の約20%を特定することが可能である。

【0072】

パターン：

好適な実施形態のコンテキストにおけるパターンは、ミーティングテキスト内で探している語句の構造を指定するテンプレートである。パターンは誰かのミーティングテキストの現れる高い可能性を有する語句のテンプレートであるため、好適な実施形態によってサポートされるパターンは選択される。例えば、カレンダーにミーティングを入力する時、多くの人は「Meet with Bob Dutton from Stanford University next Tuesday (来週の火曜日にスタンフォード大学のボブ・ダットンと会う)」といった事柄を書き込むことになる。この場合、共通パターンは、ワード「with」の後に人名（この例ではBob Dutton）が続き、その後にワード「from」が続き、最後は団体名（この例ではStanford University）で終わるといったものになる。

10

【0073】

パターンマッチングの用語：

パターンマッチングに関連する共通用語を以下に提示する。

◆パターン：パターンとは、ミーティングテキストを結びつけたい語句の構造を指定するテンプレートである。これはサブユニットを含む。

【0074】

◆要素：パターンは多くのサブユニットを含むことができる。こうしたサブユニットは要素と呼ばれる。例えば、「with \$PEOPLE\$ from \$COMPANY\$」というパターンにおいては、「with」、「\$PEOPLE\$」、「from」、「\$COMPANY\$」がすべての要素である。

20

【0075】

◆プレースホルダ：プレースホルダは、値を結びつける必要がある特別な種類の要素である。上の例を使うと、「\$PEOPLE\$」はプレースホルダである。

【0076】

◆インジケータ：インジケータは、ミーティングテキスト内で見つける必要があるが、値を結びつける必要がない別の種類の要素である。特定のパターンにおいて、1つ以上のインジケータを探す場合が多い。そのためインジケータは「アトミック」タイプではない。

30

【0077】

◆置換文字：置換文字は、互いにすべて同義語であるインジケータのセットである。入力において、これらのいずれかを見つかるのは良いことである。

【0078】

それぞれのミーティングに関して特定される5つのフィールドが存在する。

◆会社 (\$COMPANY\$)

◆人物 (\$PEOPLE\$)

◆場所 (\$LOCATION\$)

◆時間 (\$TIME\$)

◆トピック (\$TOPIC_UPPER\$) 又は (\$TOPIC_All\$)

40

括弧内は、コード内で対応するミーティングフィールドを表すものとして私が使用したプレースホルダである。

【0079】

それぞれのプレースホルダは以下の意味を有する。

◆\$COMPANY\$：大文字で始まるワードのストリングを結びつける（例えば、Meet with Joe Carter of <Andersen Consulting>）。

【0080】

◆\$PEOPLE\$：「,」、「and」、又は「&」によって結合される可能性がある大文字で始まる2つのワードのストリングの連続を結びつける（例えば、Meet with <Joe Carter> of Andersen Consulting、Meet with <Joe Carter and Luke Hughes> of Andersen Cons

50

ulting)。

【0081】

◆\$LOCATION\$：大文字で始まるワードのストリングを結びつける（例えば、Meet Susan at<Palo Alto Square>）。

【0082】

◆\$TIME\$：フォーマット#：##を含むストリングを結びつける（例えば、Dinner at <6:30pm>）。

【0083】

◆\$TOPIC_UPPER\$：トピックに関する大文字で始まるワードのストリングを結びつける（例えば<Stanford Engineering Recruiting> Meeting to talk about new hires）。

10

【0084】

◆\$TOPIC_All\$：大文字で始まるかどうかに関係なく、ワードのストリングを結びつける（例えば、Meet to talk about <ubiquitous computing>）。

【0085】

以下はBFによってサポートされるすべてのパターンを示す表である。それぞれのパターンはパターングループに属する。パターングループ内のすべてのパターンは、同様のフォーマットを共有し、どのインジケータが置換文字として使用されるかという点においてのみ、それぞれが異なる。注意点として、灰色のパターンはコードにおいてもコメントされている。BFは、こうしたパターンをサポートする能力を有するが、私達はこれらのパターンをマッチングさせることがこの時点で不可欠ではないと判断した。

20

【0086】

【表4】

パ タ ー ン グ ル ー プ	パ タ ー ン #	パ タ ー ン	例
1	a	\$PEOPLE\$ of \$COMPANY\$	Paul Martiz of Microsoft
	b	\$PEOPLE\$ from \$COMPANY\$	Bill Gates,Paul Allen and Paul Martiz from Microsoft
2	a	\$TOPIC_UPPER\$ Meeting	Push Technology Meeting
	b	\$TOPIC_UPPER\$ Mtg	Push Technology Mtg
	c	\$TOPIC_UPPER\$ demo	Push Technology demo
	d	\$TOPIC_UPPER\$ interview	Push Technology interview
	e	\$TOPIC_UPPER\$ presentati on	Push Technology presentation
	f	\$TOPIC_UPPER\$ visit	Push Technology visit
	g	\$TOPIC_UPPER\$ briefing	Push Technology briefing
	h	\$TOPIC_UPPER\$ discussion	Push Technology discussion
	i	\$TOPIC_UPPER\$ workshop	Push Technology workshop
	j	\$TOPIC_UPPER\$ prep	Push Technology prep
	k	\$TOPIC_UPPER\$ review	Push Technology review
	l	\$TOPIC_UPPER\$ lunch	Push Technology lunch
	m	\$TOPIC_UPPER\$ project	Push Technology project
	n	\$TOPIC_UPPER\$ projects	Push Technology projects
3	a	\$COMPANY\$ corporation	Intel Corporation
	b	\$COMPANY\$ corp.	IBM Corp.
	c	\$COMPANY\$ systems	Cisco Systems
	d	\$COMPANY\$ limited	IBM limited
	e	\$COMPANY\$ ltd	IBM ltd
4	a	about \$TOPIC_All\$	About intelligent agents technology

10

20

30

40

パターン グループ	パターン#	パターン	例
4	b	discuss \$TOPIC_All\$	discuss intelligent agents technology
	c	show \$TOPIC_All\$	show the client our intelligent agents technology
	d	re:\$TOPIC_All\$	re:intelligent agents technology
	e	review \$TOPIC_All\$	review intelligent agents technology
	f	agenda	The agenda is as follows: ——clean up ——clean up ——clean up
	g	agenda:\$TOPIC_All\$	agenda: ——demo client intelligent agents technology ——demo ecommerce
5	a	w/\$PEOPLE\$ of \$COMPANY\$	Meet w/Joe Carter of Andersen Consulting
	b	w/\$PEOPLE\$ from \$COMPANY\$	Meet w/Joe Carter from Andersen Consulting
6	a	w/\$COMPANY\$ per \$PEOPLE\$	Talk w/Intel per Jason Foster
7	a	At \$TIME\$	at 3:00pm
	b	Around \$TIME\$	Around 3:00pm
8	a	At \$LOCATION\$	At LuLu's restaurant
	b	In \$LOCATION\$	In Santa Clara
9	a	per \$PEOPLE\$	per Susan Butler
10	a	call w/\$PEOPLE\$	Conf call w/John Smith
	B	call with \$PEOPLE\$	Conf call with John Smith
11	A		prep for London Meeting
	B	preparation for \$TOPIC_All\$	preparation for London Meeting

【 0 0 8 7 】

図4は、好適な実施形態によるパターンマッチングの詳細なフローチャートである。処理は関数ブロック400で開始され、ここでメインプログラムがパターンマッチングアプリケーションを呼び出し、制御を関数ブロック410に渡し、パターンマッチング処理を開始する。次に、関数ブロック420において、ラップ関数がループを続け、関数ブロック430に示すように、テキストストリングの一部をパターンに結びつけることができるか

10

20

30

40

50

どうかを判断することを含め、それぞれのパターンを処理する。その後、関数ブロック 440 において、存在する場合、様々なプレースホルダが値に結びつけられ、関数ブロック 441 において、句読点によって分離された名前のリストが結びつけられ、関数ブロック 442 において、大文字で始まる 2 つのワードをフルネームとして見つけ、大文字で始まるかどうかを判断するためにワード後にあるスペース後の次の文字を捕らえることでフルネームが処理される。次に、関数ブロック 443 において、時間がストリングから適切な方法でパースされ、関数ブロック 444 においてブランクスペース後の次のワードがパースされる。次に、関数ブロック 445 において、会社、トピック、又は場所といった大文字で始まる連続する語句が結びつけられ、関数ブロック 446 において、好適な実施形態による更なる処理のために、ブランク後の次のワードが取得される。マッチミーティングフィールド処理に続いて、パターンの先頭であるインジケータの位置を特定するために関数ブロック 450 が利用され、ブランク後の次のワードが関数ブロック 452 に示すように取得され、関数ブロック 454 に示すように、このワードがチェックされ、インジケータかどうか判断される。次に、関数ブロック 460 において、パターンの終わりにないインジケータの位置を特定するためにストリングがパースされ、改行又はキャリッジリターンに続くものといった不必要な余白後の次のワードが関数ブロック 462 に示すように処理され、関数ブロック 464 に示すように、このワードが分析され、インジケータかどうか判断される。次に、関数ブロック 470 において、次のストリングの処理に備えるために、一時レコードがヌルセットにリセットされ、関数ブロック 480 において、ミーティングレコードが更新され、関数ブロック 482 において、チェックが実行され、ミーティングレコードをパースする前に、ミーティングレコードにエントリが既に行われたかどうか判断される。

10

20

【0088】

特定済みミーティングフィールドの使用：

重要と考えるミーティングテキスト内のフィールドを特定したため、これに対して行える事柄は数多く存在する。最も重要なパターンマッチングの応用の 1 つは、当然ながら、最終的には *Alta Vista* 及び *News Page* に発信する、私達が構築したクエリを改善することである。BF に追加することが可能なパターンマッチングの結果を利用したオプション及び強化は他にも数多く存在する。こうした他のオプションについては、次のセクションで説明する。本セクションの目的は、優れたサーチ結果の取得を支援するために、パターンマッチングで得られた結果をどのように使用できるかに関する十分な理解を読者に与えることである。

30

【0089】

図 5 は、クエリを準備し、インターネットから情報を取得するための好適な実施形態による詳細な処理のフローチャートである。処理は、関数ブロック 500 で開始され、Web サーチエンジンを利用したインターネットサーチに備えるラップ関数を処理するために、直ちに関数ブロック 510 へと続く。このサーチが、*Alta Vista* サーチエンジンを利用する場合は、関数ブロック 530 において、システムはミーティングレコードから情報を取り出し、関数ブロック 540 乃至 560 において、このサーチエンジンに発信するクエリを形成する。このサーチが、*News Page* サーチエンジンを利用する場合は、関数ブロック 520 において、システムはミーティングレコードから情報を取り出し、関数ブロック 521 乃至 528 においてクエリを形成する。

40

【0090】

Alta Vista サーチエンジン：

Alta Vista サーチエンジンの長所は、強化された柔軟性を提供することである。高度なクエリ方法を使用することで、あらゆる種類のブールクエリを構築し、望み通りのサーチのランク付けが可能となる。しかしながら、*Alta Vista* の最大の欠点の 1 つは、大きなクエリを取り扱うのに優れておらず、無関係な結果を返す可能性が高くなることである。ミーティングテキスト内でトピック及び会社を特定可能である場合、優れた結果が期待される非常に短い包括的なクエリを形成することができる。更に、発見

50

したトピックにも注目する必要がある。特にユーザが会社について既によく知っており、その会社と数多くのミーティングを行っている場合、会社に関する情報を見つけることは、そのユーザにとって大きなメリットではない可能性がある。こうしたユーザがリサーチしたいのはトピックについてである。

【0091】

News Page検索エンジン：

News Page検索エンジンの長所は、有効な社名を与えることが可能である場合、最新のニュースの検索において優れた仕事を行うことである。そのため、ニュースページWebサイトにクエリを発信する時は、特定可能なあらゆる社名を送り、社名が発見できない場合のみ、発見したトピックをクエリの形成に使用する。どちらも発見できない場合、検索は実行されない。Alta Vistaに発信するクエリの形成に利用されるアルゴリズムは図7に例示されている。News Pageに発信するクエリの形成に使用するアルゴリズムは図8に例示されている。

10

【0092】

以下の表では、好適な実施形態によるそれぞれの関数を詳細に説明している。関数が登場する順番は、可能な限りプロセスフローに似せている。関数が数回呼び出される状況が存在する時、この関数は、この関数を呼び出す第一の関数の後に記述され、その説明は、この関数を呼び出すその後の関数の後には繰り返されない。

【0093】

【表5】

20

手続き名	タイプ	呼び出し元	説明
Main	パブリック サブ	無し	これはプログラムが最初に開始するメイン関数である。適切なパラメータ(インターネットタイムアウト、ストップリスト等)によって BF を初期化し、プログラムのメイン部分を開始するために GoBF を呼び出す。
ProcessCommandLine (BF.Main)	プライベートサブ	Main	この関数はコマンドラインをパースする。Munin からの入力の始まりを示す区切り文字が定数 CMD_SEPARATOR に格納されていると仮定する。
CreateStopList (BF.Main)	プライベート関数	Main	この関数はミーティングテキストから不要なワードをパースし除外するために将来使用するストップリストを設定する。 連続したチェックを可能にするために、それぞれのワードの両側にコンマが存在する。
CreatePatterns (BF.Pattern Match)	パブリック サブ	Main	この手続きは、ミーティングテキストの部分を結びつけることができる可能性のあるすべてのパターンを作成するために、BF が最初に初期化される時、一度だけ呼び出される。パターンは必要に応じて、どれほど多くの要素であっても含むことができる。2つのタイプの要素が存在する。第一のタイプの要素はインジケータである。これらは、後に続くミーティングフィールド(会社等)の可能性のあるものを区切る実際のワードである。こうしたインジケータのほとんどは予測されるストップワードで、これは、ストップワードが通常はすべてのミーティングテキストに共通するワードであり、このワードがパターンを形成するのが意味をなすためであ

10

20

30

40

			る。第二のタイプの要素は、プレースホルダを表す特殊ストリングである。プレースホルダは常に\$*\$の形式であり、*は PEOPLE、COMPANY、TOPIC_UPPER、TIME、LOCATION、又は TOPIC_All のいずれかにすることができる。パターンは、2つのタイプの要素の一方で始めることが可能であり、任意の数／タイプの要素を含み、いくらでも長くすることが可能である。この手続きは、表内のそれぞれのパターンに関する新しいパターンレコードを動的に作成し、更に、パターン内のそれぞれの要素に関する tAPatternElements を動的に作成する。加えて、パターン内でインジケータを置換することが可能であるという概念が存在する。例えば、\$PEOPLE\$ of \$COMPANY\$のパターンは \$PEOPLE\$ from \$COMPANY\$のパターンと同様である。「from」は「of」の置換文字となる。私達の構造は、こうした置換文字の必要性を表現できるべきである。	10
GoBF (BF.Main)	パブリック サブ	Main	これはBFのパーシング及びサーチングサブルーチンの両方を呼び出すラッパ手続きである。Munin にデータを返送することにも責任を有する。	30
ParseMeetingText (BF.Parse)	パブリック 関数	GoBackGroundFinder	この関数は、初期のミーティングテキストを取り出し、レコードのユーザIDと、タイトル、ボディ、参加者リスト、場所、及び時間を含むミーティングテキストの他の部分とを特定する。加えて、私達は、キーワードのみが残るように、すべての不必要なワードをオリジナルのミーティングタイトル及びミーティングボディから排除するために、ヘルパ関数 ProcessStopList を呼び出す。パースして除外された情報	40

			は、MeetingRecord 構造に格納される。注意点として、この関数はエラーチェックを行わず、ほとんどの場合はミーティングテキストが Munin によって正しくフォーマットされていると仮定する。重要な変数は、この MeetingRecord が現在のミーティングに関するあらゆる情報のための temp フォルダであることである。これは最終的には呼び出し元に戻される。
FormatDelimitation (BF.Parse)	フライング ート	ParseMeetingText, DetermineNumWords, GetAWordFromString	区切り文字を配置可能な4種類の方法が存在する。私達は、ストリング内のフィールドを除いて区切り文字が存在しない Case4 へとこれらを削減することで、これらすべてのケースに対処する(例えば、A::B::C)。
DetermineNumWords (BF.Parse)	パブリック 関数	ParseMeetingText, ProcessStopList	この関数は、ストリング(stInEvalString)内に存在するワードの数を判断する。この関数は、stSeparator において指定された指定分離記号によってそれぞれのワードが分離されていると仮定する。リターンタイプは、ストリング内のそれぞれのワードが stSeparator によって分離されていると仮定して発見されたワードの数を示す整数である。この関数は常に GetAWordFromString と共に使用され、GetAWordFromString を呼び出す前に呼び出されるべきである。
		ParseMeetingText, ProcessStopList	この関数は、ストリング内のそれぞれのワードが変数 stSeparator に含まれる指定分離記号によって分離されていると仮定してストリング(stInEvalString)のithワードを抽出する。ほとんどの場合において、この関数は DetermineNumWords と共に使用する。この関数は、必要なワードを戻す。この関数は、iがストリング内のワードの合計数を超えないように、或

10

20

30

40

			いはゼロ以下にならないように、iInWordNum が境界内にあることを確認するためにチェックを行う。境界を超えている場合、空のストリングを戻し、何も得られないことを示す。最初に DetermineNumWords を呼び出すことで、これが起こらないように努力する。	10
ParseAndCleanPhrase (BF.Parse)	フライング関数	ParseMeeting Text	この関数は、最初にワードを捕らえ、誰も求めていないものを取り去るために CleanWord に送る。ParseWord 内にはワードを削除するものが存在するため、ボディのループを続け、関数全体を削除することなくワードを拒絶する方法が必要となる。KeepCleanWord を推測し、戻り値がokであることをチェックした後、私はワードを有するため、これをパースチェーンに送る必要がある。このチェーンは、ParseCleanPhrase→CleanWord→EvaluateWord へ進む。ワードが削除されることなくチェーン全体を通過する場合、これは最終的にキーワードストリングに追加され、最初は「／」を区切り文字としてチェックし、その部分を抽出する関数となる。私はこれを「StitchFace」と呼ぶ(デニスはもっとノーマルであり、これを GetAWordFromString と呼ぶ)。これがワードを発見した場合、それぞれのワードは、次にチェーンへと送られる。これらが追加又は削除されることなくチェーン全体を通過した場合、これらは捨てられずに追加される。	20 30 40
FindMin (BF.Parse)	フライング関数	ParseAndCleanPhrase	この関数は6つの入力値を取り出し、最小の非ゼロ値がどれかを確認するために評価を行う。5つの入力値を昇順でソートできるように、これは最初にフォルダとして配列を作成する。したがって、最小	

			値は、配列の最初の非ゼロ値となる。非ゼロ値を発見せずに配列全体の調査が終わった場合、エラーがあることが分かり、この関数を終了する。
CleanWord	プライベート関数	ParseAndCleanPhrase	この関数は、ミーティングテキスト内のワードを整理することを試みる。何よりも最初に、ストリングが有効な長さかどうかを判断する。その後、これを一連のテストに通し、整理されているかどうかを確認し、必要な時は、ワードを編集し、不要な文字を取り去る。こうしたテストには、ファイル拡張子、文字ではないもの、数字等を取り除くことが含まれる。
EvaluateWord (BF.Parse)	プライベート関数	ParseAndCleanPhrase	この関数は、オリジナルのミーティングテキストからワードを排除するかどうかを判断できるように、このワードがストップリストにあるかどうかを確認するためにテストを行う。ワードがストップリストにない場合、これはキーワードとして残り、この関数はエラーを起こさず、きれいに終了する。しかしながら、このワードがストップワードである場合、エラーが戻される必要がある。誤ってサブストリングを検索しないように、入力テストストリングを正しく区切る必要がある。
GoPatternMatch (BF.PatternMatch)	パブリックサブ	GoBF	この手続きは、QueryMethod が、あらゆるパターンマッチング内容を実行する必要があることを意味するコンプレックスクエリに設定されている時に呼び出される。これは単純なラッパ関数で、いくつかの配列を初期化し、その後、タイトル及びボディに関するパターンマッチングを呼び出す。
MatchPatterns (BF.PatternMatch)	パブリックサブ	GoPatternMatch	この手続きは、パターン表内のすべてのパターンをループし、sInEvalString によって指定されたミーティングテキスト内の異

10

20

30

40

Match)			なるフィールドを特定することを試みる。デバッグの目的では、特定のパターンが何回トリガされたかを作表することを試み、これを gTabulateMatches に格納し、どのパターンが最も多く使われたかを確認する。40又は50のテストストリングのバッチファイルを実行可能にし、依然としてパターンが何回トリガされたかを知ることができるようにする必要があるため、gTabulateMatches はグローバルとして格納される。	10
MatchAPattern (BF.Pattern Match)	フライング関数	MatchPatterns	この関数は現在のパターンのそれぞれの要素を調べる。最初に、要素がプレースホルダ又はインジケータであるかを判断するために評価する。プレースホルダである場合は、そのプレースホルダに何らかの値を結びつけようと試みる。インジケータである場合は、位置特定を試みる。しかしながら要領がある。現在の要素がパターンの先頭にあるか、そうではないかに応じて、異なる行動をとる必要がある。先頭にある場合は、インジケータ又はプレースホルダを探す必要がある。見つけれない場合は、現在のパターンが存在していないことが分かり、終了となる。しかしながら、先頭ではない場合は、まだどこかに先頭が存在する可能性があるため、調査を継続する。この場合は再試行する。	20
		MatchAPattern	この関数は、大きなスイッチ文を使用して、最初に問題となっているのがどの種類のプレースホルダかを判断し、プレースホルダのタイプに応じて、その後呼び出される BindNames、BindTime 等の関数において指定されるように、特定の要件及び異なる連結基準を有する。連	40

			結が成功した場合は、これを推測レコードに追加する。
BindNames (BF.Pattern Match)	プライベ ート関数	MatchMeeting Field	この関数においては、名前を対応するブ レースホルダ\$PEOPLE\$に対応させるこ とを試みる。名前は、大文字で始まる2 つの連続するワードとして定義される。 更に、and、,、又は&によって結合され た名前の連続も検索するため、こうした 分離記号のいずれかが確認できなくなる まで調べる。注意点として、単一ワード の名前を結びつける必要はなく、これ は、おそらく一般的すぎる内容から、広 範で無関係な結果が生じないようにする ためである。この関数は1つの名前を結 びつける BindAFullName を呼び出すた め、ある意味において BindNames は、 BindAFullName からのすべての結果を集 める。
BindFullNa me (BF.Pattern Match)	プライベ ート関数	BindName	この関数はフルネームを結びつけること を試みる。\$PEOPLE\$ブレースホルダが パターンの先頭ではない場合、ストリン グの先頭のを常に削除してきたた め、これがテストストリングの始まりに なることが分かる。先頭である場合は、フ ルネームに似たものが見つかるまでサ ーチを行う。見つからない場合は、テキ スト全体にこうしたパターンは存在せず、 このパターンから完全に離れる。これに より最終的には、MatchPattern の次の パターンが戻されるべきである。
GetNextWo rdAfterWhit eSpace (BF.Pattern Match)	プライベ ート関数	BindAFullNam e、BindTime、 BindCompany TopicLoc	この関数は、テストストリングの次のワー ドを捕らえる。これは、余白、@、又は/ の後の次のワードを探す。このワード は、こうした余白又は分離記号の別の1 つに遭遇した時、終了すると定義され る。

10

20

30

40

BindTime (BF.Pattern Match)	フライベ ート関数	MatchMeeting Field	直後のワードを取り出し、時間パターンに似ているかどうかを確認する。これにより時間を見つけた場合、これをレコードに追加する必要がある。おそらく時間パターンは更に追加すべきである。しかし、特に Outlook のようなツールがあるため、最近ではタイトルに時間を入れるのは好まれないようである。
BindCompanyTopicLoc (BF.Pattern Match)	フライベ ート関数	MatchMeeting Field	この関数は、大文字で始まる連続したストリングを見つけ、MatchMeetingField からのリファレンスによって渡される stMatch に結びつける。大文字で始まる連続したストリングは、... 等によって中断されない大文字で始まるワードの連続である。おそらく中断のリストには更に内容を追加することができる。
LocatePatternHead (BF.Pattern Match)	フライベ ート関数	MatchAPattern	この関数は、インジケータである要素の位置を特定することを試みる。注意点として、このインジケータは、パターンの先頭にあるべきであり、それ以外の場合、これは代わりに LocateIndicator へと行ってしまう。そのため、捕らえるワードが存在しなくなるか(終了)、或いは探しているインジケータの1つが見つかるまで、次のワードを捕らえ続ける。
ContainInArray (BF.Pattern Match)	フライベ ート関数	LocatePatternHead 、 LocateIndicator	この関数は非常にシンプルである。配列内のすべての要素をループし、マッチするストリングを見つける。
LocateIndicator (BF.Pattern Match)	フライベ ート関数	MatchPattern	この関数は、インジケータである要素の位置を特定することを試みる。注意点として、このインジケータは、パターンの先頭にあるべきではなく、それ以外の場合、これは代わりに LocatePatternHead へと行ってしまう。このため、パターンが満たされる場合、次に捕らえるワードは

10

20

30

40

			インジケータである必要があり、そうでなければ失敗することになる。したがって、1ワードのみを捕らえ、有効なインジケータかどうかを確認するためにテストし、その後、結果を戻す。	
InitializeGuessRecord (BF.Pattern Match)	フライヘートサブ	MatchAPattern	この関数は一時テスト構造を再初期化し、すでに情報をパーマネント構造に転送しているため、それぞれが1つの要素を有するように、これを再び初期化することが可能である。	10
AddToMeetingRecord (BF.Pattern Match)	フライヘートサブ	MatchAPattern	この関数は、tInCurrGuesses に格納される情報が、パーマネントレコード tInMeetingRecord に格納される準備ができたミーティングフィールドの合法的な推測を表す有効な意味を有することが分かっている時のみ呼び出される。重複が格納されていないこと、及び更に句読点等の不要物が存在しないように、必要なものを整理するものが格納されていないことを確認するためにチェックを行う。これまでに整理を行わない理由は時間の節約である。恒久的に追加することが確認されるまで、ParseAndCleanPhrase を呼び出すリソースを無駄にしない必要がある。	20 30
NoDuplicateEntry (BF.Pattern Match)	フライヘート関数	AddToMeetingRecord	この関数は、配列内のそれぞれの要素をループし、テストストリング aString が、配列に既に格納されているストリングのいずれかと同じではないことを確認する。ContainInArray とは僅かに異なる。	
SearchAltaVista (BF.Search)	パブリック関数	GoBackGroundFinder	この関数は、Alta Vistaサーチエンジンに発信するクエリを準備する。これはクエリを発信し、その後、戻ってきた結果を、検索したそれぞれの記事のタイトル、URL、及びボディ／サマリを含む適切なフォーマットにおいてパーする。検	40

			索される記事の数は、定数 NUM_AV_STORIES によって指定される。 重要な変数には、発信するクエリを格納するのに使用される stURLAltaVista と、stURLAltaVista によって指定されたページからの html を格納するのに使用される stResultHTML とが含まれる。
ConstructAltaVistaURL (BF.Search)	プライベート関数	SearchAltaVista	この関数は、高度クエリサーチモードを使用してアルタビスタサーチエンジン用の URL スtring を構築する。これには、使用されるキーワードと、言語と、サーチをどのようにランク付けしたいかが含まれる。パターンマッチングユニットの結果を使用する必要があるかどうかに応じて、クエリを異なる形で構築する。
ConstructSimpleKeyword (BF.Search)	プライベート関数	ConstructAltaVistaURL、 ConstructNewsPageURL	この関数は、入力ミーティングレコードの stTitleKW 又は stBodyKW フィールドに格納されたキーワードのリストを進んでいき、入力変数 stInConnector によって決定される結合子によって分離されるそれぞれのキーワードを有する1つのStringにこれらをリンクする。この新たに構築されたStringを戻す。
ConstructComplexAVKeyword (BF.Search)	プライベート関数	ConstructAltaVistaURL	この関数はAlta Vistaサイトに送るキーワードを構築する。単純にタイトルからクエリまでのすべてのキーワードを取り出す ConstructSimpleKeyword とは異なり、この関数はBFのパターンマッチングプロセスの結果を調べ、クエリの構築のために特定の社名又はトピックが特定できるかどうかを確認する。クエリには、特定された会社及びトピックが含まれ、会社又はトピックが特定できない場合はサンプルクエリがデフォルトとなる。
JoinWithConnectors	プライベート関数	ConstructComplexAVKey	この関数は、String内のワード間のスペースを、入力によって指定された結

10

20

30

40

(BF.Search)		Word, ConstructComplexNPKey Word, RefineWithRank	合子に単純に置き換える。
RefineWith Date (この 時点では呼 び出されな い) (BF.Search)	プライベート関数	ConstructAlta VistaURL	この関数は、アルタビスタクエリの日付部分を構築し、URLのこの部分をストリングとして戻す。アルタビスタサーチが、PAST_NDAYS まで遡った項目に関して行われるようにする。
RefineWith Rank (BF.Search)	プライベート関数	ConstructAlta VistaURL	この関数は、高度クエリサーチをランク付けするために、Alta Vistaに渡す必要があるストリングを構築する。シンプルクエリを構築している場合は、タイトルからすべてのキーワードを取り出す。コンプレックスクエリの場合は、ConstructComplexAVKeyword においてクエリを形成する方法とほぼ同じように、会社及びトピックからのワードを取り入れる。
IdentifyBlock (BF.Parse)	パブリック関数	SearchAltaVist st 、 SearchNewsPage	この関数は、特定の位置(iStart)で始まる入力として与えられた開始及び終了タグによってマーク付けされたストリング内のブロックを抽出する。検索されたブロックは、タグ自体を含まない。ブロックが特定の区切り文字によって特定できない場合、リファレンスとして使用するために渡されるパラメータ iReturnSuccess を通じて、不成功を戻す。戻りタイプは検索されたブロックである。
		SearchAltaVist st 、 SearchNewsPage	この関数は、遭遇したエラーがタイムアウトエラーによるものかどうかを判断する。マウスをデフォルトの矢印に復帰させ、その後、タイムアウトである場合は

10

20

30

40

			真を、違う場合は偽を戻す。
SearchNewsPage (BF.Search)	パブリック関数	GoBackGroundFinder	この関数はNews Pageサーチエンジンに発信するクエリを準備する。これはクエリを発信し、その後、戻ってきた結果を、検索したそれぞれの記事のタイトル、URL、及びボディ／サマリを含む適切なフォーマットにおいてパースする。検索される記事の数は、定数 NUM_NP_STORIES によって指定される。
ConstructNewsPageURL (BF.Search)	プライベート関数	SearchNewsPage	この関数はNews Pageサイトへ送るURLを構築する。入力ミーティング記録に含まれる情報を使用して、どのキーワードを使用するかを判断する。更に、必要なのがシンプルクエリかコンプレックスクエリかに応じて、ストリングを形成するために異なる関数を呼び出す。
ConstructComplexNPKeyWord (BF.Search)	プライベート関数	ConstructNewsPageURL	この関数はNews Pageサイトに送るキーワードを構築する。単純にタイトルからクエリまでのすべてのキーワードを取り出す constructKeyWordString とは異なり、この関数はBFのパターンマッチングプロセスの結果を調べ、クエリの構築のために特定の社名又はトピックが特定できるかどうかを確認する。ニュースページは社名がある時に最も優れた動作をするため、社名のみを使用し、社名がない場合のみ、トピックを使用する。

10

20

30

ConstructOverallResult (BF.Main)	プライベート関数	GoBackGroundFinder	この関数は、ストリング(stInStories)の配列と、現在のミーティングに関数情報を格納する MeetingRecord とを入力として取り込む。配列内のそれぞれの要素には、それぞれの情報ソースから検索した維持が格納される。この関数は、Munin へ送る適切な出力を単純に構築し、これには BF の応答であることを Munin に知らせるための戻りメッセージタイプが含まれ、更に、オリジナルのユーザIDとミーティングタイトルが含まれ、Munin は対象となるミーティング BF がどれであるかを知ることができる。
ConnectAndTransferToMunin (BF.Main)	パブリックサブ	GoBackGroundFinder	この関数により、BackGround Finder は、Munin に接続し、最終的には、Munin に情報を転送することができる。TCPプロトコルではなくUDPプロトコルを使用するため、リモートホスト及びポートを正しく設定する必要がある。gResultOverall を格納するためにグローバルストリングを使用し、これは、UDPでは必要ないものの、TCPでは必要になり、スイッチバックを行う場合にコードを変更したくないためである。
DisconnectFromMuninAndQuit (BF.Main)	パブリックサブ		

10

20

30

【0094】

図6は、好適な実施形態に従った、Alta Vista及びNews Pageサーチエンジンに対するサーチの準備及び発信に利用する実際のコードのフローチャートである。処理は関数ブロック610で開始され、ここでコマンドラインが利用され、特定のカレンダー情報によってカレンダーエントリが更新される。このメッセージは関数ブロック620に従って次に掲示され、関数ブロック630に従って現在のミーティング情報を格納するためにミーティングレコードが作成される。次に、関数ブロック640において、クエリがAlta Vistaサーチエンジンに発信され、関数ブロック650において、クエリがNews Pageサーチエンジンに発信される。サーチエンジンからメッセージが戻されると、これは関数ブロック660に示すように結果データ構造に格納され、この情報は関数ブロック670の記述のように、処理され、ミーティングの準備で使用するファイル内においてサマリ形式で格納される。

40

【0095】

50

図7は、好適な実施形態によるクエリの作成に関する詳細を提示している。処理は関数ブロック710で開始され、ここでミーティングレコードがパースされ、会社、人物、トピック、場所、及び時間である可能性のあるものが取得される。その後、関数ブロック720において、少なくとも1つのトピックが特定され、関数ブロック730において、少なくとも1つの社名が特定され、最後に関数ブロック740において、ユーザの最終的な消費のために、どのデータをファイルに送信するかについて判断がなされる。

【0096】

図8は、図7で提示されたクエリのテーマのバリエーションである。ミーティングレコードは関数ブロック800においてパースされ、会社が関数ブロック820において特定され、トピックが関数ブロック830において特定され、最後に関数ブロック840において、クエリの策定にトピック及び又は社名が利用される。

10

【0097】

特定のユーザ要件のために様々な特定の特徴を追加する代替実施形態については以下で説明する。

【0098】

パターンマッチングのターゲット率強化：

BFの性能を高めるためには、手続き「CreatePatterns」に、パターン／パターングループを追加する。パターンを宣言する既存のコードは、将来のパターンのテンプレートとして使用することができる。すべてが動的配列として格納されるため、カット及びペーストによってコードを再使用するのが便利である。値をプレースホルダに関連付けることに責任を有する関数BindName、BindTime、BindCompanyLocTopicは強化することができる。この強化は、結びつける値の数を増やすために、特定のミーティングフィールドを結びつける基準のセットを増やすことで実現される。例えば、BindTimeは現在、##：##又は#：##の形式のすべての値を受け入れて結びつける。結びつけることが可能な時間を増やすためには、更に美的な時間用語「o'clock」が後に続く数字1乃至12を受け入れるようにすることができる。語彙に基づく認識アルゴリズムと、BFが行うそれぞれの推測に精度レートを割り当てることとにより、有効となる特定の閾値を満たす唯一の推測が可能になる。

20

【0099】

パターンマッチングを通じてシステムがどの場所を特定したかに応じて、或いはミーティング場所としてユーザがどの場所を示したかに応じて、好適な実施形態によるシステムは、ランチ／ディナー／朝食というワードを検出した時には必ず、複数の優れたレストランを提案する。更に、会社ファインダのようなサイトを使用して、手元にあるものが本当に社名であることを確認可能であり、或いはパターンマッチングで社名が特定できない場合は、会社ファインダのWebサイトを「辞書」として使用し、大文字で始まる特定のワードが社名を表すかどうかを判断することができる。特定した会社の株価及び急報を表示することも可能である。

30

【0100】

好適な実施形態による無線バーゲン特定：

図9は、従来の物理的な非Web小売環境においてWebベースの比較ショッピングを可能にするために設計された装置及びソフトウェアのハードウェア及び論理制御フローを表したフロー図である。インターネットプロトコル能力を有する無線電話又は同様のハンドヘルド型無線装置920は、小型バーコードリーダ910（電話の内部又は短いケーブルで設置される）と統合され、書籍その他の製品900の統一商品コード（UPC）バーコードをスキャンするのに使用される。この無線装置920は、アンテナ930を介して、バーコードをポケットバーゲンファインダサービスモジュール（Webサーバで実行）940に送信し、これがバーコードを（書籍の場合は）国際標準図書番号又は（他の製品の場合は）適切な識別子に変換する。このサービスモジュールは、その後、適切なサードパーティの（複数の）Webサイトにコンタクトし、様々なWebサプライヤ950からの製品に関する価格、出荷、及び入手可能性の情報を見つける。この情報はフォーマットさ

40

50

れ、ハンドヘルド型装置のスクリーンに表示される。I P無線電話その他のハンドヘルド型装置920は、メトリコムのRicochet SE Wireless Modem等の無線モデムを利用する。この装置を利用することで、ユーザは、コーヒーストップでくつろぎながら、揺れ動いているラテを危険なほどキーボードの近くに置き、古びた小さなテーブルの上に乗ったポータブルコンピュータにより、電話回線による直接接続に匹敵する速さでインターネットにアクセスできる。

【0101】

8オンス(227g)のRicochet SE Wireless Modemは、タバコのパックほどの大きさであり、セットアップは極めて単純で、付属のベルクロの部品によってこのモデムをポータブル機器の裏側に取り付け、ケーブルをシリアルポートに差し込み、短く太いアンテナを

10

【0102】

当然ながら、無線性能には、従来のダイヤルアップ接続と同様の信頼性はない。サンフランシスコのいくつかの場所では、窓の近くにいる限り、強力な接続を得ることができた。しかし、CNETの総レンガ造りの本社内では、Ricochetはまったく接続できなかった。オンラインでつながっている時は、遅い速度へのゆっくりとした低下と共に28.8kbp/sまでの性能が利用できる。しかし、遅い速度でも失望はなかった。代替方法――セルラモデムによる接続――に比べ、Ricochetは遥かに速く、信頼性があり、使用コストが少

20

【0103】

したがって、この無線モデムを利用することで、ユーザは好適な実施形態に従って、Webサーバソフトウェア940を利用して、適切な製品950を特定し、次に、適切な装置の(複数の)キーを使用してサプライヤを選択し、注文を入れることができる。バーゲンファインダサービスモジュールは、その後、適切なサードパーティWebサプライヤ960により、注文を完了させる。

【0104】

my Site! パーソナルWebサイト及び意図価値ネットワークのプロトタイプ:
my Site!は、好適な実施形態によるインパクトの強いインターネットベースのアプリケーションであり、買い手中心世界におけるパーソナルWebサイトを介して、それぞれの顧客にサービスを与え、パーソナライズされた体験を提供するというテーマに注目したものである。このサービスは顧客の意図――基本的な生活のニーズ又は広範なプランニングを要する判断と、財務プランニング、ヘルスケア、個人的及び職業的成長、家族生活、その他の問題といったいくつかの次元に渡る調整――を満足させることを中心に直観的に編成されている。それぞれのメンバーは自分自身のプロフィールを保有及び維持しており、これにより、自分自身を限定的にターゲットとするシステムにおいてコンテンツを作成及びブラウズすることが可能となる。製品又はサービスの要求を入力した時点から、支払いの完了まで、インテリジェントエージェントを利用して、リサーチが実施され、取引が実行され、アドバイスが提供される。高度なプロファイリング及びフィルタリングを使用することで、インテリジェントエージェントはユーザについて学び、提供するサービスを改善する。顧客の意図には、日常ロジスティクス管理(例えば、電子メール、カレンダー、連絡先、予定リスト、請求書の支払、ショッピング、及び旅行プランニング)と、新しいコミュニティへの移動(例えば、新しい生活場所を見つけること、家族の所有物を移動させること、旅行に出かけること及び保険担保を送ること、仕事及び個人の連絡先に通知すること、新しいコミュニティについて学ぶこと)とが含まれる。消費者の観点から、my Site!は、ユーザが関連製品及びサービスにアクセスし、極めて容易かつ便利に日常のタスクを達成することが可能な中心的場所を提供する。

30

40

50

【0105】

ビジネスの観点から、my Site! は、効果的に顧客を集め、サービスを提供し、保持するための付加価値のある革新的な方法を意味する。意図価値ネットワークにより、ユーザは、パーソナライズされたサイトに入り、学習するインテリジェントエージェントの支援により、ネットワーク参加者とシームレスに対話することができる。好適な実施形態による意図価値ネットワークは優れた価値を提供する。カスタマイズされた情報、アドバイス、及び製品に対する、1日24時間、週7日のアクセスが提供される。この情報はパーソナライズされているため、それぞれのユーザは、必要なターゲットユーザとの関連性を保証するために高度にカスタマイズされたコンテンツを閲覧することになる。

【0106】

エゴセントリックインタフェース：

エゴセントリックインタフェースは、特定のユーザのニーズ、好み、及び現在のコンテキストを満足させるために作成されたユーザインタフェースである。これは、中央プロフィールデータベースに格納されたユーザの個人情報を利用して、インタフェースをカスタマイズする。ユーザは、セキュリティ許可をオンにし、インタフェース要素及びコンテンツに関する好みを設定することができる。エゴセントリックインタフェースに統合されるコンテンツは、ユーザについての関連情報によってカスタマイズされる。コンテンツを表示する時、エゴセントリックインタフェースには、コンテンツがどのようにユーザに関連するかを実証する形で、コンテンツとユーザとの間の関係が含まれる。例えば、ユーザが契約済みの来るべきスキー旅行に関する情報を表示する時、このインタフェースには、スキー旅行中に同じ地域にいる他の人物等、ユーザの個人カレンダー及び連絡先リストからのイベントに関する情報が含まれる。これは、個別のユーザがよく知っているコンテキストに新しい情報の断片を入れる役割を果たす。

【0107】

図10Aは、ワールドワイドウェブでの意図価値ネットワークアーキテクチャの実施を表している。簡略化の目的から、この図ではセキュリティ、スケーラビリティ、及びプライバシーに関する複雑性を無視している。顧客は、Netscape Navigator又はMicrosoft Internet Explorer等、インターネットに接続されたパーソナルコンピュータ上で実行されている任意のインターネットWebブラウザ1010、或いは無線能力を有する携帯情報端末によって、意図価値ネットワークにアクセスできる。意図価値ネットワークにアクセスする複数の方法の詳細な説明については図17を参照せよ。顧客は固有の名前又はインテグレータのWebサーバ1020に関連するIPアドレスによって意図価値ネットワークにアクセスする。インテグレータは、意図データベース1030、コンテンツデータベース1040、サプライヤプロフィールデータベース1050、及び顧客プロフィールデータベース1060といったリソースの組み合わせを使用して、意図価値ネットワークを作成する。

【0108】

意図データベース1030は、意図の構造と、意図を充足させるのに必要な製品及びサービスのタイプとに関するすべての情報を格納する。このデータベースの情報には、意図のステップ、関心のあるエリア、レイアウトテンプレート、及びパーソナライゼーションテンプレートが含まれる。コンテンツデータベース1040は、アドバイス、紹介情報、パーソナライズコンテンツ、満足度評価、製品評価、及び進捗レポート等、意図に関連するすべての情報を格納する。

【0109】

サプライヤプロフィールデータベース1050には、意図に統合された製品及びサービスのプロバイダに関する情報が含まれる。このデータベースに含まれる情報は、意図フレームワークとサプライヤとの間のリンクを提供する。これには、製品リストと、特徴及び説明と、サプライヤの製品Webサイトのアドレスとが含まれる。顧客プロフィールデータベース1060には、名前と、住所と、社会保障番号及びクレジットカード情報と、個人的嗜好と、行動情報と、履歴と、Webサイトレイアウトの好みと等、顧客に関する個人

情報が含まれる。サプライヤのWebサーバ1070は、情報及び取引のサポートを顧客に提供するために必要なサプライヤのあらゆるデータベースへのアクセスを提供する。

【0110】

製品情報データベース1080は、特徴、入手可能性、及び価格等、あらゆる製品関連情報を格納する。製品注文データベース1090は、あらゆる顧客の注文を格納する。このデータベースとのインタフェースは、SAP、Ba an、O r a c l eその他が提供している統合基幹業務アプリケーションを通じて行うことが可能であり、或いはサプライヤのWebサーバ又はアプリケーションサーバを通じて直接アクセスすることができる。顧客情報データベース1091は、取引を完了するため、或いは顧客レコードを維持するためにサプライヤが必要とするあらゆる顧客情報を格納する。

10

【0111】

図10Bは、エゴセントリックインタフェース内でWebページを作成するために利用されるロジックを提示するフローチャートである。この環境では、パブリックインターネット又はプライベートイントラネット等のように、WebサーバとWebブラウザとはTCP/IPネットワークによって接続されていると仮定する。可能なWebサーバには、Microsoft Internet Information Server、Netscape Enterprise Server、又はApacheが含まれる。可能なWebブラウザには、Microsoft Internet Explorer又はNetscape Navigatorが含まれる。クライアント（つまりWebブラウザ）は、サーバ（つまりWebサーバ）に対して、特定のWebページを要求する1001。これは通常、ユーザがWebページ内のボタン又はリンクをクリックすることで達成される。Webサーバは、クライアント（つまりWebブラウザ）に格納されるユーザIDを管理するデータベース及びユーザプロフィールデータベース1003への要求によって、その特定のユーザに関するレイアウト及びコンテンツの好みを取得する1002。次に、Webサーバは、コンテンツデータベース1005から、要求されたページのコンテンツを検索する1004。次に、カレンダー、電子メール、連絡先リスト、及びタスクリストの項目等、関連するユーザ中心コンテンツが検索される1006（このプロセスの詳細な説明については図11を参照）。データベースへのクエリは、ユーザプロフィールデータベース1003内のユーザプロフィールの一部として格納されているユーザのコンテンツの好みを利用して、戻されるコンテンツをフィルタリングする。戻されたコンテンツはその後、ユーザプロフィールにおいて定義されたレイアウトの好みに従って、Webページにフォーマットされる1007。このWebページはその後、クライアントに戻され、ユーザに対して表示される1008。

20

30

【0112】

図11は、Webページに追加するユーザ中心コンテンツを検索するプロセスを表している。このプロセスは、図10Bの1006を更に詳細な形で説明している。ここではサーバが、ユーザプロフィールと、このページに統合されることになる既存のコンテンツとを既に取得していると仮定する。このサーバは、フィルタリングされたコンテンツをパースし1110、イベント、連絡先、名前、及び電子メールアドレスのインスタンスを探す。これらのいずれかが見つかった場合は、タグが付けられ、一時保有空間に格納される。次に、このサーバは、様々なデータベースに格納された任意のユーザ中心コンテンツを見つけようと試みる1120。これには、一時保存空間内のタグ付き項目を、カレンダーデータベース1140のカレンダー項目1130と、電子メールデータベース1114の電子メール項目1115と、連絡先データベース1168の連絡先項目1117と、タスクリストデータベース1118のタスクリスト項目1119と、ニュースデータベース1120のニュース項目1121とにマッチングさせることが伴う。関連するユーザ中心コンテンツを検索した後、これは共に編集され戻される1122。

40

【0113】

ユーザペルソナ：

このシステムでは、ユーザは、プロフィール情報を異なるコンテキストで有効なセットに集合させる多数の異なるペルソナを作成することができる。ユーザは、自宅用に購入を行

50

う時、1つのペルソナを作成することができる。このペルソナは、自宅の住所を含む可能性があり、このユーザがショッピング時に良質なバーゲン品を探していることを示す可能性がある。同じユーザは、仕事のコンテキストにいる時に使用することが可能な第二のペルソナを作成することができる。このペルソナは、ユーザの職場の所在地を格納する可能性があり、このユーザが特定のベンダを好むこと、或いは割引プログラムを設定している特定の企業のために働いていることを示す可能性がある。仕事関連品目のショッピングの時、ユーザは、このペルソナを使用することができる。ペルソナは規則や制限も含むことができる。例えば、仕事のペルソナは、ユーザが一社のみの旅行代理店で飛行機の予約を行い、雇用主が設定した手配規則を利用するように制限することができる。

【0114】

図12は、ユーザと、複数のペルソナと、複数のプロフィールとの間の関係を表している。ユーザレベルにあるのはユーザプロフィール1200である。このプロフィールはユーザとユーザの口座情報を表す。データベースには、口座を有するそれぞれのユーザに関する固有のレコードが1つ存在する。それぞれのユーザには複数のペルソナ1220、1230、及び1240が付属する。これらのペルソナは、複数のプロフィールを有効なコンテキストにグループ化するのに使用される。例えば、ユーザがサンフランシスコに住み、パロアルトで働いているが、タホ湖に山荘を所有しているとする。彼は、自分のサイトにアクセスする可能性のある3種類のコンテキストを有している。1つのコンテキストは仕事関連である。他の2つは家庭生活関連だが、場所が異なっている。このユーザは仕事用ペルソナ1220と、自宅用ペルソナ1230と、山荘用ペルソナ1240とを作成することができる。それぞれのペルソナは、異なる一般プロフィール1250、1260、及び1270を参照し、これにはその場所の住所が含まれる。したがって、3つの一般プロフィールが存在する。それぞれのペルソナは更に、2つの旅行プロフィールの一方を参照する。このユーザは、仕事用旅行プロフィール1280を維持しており、これにはチケットの手配及び予約の実行に関する会社のあらゆる規則が含まれる。このプロフィールは、例えば、この人物がビジネスクラス又はファーストクラスでのみ旅行すること、及び彼の好みの航空会社がユナイテッド航空であることを指定できる。仕事用ペルソナは、この仕事用旅行プロフィールを参照する。このユーザは更に、彼がエコノミクラスで旅行することを好み、全般的に安いことから払い戻し不可のセールを求めていることを指定する自宅用旅行プロフィール1290を維持することが可能である。自宅用ペルソナ及び山荘用ペルソナの両方が自宅用旅行プロフィールを指し示す。

【0115】

図13は、ペルソナ概念をサポートするデータモデルを表している。ユーザ表1310には、システム内に口座を有するそれぞれのユーザの記録が含まれる。この表には、ユーザ名及びパスワード1320と、固有の識別子とが含まれる。それぞれのユーザは複数のペルソナ1330を有することが可能であり、これはプロフィール1340と呼ばれる更に特化した構造のコンテナとして機能する。プロフィールには、プロフィールフィールド1350レコード内の詳細な個人情報が含まれる。それぞれのプロフィールには、プロフィール制限1360レコードのセットが付属する。これらのそれぞれには、名前1370と規則1380とが含まれ、これが制限を定義する。規則は、パターンに似た形式（もしxならばy）で、これにより規則を特定の用途に制限することができる。プロフィール制限の一例には、リストに含まれる特定の航空会社でユーザが飛行機を予約できないことを定めた規則がある。このプロフィール制限は、例えばユーザの雇用主によって設定された「仕事用」ペルソナの「旅行」プロフィールに含めることができる。それぞれのプロフィールフィールドには更に許可1390のセットが含まれ、これはそのレコードに含まれる。こうした許可は、特定のプロフィールフィールドの情報に対するどのようなアクセス権を誰が有しているかを定める。

【0116】

意図中心インタフェース：

退職又は移転のプランニングといった顧客の意図を満足させるには、特化したインタフェ

10

20

30

40

50

ースが必要である。顧客の意図は、例えば経済的安全性、住宅、及び運搬手段から、ヘルスケア、個人的及び職業的成長、及び娯楽に至るまでの多数のエリアに渡る広範なプランニング及び調整を必要とする。意図の満足させるには、消費者のニーズの充足を支援するために業界を超えて機能する相補的なビジネスのネットワークが必要となる。

【0117】

意図中心インタフェースは、ユーザが個人の意図を管理するのを支援するために設計されたユーザインタフェースである。任意の一定の時点において、インタフェースのコンテンツはカスタマイズされ、こうした特定の意図に関連するコンテンツのみが示される。意図中心インタフェースにより、ユーザは、こうした特定の意図を満足させるプロセスを管理することができる。これには、一連の別個のステップと、ユーザがアクセス可能なコンテンツエリアのセットとが関与する。任意の時点において、ユーザは異なる意図を管理するインタフェースに切り替えることも可能であり、この行為により、インタフェースのコンテンツは変化し、新たに選択された意図を満足させることに関連するコンテンツのみが含まれるようになる。

10

【0118】

図14は、意図中心インタフェースをサポートするのに必要なデータモデルの詳細な説明を提示している。それぞれのユーザペルソナ1410（ペルソナのデータモデルの詳細な説明については図13を参照）は、任意の数のアクティブユーザ意図1420を有する。それぞれのアクティブユーザ意図にはニックネーム1430が与えられ、これはユーザがスクリーン上で見る表示名である。それぞれのアクティブユーザ意図は更に、多数のデータフィールド1440を含み、これにはユーザとの対話全体を通じて収集された任意のユーザデータが含まれる。例えば、ユーザがスクリーン上のフォームを埋め、そのフィールドの1つが社会保障番号である場合、対応するデータフィールドは、名前＝「SSN」1460、値＝「999-99-9999」1460を含むことになる。それぞれのユーザ意図は更に、意図ステップ1470完了ステータスを追跡する。完了1480フィールドは、ユーザがステップを完了したかどうかを示す。すべてのユーザ意図は、一般意図1490のユーザ固有バージョンであり、一般意図はあらゆるユーザのための意図に関するデフォルトモデルである。一般意図は、意図のサブステップに付属するカスタム規則1411及び1412を通じてカスタマイズされる。こうしたカスタム規則は、システムが個々のユーザのプロフィール情報を使用して、それぞれのユーザの意図をどのようにカスタマイズするかを表すパターンである。

20

30

【0119】

統計エージェント：

エージェントは、それぞれのユーザの主要な統計値を追跡する。こうした統計値は、ユーザに特定の行動を促すために仮想現実ペットのたまごっちと同様の形で使用される。こうした記録される統計値は、ログインの頻度、ニュース記事等のコンテンツの評価の頻度、及び特定の期間に実行したタスク数によって測定されるエージェントの活動である。この情報は、感情的に訴えかけて特定の行動をユーザに促すために、システムによって使用される。

【0120】

図15は、エージェントの現在の統計値を表示するページを生成するプロセスを表している。ユーザがクライアントブラウザによりエージェント統計ページを要求した時1510、サーバはユーザのプロフィールデータベース1530からユーザの統計値を検索する1520。次に、このサーバは、標準化された統計値セットを作成するために必要な数学的計算を実行する1540。次に、このサーバは、コンテンツデータベース1560から、ユーザ中心統計値を計算するために使用する式を検索する1550。その後、一般式と、そのユーザの統計値とを使用して、グラフが生成される1570。このグラフは、統計ページを作成するためのテンプレートに挿入される1580。その後、このページがユーザに戻される1590。

40

【0121】

50

パーソナライズ製品レポートサービス：

このシステムは、ユーザプロフィールに基づいてユーザ毎にカスタマイズされた消費者レポートのようなサービスを提供する。このシステムは、製品の品質と、多数の次元での望ましさとに関するユーザからの評価を記録及び提供する。このシステムと従来の製品品質測定サービスとの違いは、ユーザに返ってくる評価がパーソナライズされている点である。このサービスは、ユーザのプロフィールとのマッチングが最も緊密であり、求められている製品をこれまでに評価したことがある人物を見つけることで機能する。このアルゴリズムを使用することは、ユーザに対して、そのユーザに似た人物からの統計値のみを含む製品レポートを確実に送り返すのに役立つ。

【0122】

図16は、ユーザ向けパーソナライズ製品評価を決定するアルゴリズムを表している。ユーザが製品Xに関する製品レポートを要求した時1610、このアルゴリズムは、その製品を以前に評価したことがあるユーザのプロフィールデータベース1630（製品評価を含む）から、プロフィールを検索する1620。次に、システムは、コンテンツデータベース1650からプロフィールマッチングアルゴリズムのデフォルトの閾値を検索する1640。次に、プロフィールマッチングアルゴリズムにおいて指定されたいくつかの次元に、ユーザの選抜候補者リストをすべてマップする1660。その後、最も近い近接者の上位n名（閾値変数として事前に指定）を決定し、プロフィールマッチングアルゴリズムの結果を使用して、この人達がセット内のユーザのプロフィールからの距離y（同じく閾値変数として事前に指定）の中にあるかどうかを判断するテストを実行する1670。この閾値の中にない場合は、閾値変数が緩和され1680、再度、テストが実行される。この処理は、テストで真が戻されるまで繰り返される。その後、最も近い近接者n名の小さなセットからの製品評価を使用して、いくつかの次元に沿った多数の製品統計値が決定される1690。こうした統計値は、製品レポートのテンプレートに挿入され1695、製品レポートとしてユーザに戻される1697。

【0123】

個人プロフィール及びサービスの偏在：

このシステムは、個人のプロフィールを1箇所で集中的に記憶する場所を提供する。この記憶場所は、パブリックインターネットを通じて利用可能なサーバであり、インターネットに接続された適切なアクセス能力を有する任意の装置によってアクセス可能である。至るところでプロフィールにアクセスできることから、ユーザのプロフィールに基づいてユーザ向けのサービスをカスタマイズするために、多数のアクセス装置を使用することができる。例えば、業者のWebサイトでは、このプロフィールを使用して、ユーザにパーソナライズされたコンテンツを提供することができる。インターネットにアクセス可能な携帯情報端末（PDA）では、PDA上の個人のカレンダー、電子メール、連絡先リスト、タスクリスト、及びメモを、インターネットサイトに格納されているバージョンと同期させることができる。これにより、個人は、必要な時期及び必要なフォーマットに関係なくデータを利用可能にしておくために、このデータの1つのバージョンのみを維持しておけばよい。

【0124】

図17は、こうした中央格納プロフィールにアクセスする多数の異なる方法に関連する詳細なロジックを提示している。プロフィールデータベース1710は、ユーザのプロフィール情報の中央記憶場所である。プロフィールゲートウェイサーバ1720は、ユーザ自身からのものか、或いはユーザにサービスを提供しようとしている業者からのものであるかに関係なく、プロフィール情報に関するあらゆる要求を受領する。このプロフィールゲートウェイサーバは、プロフィール所有者が明確に許可を与えた時のみ、情報を公表する状態を確保することに責任を有する。TCP/IP（標準ネットワーク通信プロトコル）でパブリックインターネット1730にアクセス可能な任意の装置は、インテリジェントHTTP要求によって、プロフィールデータベースからの情報を要求することができる。消費者は、例えば、テレビ1740、携帯電話、スマートカード、ガスメータ、水道メー

10

20

30

40

50

タ、台所用品、セキュリティシステム、デスクトップコンピュータ、ラップトップ、ポケットオーガナイザ、PDA、及び自動車等の装置からサービスへのアクセスを得ることができる。同様に、業者1750も（それぞれのプロフィールを所有する消費者からの許可を受けて）こうしたプロフィールにアクセス可能であり、これにより、カスタマイズ及びパーソナライズされたサービスを消費者に提供することができる。

【0125】

この偏在プロフィールの潜在的な使用方法の1つは、ホテルチェーンに関する。消費者は、自分を一意に特定するデジタル証明書が入ったスマートカードを携帯することができる。このスマートカードのデジタル証明書は、システムによって発行されたものであり、システムは消費者のプロフィール情報をプロフィールデータベースに記録している。消費者は、このカードをホテルチェーンに持っていき、チェックインする。ホテルの雇用者はカードを読み取り機に通し、消費者は暗証番号を入力し、デジタル証明書のロックを解除する。証明書は（安全な送信プロトコルを使用して）プロフィールゲートウェイサーバに送られ、認証される。これにより、ホテルでは、消費者が以前に指定した消費者のプロフィールの特定部分にアクセスできるようになる。ホテルでは、消費者のあらゆる請求情報及びホテルの部屋の好みを検索することができる。更にホテルでは、消費者の映画及び食事の好みにもアクセスし、この両方についてカスタマイズしたメニューを提供することができる。ホテルでは、消費者の配偶者に電子メールを送り、消費者がホテルにチェックインしており無事であることを知らせるサービスを提供できる。あらゆる取引情報は、ホテルにチェックインした後、消費者のプロフィールにアップロードすることができる。これにより、ホテルのパートナーは（同様に消費者からの許可を受けて）ホテルが収集した消費者に関する情報を利用することができる。

【0126】

意図価値ネットワーク：

意図価値ネットワークにおいて、全体的なインテグレータシステムは、ユーザへの製品及びサービスの納入を調整する。このインテグレータは、ユーザのプロフィールに反映されるユーザの好みに基づいて、ユーザに物理的及び仮想的に製品及びサービスを提供する承認サプライヤのネットワークを管理する。このインテグレータは、サプライヤと消費者との間の関係を管理し、サプライヤによる消費者の意図の充足を調整する。インテグレータは、例えば、消費者に製品及びサプライヤに関する情報を提供すること及び客観的なアドバイスを与えることによって、これを行う。

【0127】

図18では、消費者と1つのサプライヤが関与するインテグレータとの間の詳細な対話を開示している。このユーザはWebブラウザ1810にアクセスし、インテグレータに製品及び価格情報を要求する。この要求は、ユーザのWebブラウザからインテグレータのWeb/アプリケーションサーバ1820に送られる。ユーザの好み及び個人情報インテグレータの顧客プロフィールデータベース1830から取得され、Web/アプリケーションサーバに戻される。要求された製品情報はサプライヤの製品データベース1840から抽出され、特定の顧客向けにカスタマイズされる。Web/アプリケーションサーバは、顧客に関する質問情報により、サプライヤの顧客情報データベース1850を更新する。その後、製品及び価格情報はWebページ1860にフォーマットされ、顧客のWebブラウザに戻される。

【0128】

サマリエージェント：

アプリケーション及びWebサーバで実行されるソフトウェアエージェント一式は、ユーザのための反復するタスク又は日常的なタスクを管理するようにプログラムされる。このエージェントは、ユーザが設定した規則に従って動作し、ユーザによって明示的に定義されたタスクのみを実行することができる。このエージェントは、ユーザへの請求を支払うこと、コンテンツ及び電子メールをフィルタリングすること、タスク及びエージェントの活動のサマリ表示を提供することを管理できる。エージェント用のユーザインタフェース

は、特定のユーザに合わせて変更することができる。

【0129】

図19では、ユーザ向けの口頭サマリを生成するエージェントによる処理の好適な実施形態に従ったロジックを開示している。ユーザがサマリページを要求した時1900、サーバは、ユーザプロフィールデータベース1930から、エージェントのタイプ、規則、及びサマリレベルといったユーザのエージェントの好みを取得する1920。サーバは、コンテンツデータベース1950から、電子メール、予定リストの項目、ニュース、及び請求といったコンテンツを取得する1940。エージェントは、プロフィールデータベースに格納された規則を使用して、こうしたすべてのコンテンツをパースし、コンテンツのサマリを作成する1960。コンテンツはテンプレートに従ってWebページにフォーマットされる1970。コンテンツデータベース1990のコンテンツとデータベースに格納されるスピーチテンプレートとを使用して、エージェントのスピーチ用テキストが生成される1980。このスピーチテキストはWebページに挿入され1995、このページはユーザに戻される1997。

10

【0130】

信頼される第三者：

上のシナリオでは、Webサイトは、公表されたポリシーに従って、情報のプライバシーの保証を維持する必要がある。このシステムは、消費者の信頼される第三者であり、あらゆる場合において、消費者に代わって行動し、商取引の機会のシミュレーションではなく情報のプライバシーに偏り過ぎる。この信頼される第三者は、明言されたポリシーにより特定の連座を保証する適切なプロセスのセットを有する。

20

【0131】

「meコマース」：

この言葉は「eコマース」という言葉を拡張したもので、「パーソナライズ電子商取引」を意味する。図20は、好適な実施形態によるディスプレイログインを表している。このディスプレイは、エージェント2000を有するMicrosoft Internet Explorerアプリケーションとして実施され、エージェントは、様々なシステムコンポーネントをカスタマイズ及びパーソナライズし、情報を集め、ユーザの個人的要件と対話するために、システムとの対話のプロセスを通じて、ユーザをガイドする。ユーザはユーザ名を2010に、パスワードを2020に入力し、ボタン2040を選択して、ログイン手続きを開始する。ロゴ2030が示すように、このシステムは電子商取引を、パーソナライズされた、いわゆる「me」コマースに変える。

30

【0132】

図21は、好適な実施形態による日常ロジスティックス管理ディスプレイを表している。ユーザはアニメーションエージェント2100にパーソナライズメッセージ2190によって迎えらる。ユーザは、旅行2110、家庭の雑用2120、金融2130、及び市場活動2140を含め、要件に基づいて様々な活動を選択できる。ある活動から別の活動への迅速なナビゲーションを促進するために、電子メール、カレンダーリング、ドキュメント作成といったルーチンタスク用のアイコン2142も提供される。ニュースその他の関心のある項目の転送を可能にするために、ダイレクトリンク2146も提供される。ユーザがいる場所に基づいて、様々なプロフィールを選択することができる。これは例えば、仕事、自宅、又は休暇等である。このプロフィールは、ユーザが別の場所のための新しいプロフィールを必要とする時に追加することができる2170。個人情報の様々な項目2180は、様々な試みをサポートするためにユーザから収集される。更に、情報がタイムリで現行のものであることを確認するために項目2180毎に許可2150が設定される。

40

【0133】

図22は、好適な実施形態によるユーザメインディスプレイを表している。ユーザの好みに基づいて、世界のニュース2200及びローカルニュース2210が提供される。このユーザは、メインディスプレイに直接情報を提供する項目として不動産2230も選択し

50

ている。更に、ユーザの好みに基づいて、異なるエージェント 2 2 2 0 が提供される。

【0 1 3 4】

図 2 3 は、好適な実施形態によるエージェント対話を表している。エージェント 2 3 1 0 は、ユーザに情報 2 3 0 0 を伝達し、ユーザの生命保険のニーズが変化したことを示し、ユーザ向けの情報を最善の形でまとめたチャートをユーザに示している。現在のユーザ統計値に基づいて、更に詳細な情報を促すために、特定のヒント 2 3 9 5 が提供される。ユーザの生命保険のニーズに関するチャート 2 3 7 0 も、ユーザが適切な行動を判断するのを支援するために、ディスプレイの中央で強調表示される。保険証書の変更を容易にするためにボタン 2 3 8 0 が提供され、ユーザの保険要件に関する様々な表示を選択するのを支援するためにボタンのセット 2 3 9 0 が提供される。

10

【0 1 3 5】

イベントバックグラウンダ：

イベントバックグラウンダは、来るべきイベントの短い説明で、イベントの直前にユーザに送られる。イベントバックグラウンダは、このイベントに関連する最新情報によって絶えず更新される。旅程及びロジスティックスといった関連情報が含まれ、同じ場所にいる可能性のあるユーザの知人といった他の有効な情報も提供される。イベントバックグラウンダの目的は、パブリック Web サイト及びユーザのカレンダ及び連絡先リストといった多数のリソースから引き出した、イベントに関する最も新しい情報を提供し、ユーザが一定の状況において適切に反応するのを可能にすることである。

20

【0 1 3 6】

ビシニティフレンドファインダ

このソフトウェアは、友人、家族、又は知人がユーザと同じ周辺にいること、或いは同じ周辺に来ることをユーザに伝える機会を探す。このソフトウェアは、来るべきイベントについてユーザのカレンダをスキャンする。次に、地形マップを使用して、こうしたカレンダイベントを、連絡先に載っている人物のカレンダイベントと比較する。その後、ユーザに任意の一致を通知し、これにより、特定の時間にユーザの近くにいる予定である人物がいることをユーザに伝える。

【0 1 3 7】

情報過多：

情報過多という用語は、その定義と、意味及び結果との両方において、現在では比較的理解されるようになっている。人間が一度に利用可能な注意の量は限られているが、こうした注意を求める量は毎日増え続けている。簡単に言えば、多すぎる情報と少なすぎる時間こそが、今日の知識労働者の生活を複雑にする主な要因である。

30

【0 1 3 8】

情報過多に動的に対処する第一の試みでは、情報の量が減るように情報のインテリジェントフィルタリングを行うことに注目する。しかしながら、情報の小片を単純にランダムに取り除くのではなく、こうしたアプローチのほとんどでは、どの情報を最終的にユーザに提示するかについて知的になることを試みている。これは、ユーザの関心に基づいてそれぞれのドキュメントを評価し、関連性の少ないものを捨てることで達成される。そのため、結果として、質も向上する。

40

【0 1 3 9】

情報のフィルタリングは、この新しい時代において情報に対処するための第一のステップに過ぎない。おそらく間違いなく、ドキュメントの質と同じように重要なのは、それに容易にアクセスできることである。ミーティングに出席してしまえば、そのミーティングの題材について重要な情報を含んでいるドキュメントが自分のオフィスに届けられても、価値はほとんどない。相互通信の技術に刺激されビジネスの速度が高まるにつれ、どんな時でも、どんな場所においても、質の高い情報を受け取ることができる能力は不可欠になっている。この新しいアプローチは、インテリジェント情報伝達と呼ばれ、新しい情報自体の先駆けとなっている。

【0 1 4 0】

50

好適な実施形態では、情報過多を減少させるだけでなく、高品質の情報をユーザが求める場所及び時間に伝達する試みにおいて、上で説明したインテリジェント情報伝達の理論を実証している。言い換えれば、このシステムは、適切な情報を、適切な人物へ、適切な時間に、適切な場所で伝達する。

【0141】

アクティブナレッジ管理システムの説明：

図24は、好適な実施形態によるアクティブナレッジ管理システムのブロック図である。このシステムは以下の部分、つまり1つ以上のサーバとのバックエンド2400接続と、パーソナル移動無線クライアント（アウェアネスマシン）2430、2436と、パブリッククライアント（マジックウォール）2410、2420と、Webクライアント2446、2448と、電子メールクライアント2450、2460とによって構成される。

10

【0142】

バックエンドサーバ（2400）プロセス：

図25は、好適な実施形態によるバックエンドサーバのブロック図である。このバックエンド（図24の2400）は、以下のソフトウェア活動要素、つまりインテリジェントエージェントコーディネータ（Munin）2580と、情報優先度決定サブシステム2530、継続的及び断続的に実行される情報収集及び処理用インテリジェントエージェント2500、2502、及び2504と、ユーザプロフィールデータベース2542及びサポートソフトウェアと、情報チャネルデータベース2542及びサポートソフトウェアと、通信ソフトウェア2550と、情報変換ソフトウェア2560と、補助ソフトウェアとを有するコンピュータシステムである。

20

【0143】

アウェアネスマシン（図24の2446及び2448）：

●アウェアネスマシンは、ハードウェア装置とソフトウェアアプリケーションとの組み合わせである。ハードウェアはハンドヘルド型コンピュータと無線通信装置とによって構成される。アウェアネスマシンは、無線の情報細流を継続的に受領することで、絶えず更新される所有者の世界の状態を反映する。この情報は、インテリジェントエージェント一式によって掘り出され処理されるものであり、メールのメッセージ、それぞれのユーザの好みに合ったニュース、スケジュールの更新、来るべきミーティング及びイベントの背景情報、更には天気情報及び交通情報によって構成される。

30

【0144】

図25のインテリジェントエージェントコーディネータ2580も、システムに対するユーザの「インタフェース」であり、ユーザがシステムと対話するときは常に、GUIその他のエンドユーザインタフェースに関係なく、これらは最終的にはインテリジェントエージェントコーディネータとやり取りをする（質問を行い、コマンドを送る）。インテリジェントエージェントコーディネータには主に4つの責任があり、それは1）ユーザの活動を監視すること、2）情報要求を処理すること、3）それぞれのユーザのプロフィールを維持すること、及び4）ユーザ及びそれぞれのエージェントとの間でやり取りする情報の経路を定めることである。

40

【0145】

ユーザの活動の監視：

ユーザがセンサを始動させた時は常に、インテリジェントエージェントコーディネータは「環境キュー」を受領する。このキューは、インテリジェントエージェントコーディネータが、情報伝達の目的で、ユーザがいる場所の理解を得ることを可能にするだけでなく、それぞれの人物の生活の標準パターン（到着時間、出発時間等）を学習することも可能にする。こうしたパターンは、情報を伝達する時のシステムの知性を高める試みにおいて、絶えず更新され改良される。例えば、現在、1人の人物がいくつかの電子メールアカウント（仕事ベース、自宅ベース、移動ベース等）を有することは珍しくなく、こうしたすべてのアカウントに関する検索プロセスにいくつかの異なるコンピュータが関与することも珍しくない。したがって、インテリジェントエージェントコーディネータが正しい場所へ

50

の情報伝達を成功させるには、ユーザが情報を見る確率を最大にするために、こうしたすべてのアカウントと、ユーザがこれらにアクセスする可能性のある時間とを考慮する必要がある。これについては、別のセクションで更に説明する。

【0146】

情報要求の処理：

インテリジェントエージェントコーディネータは、それぞれのユーザを意図して情報をパーソナライズし、ユーザが受け取る情報において、それぞれのユーザの関心を更に正確に反映させるために、他のエージェントからの情報要求を処理する。こうした要求は一般に、ユーザのプロフィールに関連している。例えば、エージェントは、ユーザのために交通レポートを作成している時、インテリジェントエージェントコーディネータに、そのユーザの交通地域（サーチストリング）を要求する可能性がある。ユーザのプロフィールデータに対するすべてのアクセスは、この方法においてアクセスされる。

10

【0147】

ユーザプロフィールの維持：

ユーザプロフィールにはユーザに関する広範な情報が含まれる。この情報は、ユーザ指定データと、インテリジェントエージェントコーディネータがそれぞれのユーザの情報及び活動から学習及び推定した情報との混合物である。プロフィールに含まれるデータを保護するために、インテリジェントエージェントコーディネータはすべての情報要求を処理する必要がある。インテリジェントエージェントコーディネータは、更に多くのルーチンである日常的タスクを支援するために、ユーザの活動を観察し、その生活パターンの学習を試みることにより、こうしたプロフィールを絶えず修正及び更新する。インテリジェントエージェントコーディネータは更に、それぞれのユーザの毎日の活動から意味を拾い集めるために、他のエージェントを利用する。これらのエージェントは、こうしたデータを掘り出し、現在の関心事、長期的な関心事、更にはそれぞれのタイプの情報を伝達する時間の好みを示すものを見つけようとする。インテリジェントエージェントコーディネータの観察に関する別の重要な態様は、経路指定の目的で、それぞれのユーザが1日を通じて物理的にどこに位置するのかを判断しようとすることである。

20

【0148】

情報の経路指定：

ほとんどの人間は1日を通じて移動する。インテリジェントエージェントコーディネータは、観察（監視されない学習）と環境からのキューとの両方により、ユーザがどこに位置するか、或いはどこに位置する可能性が高いかを判断しようとする。この事実に対して敏感になる。これはユーザの情報をどこに送るかを判断するためだけでなく、どのフォーマットで情報を送るかを判断するためにも間違いなく重要である。例えば、ユーザが机に向かっており、Webクライアントを使用しているばあい、インテリジェントエージェントコーディネータはユーザのPCから活動を示すものを受領し、任意の必要な情報をそこへ送るべきであることが分かる。加えて、デスクトップPCは通常非常に強力であるため、機能をすべて使った画像の多いバージョンを送ることができる。しかしながら、これに代わる状況、つまりインテリジェントエージェントコーディネータが（出口の近くにあるキーカードリーダーにより）あなたがビルを出たばかりであることを示すものを受領した状況も考えられる。その数分後、インテリジェントエージェントコーディネータが、あなたが宛の緊急メッセージを受領したとの通知を受領したとする。インテリジェントエージェントコーディネータは、あなたがビルを出たことを知っており、それ以外の行動を示すものを受領していないため、ハンドヘルド型装置（その能力についても知っている）によって連絡できるとみなし、画像優先バージョンではなく、緊急メッセージのテキストをそこに送る。

30

40

【0149】

内在するイノベーション：

アクティブナレッジ管理システムは、ナレッジ管理と人間及びコンピュータの対話との世界において、いくつかの最も高度な思考を意味している。主なイノベーションの一部には

50

以下が含まれる。

- 上で例示したインテリジェントエージェントコーディネータ。
- インテリジェント情報伝達の理論の開発、実証、及び実現。
- すべてが共通のバックエンドを利用した情報伝達のいくつかのチャンネルのサポート。例えば、ユーザがマジックウォールの前にいる場合、情報はマルチメディア対応形式で提示される。ユーザは移動中であるとシステムが判断した場合、情報は標準のテキストによってアウェアネスマシンに送られる。これにより、ユーザが情報を求める時間及び場所に関係なく情報を伝達することを促進する。
- 静的なユーザプロフィールに基づくだけでなく、ユーザの対話履歴と、「人、場所、及び時間」の認識を含む現在のリアルタイム状況とを考慮することによる情報のパーソナライズ。
- インテリジェントエージェントコーディネータの意見、ユーザの好み、及びユーザの対話履歴を考慮する高速でスケーラブルな情報優先度決定サブシステムの利用。インテリジェントエージェントの部分から日常的判断の負荷を取り去ることにより、システムのスケーラビリティを損なうことなく、エージェントを更に精巧で精度の高いものにできる。
- インテリジェントエージェントのアニメーション表示及び触覚入力と組み合わせた音声認識及び音声合成により、効率的で、直観的で、感情的に満足感を与えるシステムとの対話を提供する。

10

【0150】

クライアント報告サブシステムモデル：

20

コンテキスト：

報告サブシステムは、レポート（読み出し、記録事項の作成）を行うためにクライアント上の他のサブシステムによって使用される。このサブシステムは、取り扱うデータのタイプを想定しない—このデータは、（アーキテクチャサービスの一環としての）誤ったレポート又は（アプリケーションデータサービスの一環としての）リード管理情報である可能性がある。報告サブシステムは、この意味において、インフラの一部であり、クライアント上のあらゆる場所で利用可能な一連の基本サービスである。報告サブシステムは、通信サブシステムを使用して、データの格納及び転送を行う。

【0151】

アーキテクチャの概要：

30

報告サブシステムはあらゆるクライアントサブシステムにサービスを提供する。このサブシステムは、クライアントアプリケーション内と、クライアント及びサーバ間とにおいてメッセージを伝えるメカニズムを備える。この報告メカニズムは、通信サブシステムを使用して、データの格納及び転送を行う。例外は、この報告メカニズムを使用して、エラーに関する情報のみをレポートする。

【0152】

役割：

報告メカニズムは、インフラサービスのセットを提供し、これによりアーキテクチャコンポーネントは、情報をレポートすることができる。このサブシステムは、レポートされる情報について規定しないが、システム障害又は顧客対話情報といった特定のタイプのレポート情報へのマッピングを行うコンポーネントを含む。

40

【0153】

このサブシステムのコンポーネントの一部は、例外のセットとして提示され、これにより、処理中に遭遇したエラー条件の自動レポートが可能になる。

【0154】

このサブシステムはデータを受け入れ、適切なフォーマットで通信サブシステムに転送する。このサブシステムは、エラー条件の結果である処理中に生成された例外を捕捉しレポートする。このサブシステムは、エラーロギングから販売のリードに至るまで、作成する必要のある任意のタイプのレポートに対処することが可能である。必要に応じて新しいタイプの情報を記録するのに十分な柔軟性を有する。更に、必要に応じて新しいタイプのレ

50

ポートを追加できる十分な柔軟性を有する。加えて、ロギングプロセス中に特定の情報が利用できない状態に対処することができる。

【0155】

責任：

クライアント報告サブシステムは、アプリケーションのあらゆる部分で使用する例外のセットを提供することに責任を有する。このサブシステムは更に、ロギング失敗レポートと、アプリケーションハートビートレポートと、メッセージ受領証と、紹介又はリードと、管理情報システムのエントリとに責任を有する。

【0156】

免責：

このサブシステムは、インタフェースでの対話その他の場所から情報を収集することに責任を有しておらず、どのデータセットをレポートする必要があるかを判断することにも責任を有していない。レポートすることには、レポートを印刷することは含まれない。

【0157】

コンポーネントの仕様：

クライアント例外報告コンポーネント

これは例外クラスのセットであり、クライアント報告コンポーネントの報告サービスを使用して、障害情報を格納及び送信する。

【0158】

クライアント報告コンポーネント

これはあらゆるレポートの情報を収集し、フォーマットし、クライアント通信サブシステムに送信するメカニズムである。

【0159】

作成、存続、及び管理：

このサブシステムの重要な要素は、レポートオブジェクトの作成を管理する静的なクラスレポートファクトリである。このクラスは、通信サブシステム（クライアントの構成を管理）によってインスタンスを作成され、常に利用可能である。そうではない場合には深刻なエラーとなる。レポートは、必要に応じて、その場限りで生成される。

【0160】

サイジング及び容量：

クライアントアーキテクチャの要件がどんなものであっても、クライアント報告サブシステムの「スループット」（一定の時間に要求されるレポート数、タイプの合計数として理解される）は、システムリソースに大きな負担をかけない。レポートに関する容量要件のほとんどは、こうしたレポートの格納及び伝送用に準備する必要がある通信サブシステムによって吸収される。

【0161】

それでもなお、このサブシステムはアーキテクチャによって要求されるあらゆるスループットに対処できる必要があり、その設計では、アーキテクチャのそれぞれの部分で生成される推定仕事量を考慮に入れる。

【0162】

性能：

クライアント報告サブシステムでは性能は重要ではない。レポートデータは、僅かな例外を除き、伝送前に格納されるため、データが送信される前の遅延は予期される。特定のタイプの深刻又はクリティカルな障害は一度に報告する必要があるが、こうした伝送に必要な低い帯域幅は性能の問題を提示しない。

【0163】

設計のガイドライン：

アーキテクチャのレポートのニーズは変動するが、能力の中核セット（障害レポート、リード管理、対話レポート）は常に要件として残る。このサブシステムは、能力の拡張又は削減だけでなく、それぞれの能力のために記録するデータの詳細度及び性質の調整を可能

10

20

30

40

50

とする柔軟性を有する、
このシステムの実施はプロジェクト J a v a コード化規格に従う。

【0164】

論理コンポーネント：

クライアント例外報告コンポーネント内の例外はクライアント報告コンポーネントを呼び出し、障害レポートを作成する。

【0165】

コンポーネントの説明：

クライアント例外コンポーネント

これは例外クラスの階層であり、例外の処理と受け渡しを支援するために構築される。これらの例外は、それが表す例外イベントに関する情報を受け入れる。こうした情報と、このクラスが自分のイベントについて知っている他のすべての事柄とによって、このコンポーネントはクライアント報告コンポーネントを使用して障害レポートを作成する。

10

【0166】

クライアント報告コンポーネント

静的なレポートファクトリは、合図されたイベントの形式で、他のコンポーネントからの要求を受け入れる。このイベントに基づいて、このファクトリは特定のタイプのレポートを作成する。このレポートは、その後、呼び出しコンポーネント又はレポートによって供給された情報に占有される。

【0167】

レポートの発信：

コンポーネント A は、クライアント報告サブシステムに対して合図を送り、レポートすべきイベントが発生したことを示す。次に、クライアント報告サブシステムは、イベントに関する情報を要求し、レポートを作成する。最後に、クライアント報告サブシステムは、クライアント通信サブシステムに、レポートを送る準備ができたことを合図する。

20

【0168】

例外の投げ込み：

第一に、コンポーネント A は、適切な例外のインスタンスを作成することにより例外的なイベントが発生したことを合図する。第二に、コンポーネント A は、レポートすべきイベントの関連情報を例外に渡す。第三に、コンポーネント A は、例外を投げ込むことを要求する。第四に、クライアント例外報告コンポーネントは、上で概要を説明したクライアント報告コンポーネントを使用して、利用可能な情報に基づき、送信されるレポートを発信する。最後に、クライアント例外報告コンポーネントは例外を投げ込む。

30

【0169】

ローカルコンテンツサブシステムモデル：

コンテキスト：

ローカルコンテンツサブシステムは、アプリケーションによって求められるあらゆるコンテンツを提供する。これには静的コンテンツ及び動的コンテンツの両方が含まれる。更に、アプリケーションが求めるビジネスサービスを提供する。「格納及び検索」に基づいて動作し、ユーザ又は企業から取得したデータを格納し、データを検索するメカニズムを提供する。ローカルコンテンツサブシステムは、コンテンツを提供するために I S F サブシステムによって使用される。通信サブシステムを使用して、ビジネスデータを受領する。

40

【0170】

アーキテクチャの概要：

ローカルコンテンツサブシステムのオブジェクトは、アプリケーションアーキテクチャの初期化サブシステムによって作成される。ローカルコンテンツサブシステムによって提供されるサービスは更に、アプリケーションアーキテクチャを通じて、初期化サブシステム及び I S F サブシステムを介してアクセスされる。

【0171】

ローカルコンテンツサブシステムのコンテンツの例は、アプリケーションによって求めら

50

れる様々なタイプのビジネス知識及びプロセスを反映する。ユーザデータ及びビジネスデータには、それぞれユーザ及び企業から収集したデータが関与する。計算コンポーネントは複雑な計算を実行し、製品コンポーネントは企業が使用する製品を表す。コンテンツプロバイダコンポーネントは静的媒体コンテンツを定義する。

【0172】

役割：

ローカルコンテンツサブシステムは、アプリケーションに静的及び動的コンテンツを提供する。更に、アプリケーションによって求められる、あらゆるビジネス固有サービスを提供する。

【0173】

責任：

ローカルコンテンツサブシステムは、アプリケーションに静的及び動的媒体を提供する。ローカルコンテンツサブシステムは更に、ユーザが入力した詳細、及びビジネスデータを格納し、ビジネスに関する計算を実行する。

【0174】

免責：

すべてのアプリケーション固有動作は、アプリケーションパーソナリティを通じて提供される。この動作は、ハムレットスクリプトによって定義され、これはスクリプト間のナビゲーションも定義する。このスクリプトはメタファに区分され、そのそれぞれは対話のスタイルの実施形態である。

【0175】

媒体へのアクセスは、このサブシステムに属するコンテンツプロバイダコンポーネントを通じて提供される。しかしながら、このコンポーネントのオブジェクトは、アプリケーションパーソナリティで定義されたコンテンツファイルから、システムイニシャライザコンポーネントによって自動的に作成される。

【0176】

作成、存続、及び管理：

ビジネスデータコンポーネントは、システム初期化の時期に作成及び初期化される。これはシステムの寿命に渡って存続する。

【0177】

ユーザデータコンポーネントは、顧客がシステムを使用している時間全体に渡って利用できる。顧客のセッションが終了した時、ユーザデータコンポーネント内のすべてのオブジェクトに対するリファレンスは、オブジェクトをガーベジコレクションできるようにリリースされる。

【0178】

ローカルコンテンツサブシステムのオブジェクトは（ビジネスデータコンポーネントを除き）、内部的に、或いは初期化サブシステムによって、作成及び初期化される。こうしたオブジェクトに対するリファレンスは、ISFサブシステムによって管理される。ビジネスデータオブジェクトは、通信サブシステムによって作成、初期化、及び管理される。

【0179】

論理コンポーネント：

ISFは、実際にはローカルコンテンツサブシステムに関する知識を持たない。ローカルコンテンツサブシステムは、ISFによって定義されたインタフェースのセットを実施する。初期化サブシステムは、アプリケーションパーソナリティで定義されたスクリプトを使用して、コンテンツを検索するために、（こうしたインタフェースを実施する）どのオブジェクトを使用する必要があるかを定義する。ISFは初期化コンポーネントを使用して、こうしたオブジェクトを作成し、その後、これらを管理する。

【0180】

アプリケーションパーソナリティの開発者は、ローカルコンテンツオブジェクトを直接管理することとして、自分のスクリプトを容易にとらえることができる。これにより、ロー

10

20

30

40

50

カルコンテンツオブジェクトは、アプリケーションアーキテクチャレイヤの知識を持たずに開発することができる。

【0181】

コンポーネントの説明：

ユーザデータ

ユーザによって入力されたデータを格納及び検索し、格納されたデータに関する計算を開始する。

ビジネスデータ

企業によって提供されたデータを格納及び検索する。

計算

複雑な計算を実行する。

製品

特定のマーケティング製品に関連する情報へのアクセスを提供する。

コンテンツプロバイダ

静的媒体へのアクセスを提供する。これはインタフェースの実施であり、別個のコンポーネントとしてドキュメントされない。

【0182】

インタフェースサポートフレームワークサブシステムモデル：

コンテキスト：

インタフェースサポートフレームワークサブシステムは、アプリケーションアーキテクチャレイヤの一部である。動的なマルチメディアインタフェースのあらゆる可能性を活用する豊かなインタラクティブ環境を提供する。このISFは、すべてのオブジェクトが動的な動作を行うことが期待される劇場のメタファを中心に構築される。

【0183】

ISF内のオブジェクトは、アプリケーションアーキテクチャレイヤ内のアプリケーション初期化サブシステムによって初期化され、テクニカルアーキテクチャレイヤのコンテンツプレイヤ、印刷及び報告サブシステムを利用する。

【0184】

アーキテクチャの概要：

ISF内のオブジェクトはアプリケーション初期化サブシステムによって初期化される。報告及び取引インタフェースサービスは、テクニカルアーキテクチャレイヤのためにISFデータをログし、レポート又は印刷するために使用される。テクニカルアーキテクチャ内のコンテンツプレイヤサブシステムは、ユーザに媒体を提示するために、ISFによって使用される。

【0185】

ISFサブシステムは、モデル・ビュー・コントローラのパターンに従った階層化アーキテクチャの上に構築される。事実コンポーネントは、ビジネスのオブジェクトモデルと、ビジネス媒体コンテンツの定義とを含む。視覚コンポーネントは、ユーザにビジネスモデルのビューを提供するために媒体を表示及び操作する。動作コンポーネントは、視覚コンポーネントと事実コンポーネントとの間のあらゆる対話を制御する。

【0186】

役割：

ISFサブシステムは、マルチメディアコンテンツを制御された形で提示するために、アプリケーションにサービスを提供する。更に、ユーザの入力に反応し、シーンの変化に影響を与える能力を提供する。

【0187】

責任：

ISFサブシステムは、アプリケーションのそれぞれのシーンを表示し、表示されている間にシーンのコンテンツを修正する。加えて、ISFサブシステムは、シーン間のナビゲーションを可能にし、ユーザ対話に反応し、ビジネスコンテンツを検索し、ビジネス関数

10

20

30

40

50

及び計算を実行し、共通ユーザインタフェース構成子を提供する。I S Fのその他の責任には、印刷ジョブ及びビデオ会議セッションを開始することと、ユーザのエントリをシーンにレポートすることと、セッションの持続時間と、役割とのユーザ対話と、ユーザのナビゲーションと、I S F内で発生したエラーをレポートすることとが含まれる。最後に、I S Fはユーザセッションを管理し、システムの起動及びシャットダウンといったシステムレベルのイベント又はスクリーンペイント要求に反応する。

【0188】

設計のガイドライン：

I S Fサブシステムは、マルチメディアコンテンツを提示するためにアプリケーションにサービスを提供する。アーキテクチャ的にはモデル・ビュー・コントローラパラダイムに対応する3つの別個のコンポーネントに階層化される。これにより、コアビジネスオブジェクト及びそのデータ（モデル）は、この情報の視覚表現（ビュー）から分離され、モデル又はビューを制御し、これらの変化に反応するロジック（コントローラ）から分離される。このアーキテクチャは、システムのグラフィック的スタイル（ステージ、役割、及びシーン）と、動作コード（アクタ、シーンディレクタ、及びステージマネージャ）と、基盤となるコンテンツプロバイダ（ビジネスオブジェクト）との間の境界を提供する。これらのセクションは、視覚、動作、及び事実コンポーネントとなる。

10

【0189】

事実レイヤは視覚レイヤを認識しない。これにより、基盤となるビジネス領域モデルを混乱させることなく、視覚メタファは変化することができる。動作レベルは、事実レイヤと視覚レイヤとの間を仲介し、いずれかのレイヤとの非常に複雑な対話は避けるべきである。可能な場合は、レイヤ間の更なる相互依存性を回避するために、パブリッシュ／サブスクライブパターンによる匿名通信を使用がされる。

20

【0190】

ステージはディスプレイコンテキストとして特定される。これは制御するロケーションとのみ通信することができる。これはステージマネージャの影に隠れており、ステージマネージャではあらゆる視覚要求が管理される必要がある。

【0191】

I S Fは階層システムである。シーン内のすべての役割は一連の視覚シブリングを形成する。これらの役割は、実際には、他の役割を含みカプセル化する。これにより、再帰を通じて、任意の数の別個の処理レイヤが可能となる。それぞれの子は直接の親とのみ通信し、その親を超えた通信の制御を放棄する。この封じ込め関係は、視覚レイヤ及び動作レイヤの両方において可能である。

30

【0192】

ナビゲーションを支援するために、シーンサムネイルが維持される。ユーザはシーンサムネイルに触れることで、以前訪れたシーンに戻ることができる。

【0193】

コンポーネントの説明：

視覚コンポーネント

ユーザ対話及びマルチメディアコンテンツの提示。

40

動作コンポーネント

アプリケーション動作及びマルチメディアコンテンツの検索。

事実コンポーネント

マルチメディアコンテンツとビジネス関数サービス及び計算を提供する。

【0194】

プロセス制御：

この表ではI S F内で実行される様々なキースレッドについて説明している。

【0195】

【表6】

スレッド	目的
AWT	ウィンドウメッセージ(スクリーンタッチ等)をプロセスコントローラに送る。このスレッドはJava仮想マシンによって作成される。
メイン	アプリケーションを初期化し、その後、終了する。このスレッドはJava仮想マシンによって作成される。
プロセッシング	プロセスコントローラによって開始された行動を実行する。
タイマ	セッションのタイムアウト等、時間ベースのシステムイベントを生成及び実行する。
ビデオステータス	ビデオ終了イベントの通知を受領し、これをプロセスコントローラにディスパッチする。
オーディオステータス	オーディオ終了イベントの通知を受領し、これをプロセスコントローラにディスパッチする。

10

【0196】

ユーザがステージ上のロケーションにタッチ：

20

説明：

エンドユーザが、アプリケーションの可視ウィンドウ、ステージにタッチする。これにより、アプリケーションからの応答が開始される。

アクタ：エンドユーザ

関与するコンポーネント：視覚

関与するキーオブジェクト：ステージ

【0197】

ステージがユーザタッチを処理：

説明：

アプリケーションウィンドウは、タッチされたエリアによってどの位置が影響されるかを判断し、対応するタッチの役割に通知する。このステージは、ウィンドウ上に表示される視覚キューも制御する。

30

アクタ：エンドユーザ

関与するコンポーネント：視覚

関与するキーオブジェクト：タイマ、ユーザ対話レポータ、ロケーション、メディアプレイヤー

【0198】

役割がユーザタッチイベントを受け入れ：

説明：

それぞれの役割がユーザタッチを通知される。これにより、役割は、対応する媒体での媒体の変更を要求することを強制され、完了後、アクタにユーザ対話を通知する。

40

アクタ：ステージ

関与するコンポーネント：ビジュアル、動作

関与するキーオブジェクト：アクタ、役割

【0199】

アクタがイベントキャストリングをアクティブに：

説明：

アクタはすべての登録済みキャストリングリストを循環し、特定のイベントに関心を有するすべてのキャストリングをアクティブにする。キャストリングは多形的に動作し、そのため、どのように反応するかは動作は、アクタではなくキャストリングに保持される。

50

アクタ：役割、実行キャスティング

関与するコンポーネント：動作

関与するキーオブジェクト：アクタ、キャスティング、ステージマネージャ、シーンディレクタ

【0200】

ステージマネージャがシーンの移行を実行：

説明：

ステージマネージャは、ナビゲーションキャスティングの情報に基づき、現在アクティブであるシーンを新しいシーンに置き換える。どのように変化が発生するかを制御し、キオスクワールドの視覚的イリュージョンを持続することが重要である。

10

【0201】

このシナリオは、アプリケーションの起動、又は再起動時にも呼び出される。この場合、現在のシーンは存在しないが、アプリケーションはアプリケーションの第一のシーンへ移行するように命令される。

【0202】

アクタ：ナビゲーションキャスティング、システムイニシャライザ

関与するコンポーネント：動作、視覚

関与するキーオブジェクト：ステージ、シーンディレクタ、セッションマネージャ

【0203】

シーンディレクタがキャスト変更を実行：

20

説明：

シーンディレクタは、すべてのタイミングのアクティブ化を調整し、ステージマネージャへの任意のメッセージが共にグループ化され、適切な時期に連動するようにする。これにより、シーン上で見ることができる役割のあらゆる変化が同時に起こるようになる。

【0204】

アクタ：コンテンツキャスティング、スライドキャスティング

関与するコンポーネント：視覚、動作

関与するキーオブジェクト：シーンディレクタ、ステージマネージャ

【0205】

シーンディレクタがスライド効果によりキャスト変更を実行：

30

説明：

シーンディレクタは、スライドしないすべての役割をステージに上げ、その後、スライドキャスティングのコンテンツを、直線に沿って、最終的な目的地に表示する必要がある。

【0206】

アクタ：スライドキャスティング

関与するコンポーネント：視覚、動作

【0207】

ビジネスオブジェクトによるビジネス関数の呼び出し：

説明：

アプリケーションは、ビジネス領域モデルからのビジネス情報を要求することができる（例えば、支払額を戻す）。このインタフェースは、ビジネスオブジェクトに値を入れることもサポートする（例えば、ローン期間の金額として格納する）。それぞれのビジネスオブジェクトには、動作を呼び出すために一般的なインタフェースが提供される。望ましい動作は、ビジネス関数キャスティングにおいて指定され、これはビジネスオブジェクトに関連するアクタに情報を戻すことができる。

40

【0208】

アクタ：ビジネス関数キャスティング

関与するコンポーネント：事実

【0209】

ステージマネージャがインアクティブ状態からタイムアウト：

50

アプリケーションは、アトラクタシーンに戻る前にユーザがアプリケーションから立ち去った場合に備え、タイムアウト機能をサポートする必要がある。これにより、ユーザが入力した詳細のプライバシーが守られる。

アクタ：ウェイトタイマ

関与するコンポーネント：動作

【0210】

セッションマネージャがアプリケーションをリセット：

説明：

アプリケーションの実行中、ステージを第一のシーンにリセットし、セッションをクリアしなければならない可能性がある。これはシステムのインアクティブ状態によって、或いはビジネスオブジェクトを通じてディスパッチされる直接的なユーザの要求によってトリガすることができる。

10

【0211】

アクタ：ステージマネージャのウェイトタイマ、ビジネスオブジェクト

関与するコンポーネント：動作

【0212】

セッション情報をリセット：

それぞれのセッションは、ユーザについて収集した情報を格納する。セッションの終わりにおいて、或いはユーザの要求により、すべての入力データを消去することができる。これはプライバシーをサポートする。

20

【0213】

アクタ：セッションマネージャ

関与するコンポーネント：動作

【0214】

反応媒体が媒体を表示：

説明：

ISFの視覚コンポーネントは、あらゆる媒体（画像、音声、動画、テキスト）をステージに表示することに責任を有する。これは、クライアントテクニカルアーキテクチャ内の基盤となる媒体サブシステムと実際にインタフェースする。それぞれのプレイヤは、ゲー

30

【0215】

アクタ：役割

関与するコンポーネント：視覚、動作、事実、コンテンツプレイヤ

関与するキーオブジェクト：反応媒体、媒体プレイヤ、ステージ

【0216】

アプリケーションがハードコピーの印刷を要求：

説明：

クライアントアプリケーションは、静的情報（チェックリスト）又は動的情報（現在の利率と、製品、製品シミュレーション、及び又は線グラフのその他の動的コンポーネントとを含む製品説明）の印刷を要求できる。これは、エンドユーザによって、印刷キャスティングを通じて開始される。

40

【0217】

アクタ：印刷キャスティング

関与するコンポーネント：動作、印刷

関与するキーオブジェクト：ビジネスオブジェクト

報告インタフェースサブシステムモデル

【0218】

コンテキスト：

報告インタフェースサブシステムは、アプリケーションアーキテクチャレイヤによってロ

50

グされた情報を収集し、テクニカルアーキテクチャレイヤのクライアント報告サブシステムに送る。

【0219】

アーキテクチャの概要：

報告インタフェース情報は、アプリケーションアーキテクチャレイヤ内のコンポーネントによってログされ、テクニカルアーキテクチャレイヤのクライアント報告サブシステムに送られる。

【0220】

役割：

報告インタフェースサブシステムは、キオスクとのユーザ対話をログするサービスを提供し、アプリケーションアーキテクチャレイヤ内で発生したソフトウェア及びハードウェア障害をレポートする。

10

【0221】

責任：

報告インタフェースサブシステムは、ユーザがシステムに行ったこと、例えば、ユーザがどのシーンを訪れているか、どの視覚要素と対話しているかを捕らえることで、キオスクとのユーザ対話を収集及びログすることに責任を有する。報告インタフェースサブシステムは、顧客が関心を持っているビジネス製品と、顧客が呼び出したビジネス関数の出力と、顧客が入力したビジネスデータとに関する情報も収集及びログする。最後に、報告インタフェースサブシステムは、キオスクのソフトウェア及びハードウェアの性能に関する情報を捕らえる。この情報は、その後、エラー処理及び障害管理分析のために使用することができる。

20

【0222】

免責：

報告インタフェースサブシステムには、顧客の名前及び電話番号といった顧客関連情報を収集及びログするサービスは含まれない。

【0223】

システム管理サブシステムモデル：

コンテキスト：

システム管理には、自動及び手動手続きの組み合わせの定義が関与する。自動化は、主に System Management Server (SMS) の使用を通じて達成される。SMS は、Microsoft Backoffice ツールスイートのツールであり、分散環境でシステムソフトウェア及びハードウェアを集中的に管理することができる。

30

【0224】

システム管理サブシステムモデルのアーキテクチャ：

システム管理サブシステムのアーキテクチャは、2つのコンポーネント、System Management Server (SMS)、及び障害監視によって構成される。

【0225】

System Management Server (SMS)：

System Management Server (SMS) はマイクロソフトのツールであり、ソフトウェア／コンテンツの配布、ソフトウェア監査の実施、障害診断の実行、及びリモート制御の実施に使用することができる。SMS は、このアーキテクチャ用に開発されたコンポーネント、つまりファイル転送ユーティリティによって補足される。

40

【0226】

障害監視：

キオスクは、ハートビートメッセージシステムを通じてリアルタイムで監視される。ハートビートパルスは、設定された割合（例えば1分毎）でキオスクから送られ、キオスク監視アプリケーションを実行するコンソールで監視される。キオスクのステータスが変化して障害を示した場合、監視アプリケーションは適切な行動を開始する。一部のエラーは、既存の運用センタによって処理される。こうしたエラーの経路指定は、アプリケーション

50

サービスサブシステムによってカバーされる。

【0227】

役割：

SMSは、中規模のソフトウェア配布（コード及びコンテンツの両方）及びリモートキオスクの障害診断に使用される。SMSを柔軟にする多数の特徴と、有効なサポート機能とが存在する。障害診断は、それぞれのキオスクと関連する周辺機器とのリアルタイムのステータスを見る手段を提供する。特定のキオスクで問題（用紙切れ等）が発生した時、そのキオスクに運用担当者の注意が向けられる。問題の解決方法を検証するためにキオスクのステータスを観察することができる。

【0228】

責任：

SMSは、ソフトウェア配布と、ハードウェア障害管理及び診断と、クライアントのリモートリブートとに責任を有する。SMSは、ネットワーク内のあらゆるキオスクのステータスを選択的に見るためのユーザインタフェースも提供する。

【0229】

作成、存続、及び管理：

SMSは、好適な実施形態による専用サーバに常駐し、常に利用可能である。

【0230】

性能：

MMT上での障害の発生と、その後の運用端末での障害の表示との間の経過時間は、ハートビートの頻度、アプリケーションサーバがハートビートを処理する能力、及び運用端末でのリフレッシュの頻度に応じて変化する。

【0231】

好適な実施形態による例を以下に提示する。

【0232】

【表7】

	経過時間(分)
ハートビートメッセージが送られた直後に障害が発生する。	1:00
ハートビートメッセージがアプリケーションサーバに送られる。	0:01
アプリケーションサーバがハートビートメッセージを受領及び処理する。	0:01
運用端末の障害監視アプリケーションがリフレッシュされたばかりで、2分に一度のみリフレッシュされる。	2:00
障害監視アプリケーションがキオスクステータスのメインウィンドウをリフレッシュする。	0:05
障害監視アプリケーションがキオスクステータスビューをリフレッシュする(この時間は開いている1ウィンドウのみに関するものである。1つ以上のウィンドウが開いている場合、この時間に開いているビューウィンドウの数をかけるべきである)。	0:05
合計	3:11

【0233】

ハートビートメッセージが5分以内にキオスクから受領されない場合（設定可能）、障害監視はキオスクのステータスを未知に設定する。このタイムラグは、実際の問題がハートビートメッセージの通常の処理時間よりも僅かに遅れて存在する時に、MMTマシンを誤

って未知としてレポートするのを避けるために必要である。

【0234】

論理コンポーネント：

SMSサイトは、2つのコンポーネント—一次サイト及びクライアントによって構成される。一次サイトはSMS階層の最上レベルである。このサイトは、このサイト自体と下にある二次サイトとに関するシステム及びインベントリ情報を格納するための独自のSQLデータベースを含んでいる。クライアント（キオスク）は、一次サイトから管理される。クライアントは、SMSインベントリサービスを通じて、SMSサーバにハードウェア／ソフトウェア情報を送る。

【0235】

10

障害監視：

設定可能な間隔の後、クライアントはマシン、プリンタ、及びアプリケーションのステータスチェックを行い、結果をハートビートメッセージによりサーバに送る。次にサーバは、運用スタッフによって障害の監視が行われるキオスクステータスデータベースにこのステータスを入れる。

【0236】

サーバ通信サブシステムモデル：

コンテキスト：

サーバ通信サブシステムモデルは、サーバテクニカルアーキテクチャの一部である。

【0237】

20

サーバ通信サブシステムは、クライアントと、中央サーバと、メインフレームホストとの間のあらゆる通信を処理する。

【0238】

アーキテクチャの概要：

非同期メッセージングコンポーネントは、標準インターネットメールプロトコル、SMTP、及びPOP3を使用して、クライアントとサーバとの間の非同期メッセージベースの通信を提供する。

【0239】

ビジネスプロセスアクセスモジュールコンポーネントは、MMTからの対話情報を記録するといった事前に定められたビジネス機能呼び出すための共通ポイントを提供する。HTTPは、ワールドワイドウェブ上のHTTPサーバと通信するために使用されるプロトコルである。これは、起動時のクライアント設定プロセス中、アプリケーションコンポーネント及びコンテンツの小さな更新を配布するためにMMTで使用される。メインフレーム常駐データベースへのアクセスは、必要なデータベースの表を対応するサーバ常駐データベースの表に複製することで行われる。逆のプロセスは、サーバで蓄積されたデータをメインフレームデータベースの表で集中的に格納するために使用される。サーバからの一般的な警告は、メインフレームのフロントエンドプロセッサとのインタフェースを通じて、メインフレームへ伝送される。

30

【0240】

役割：

40

サーバ通信サブシステムの役割は、サーバに、MMT（又はインターネット）クライアントとネットワークサーバとの間、及びネットワークサーバとメインフレームシステムとの間の通信機能を提供することである。加えて、サーバ通信サブシステムは、組織固有機能へのアクセスを分離及び提供する。

【0241】

責任：

サーバ通信サブシステムは、標準インターネットプロトコルに従い、その伝達チャネルへの容易な移植を可能とする。加えて、サーバ通信サブシステムは、クライアントとアプリケーションサーバとの間の信頼性の高い非同期通信、及び組織固有機能への制御された信頼性の高いアクセスを提供する。最後に、サーバ通信サブシステムは、アプリケーション

50

コンポーネント及びコンテンツ等のクライアントの設定に対する更新を伝達する機能を提供し、データベースの複製及び一般的な警告のレポート等の事前に定められたプロセスを通じて組織のレガシシステムへのアクセスを提供する。

【0242】

免責：

サーバ通信は、A C Tメッセージングアーキテクチャに準拠するモジュールを呼び出すことに限定される。別のプラットフォームとの通信が必要な場合、これは組織のメッセージング又はアクセス方法を使用する外部システムモジュールの中に位置している必要がある。

【0243】

取引インタフェースサブシステムモデル：

アーキテクチャの概要：

取引は、アプリケーションアーキテクチャレイヤ内のコンポーネントによって開始され、テクニカルアーキテクチャレイヤ内のクライアント報告サブシステムに送られる。

【0244】

このドキュメントにおいて特定される取引サービスは、それ自体では独立したコンポーネントとして実施されないが、既存のアプリケーションアーキテクチャコンポーネントの拡張として実施される。

【0245】

役割：

取引インタフェースサブシステムは、エンドユーザに関するコンテンツ情報を格納するために、アプリケーションにインタフェースを提供することに責任を有する。これに含まれるのは、その一部として、ローンを完了するために必要な情報、顧客の人口統計に関する調査ベースの情報、勘定残高の質問、ファンドトランスファである。

【0246】

顧客リード取引実行：

顧客リード取引実行アプリケーションは、インタフェースサポートフレームワークを促進し、テクニカルアーキテクチャの報告サブシステムのサービスがエンドユーザに関する情報の収集及び格納をサポートすることを可能にする。

【0247】

免責：

取引インタフェースの現在の使用方法では、非同期通信のみが利用可能であることを仮定している。

【0248】

サーバアプリケーションサービスサブシステムモデル：

アーキテクチャの概要：

このアプリケーションサービスサブシステムは、M M Tサーバアプリケーション及びアーキテクチャモジュールを構築するための定義のセットを含む。

【0249】

役割：

サーバアプリケーションサービスサブシステムは、アプリケーション及びアーキテクチャ機能にアクセスするためのサービス及び定義のセットを含む。このサブシステムは、アプリケーションサーバ又は運用ワークステーション端末上でアプリケーション及びモジュールを構築及び実行するための構造及びサポートサービスを定義する。サポートされる機能には、M M Tとの間の取引処理、例えば、顧客紹介情報、顧客対話情報、M I S情報、障害情報、製品レート及び価格情報、アプリケーション設定情報、メッセージ受領情報、及びハートビートステータス情報が含まれる。

【0250】

責任：

サーバアプリケーションサービスサブシステムは、基盤となるデータベース管理システム

10

20

30

40

50

から独立して、サーバ上のアプリケーションビジネスロジックを処理する。このアプリケーションビジネスロジックには、顧客紹介情報、顧客対話情報、M I S 情報、障害情報、製品レート及び価格情報、アプリケーション設定情報、メッセージ受領情報、及びハートビートステータス情報が含まれる。

【0251】

サーバアプリケーションサービスサブシステムは更に、エラーのログ、一定のコードのデコード（例えば、「1」=N S W、「2」=Q L D等）、及びそれぞれのマシンに位置するレジストリからの設定情報の検索を行うために、サーバ及びクライアントアプリケーションが利用可能な共通サービスを提供する

【0252】

最後に、サーバアプリケーションサービスサブシステムは、一定のB Pメッセージに関するビジネスプロセス（B P）を呼び出す。

【0253】

論理コンポーネント：

サーバアプリケーションサービスサブシステムは、共通サービスと、データアクセスモジュールと、ビジネスプロセスと、ビジネスプロセスアクセスモジュールとを含む。それぞれのコンポーネントの定義は下の通りである。

【0254】

コンポーネントの説明：

定義：

共通サービス。コード表のデコードと、レジストリからの設定情報の検索と、メッセージ処理と、サーバアプリケーションのエラーのログ及び処理に関するサポートとを含むアプリケーション機能の開発をサポートする共通サービス。

【0255】

データアクセスモジュール。データアクセスモジュール（D A M）は、アプリケーションデータベース内のデータへのアクセスを提供する。D A Mは、挿入、削除、更新、選択、1つ以上の表の全選択といった特定のデータアクセスを実行する。これは、基盤となるD B M SにアクセスするためにO D B Cを使用するレコードセットオブジェクトをカプセル化するM F C拡張D L Lである。D A Mの定義は、こうしたモジュールがどのように使用及びコード化されるかの概略を表す。

【0256】

ビジネスプロセス。ビジネスプロセス（B P）は、ビジネスプロセスアクセスモジュールアーキテクチャによって呼び出すことが可能なアプリケーション又はアーキテクチャ機能である。B Pはメッセージタイプによって特定される。B Pは、B Pによって定義された要求メッセージを受け入れ、同期メッセージに応答メッセージを提供することができる。データベースアクセスは、D A M SによってB Pに提供される。B Pは定義済みエントリポイントを有するM F C拡張D L Lである。

【0257】

ビジネスプロセスアクセスモジュール。このコンポーネントは通信サブシステムにおいて詳述されている。ビジネスプロセスアクセスモジュール（B P A M）は、ビジネスプロセスへのアクセスを提供するアーキテクチャコンポーネントである。B P A Mは一定のメッセージタイプに関するB Pを呼び出す。B P A Mはメッセージアドレス表にアクセスし、B Pモジュールの詳細をルックアップする。このコンポーネントは通信サブシステムにおいて詳述されている。

【0258】

好適な実施形態による無線電子バレット：

本発明の1実施形態は、モバイルポータル及び電子バレットを含むモバイルポータルプラットフォームである。電子バレットは、シンクライアントソフトウェアを実行するハンドヘルド型無線コンピュータ装置である。電子バレットには、G P S、バイオセンサ、及び環境センサといった様々なセンサが統合される。加えて、カメラ及び音声レコーダ等の記

10

20

30

40

50

録機器も電子バレットに統合される。モバイルポータルには、インターネット又はモバイルポータルエクストラネットを通じて、様々なサードパーティのコンテンツ及びサービスプロバイダに接続されるモバイルポータルサーバが含まれる。

【0259】

図26は、本発明の1実施形態のハードウェア及びソフトウェアがどのように動作するかを表したフローチャートである。電子バレット2602は、この無線ハンドヘルド型装置に統合されたセンサ、GPS、カメラ、マイクロフォン、及びその他のユーザ入力2600からの入力データを受領する。電子バレット2602上で実行されるシンクライアントアプリケーションにより、下で詳しく説明するように、電子バレット2602は、大量の内部メモリ及び記憶容量を必要とせず、多数の異なるソフトウェアアプリケーションを
10 実行することができる。電子バレット2602は、受領したデータ及びユーザ入力に基づいてメッセージを形成する。その後、電子バレット2602は、アンテナ2604を介して、モバイルポータル2606にメッセージを送送する。モバイルポータル2606は、電子バレット2602から受領したメッセージをパースし、受領したメッセージに基づいて新しいメッセージを形成する。次に、モバイルポータル2606は、無線ハンドヘルド型装置2602から受領したメッセージのコンテンツに基づいて、新しいメッセージを送送するのに適切なサードパーティサービスプロバイダ2608を判断し、その後、新しいメッセージを送送する。次に、サードパーティサービスプロバイダは、適切なサービスを実行し、結果をモバイルポータル2606に返送する。その後、モバイルポータルはサードパーティサービスプロバイダ2608から受領したデータに基づいてメッセージを形成し、このメッセージを電子バレット2602に返送する。その後、電子バレットは受領したデータをフォーマット及び表示する。電子バレット2602は、メトリコムのRicochet SE Wireless Modem等の無線モデムを利用する。
20

【0260】

当然ながら、無線性能には、従来のダイヤルアップ接続と同様の信頼性はない。サンフランシスコのいくつかの場所では、窓の近くにいる限り、強力な接続を得ることができた。しかし、CNETの総レンガ造りの本社内では、Ricochetはまったく接続できなかった。オンラインでつながっている時は、遅い速度へのゆっくりとした低下と共に28.8kbpsまでの性能が利用できる。しかし、遅い速度でも失望はなかった。代替方法——セルラモデムによる接続——に比べ、Ricochetは遥かに速く、信頼性があり
30、使用コストが少ない。当然ながら、このSE Wirelessはバッテリー電源である。このモデムの連続バッテリー寿命は12時間までである。

【0261】

したがって、無線モデムを利用することで、ユーザは電子バレット2602を介してモバイルポータル2606を利用することができる。適切な(複数の)キーを使用することで、ユーザは、センサ、GPS、カメラ、マイクロフォン、及びその他のユーザ入力2600から取得した適切なデータに呼応して、使用するサービスを選択することができる。

【0262】

特定の状況においては、電子バレット2602から取得したデータのタイプ及び値に基づいて選択したサービスにデータを自動的に送ることができる。例えば、統合されたバイオセンサが特定の事前に定められたデータ値を取得した時、適切な救急治療提供者が自動的に連絡を受ける。加えて、センサ、GPS、カメラ、マイクロフォン、及びその他のユーザ入力2600から取得したデータは、適切なサービスプロバイダへ送る前に組み合わせることもできる。例えば、上の例では、GPS位置データをバイオセンサのデータと共に救急治療提供者に送ることができる。救急治療提供者はこれにより、患者の生物学的データと患者の位置とを知るようになる。その後、適切なサービスが提供される。
40

【0263】

モバイルポータルプラットフォーム：

モバイルポータルプラットフォームは、好適な実施形態によるインパクトの強いサーバベースのアプリケーションであり、サーバ上に位置するパーソナルサイトを通じて、それぞ
50

れの顧客にサービスを与え、パーソナライズされた体験を提供するというテーマに注目したものである。直観的に編成されるこのサービスは顧客の意図－基本的な生活のニーズ又は広範なプランニングを要する判断と、財務プランニング、ヘルスケア、個人的及び職業的成長、家族生活、その他の問題といったいくつかの次元に渡る調整を満足させることを中心に直観的に編成されている。それぞれのメンバーは自分自身のプロフィールを保有及び維持しており、これにより、自分自身を限定的にターゲットとするシステムにおいてコンテンツを作成及びブラウズすることが可能となる。サービスの要求を入力した時点から、インテリジェントエージェントを利用して、リサーチが実施され、取引が実行され、アドバイスが提供される。高度なプロファイリング及びフィルタリングを使用することで、インテリジェントエージェントはユーザについて学び、提供するサービスを改善する。

10

【0264】

システムの好適な実施形態では、GPSレシーバ付きWindows CE PDAを利用する。この実施形態は、複数の店舗を含むショッピングセンタ用に構成されている。このシステムはGPSレシーバを使用して、ユーザの位置を判断する。このシステムの利点の1つは、ショッピングセンタ自体での何らかの特別な機器の存在に依存することなく、近くの店舗でデータを検索することが可能になる点である。小型で安価なレシーバの精度は約75乃至100フィート(23乃至31m)に限られているが、これはこれまでのところ、近隣の店舗を正確に特定するために必要なすべてであることが証明されている。このシステムは、実際の店舗の広告及び価格ではなく、生成されたデータを使用する。十分に構造化されたオンラインカタログが使用される。他の実施形態では、比較的小さな量の知識を使用して一定の店舗で「止まることを学習する」エージェントを利用する。更に、小売店がオンラインカタログを使用するための標準パッケージを使用し始めるにつれ、異なるフォーマットの数が減ることが予想され、結果として、競合するフォーマットの数是对応が容易なものになる。電子商取引が進歩するにつれ、商品の提供をどのように表すかの管理を発展させる標準を期待することも不合理ではない。

20

【0265】

目的指定：

買い物に出かける前に、買い物客は、約85製品カテゴリ(男性用カジュアルパンツ、女性用フォーマルシューズ、花、その他)の既存のセットから、購入する品目のリストを作成する。買い物客は更に、ショッピングセンタのリストから、訪れるつもりの買い物場所を示す。

30

【0266】

初期の店舗選択：

ショッピングセンタに到着した後、最初に、目的指定中にユーザが入力したタイプの品物を少なくとも1つ販売している最寄りの店舗を示す。店舗名と共に、好適な実施形態のシステムは、購入可能な具体的な品目及びその価格のリストを作成する。ショッピングセンタの地図では、店舗の正確な位置と、その買い物客の現在の位置とを両方表示する。買い物客は、任意の時期に、現在の位置に基づいて店舗を示すように問い合わせる。

【0267】

ブラウジング：

40

全般的に特定の目的地を決めずにショッピングセンタを訪問又は買い物している多くの買い物客のニーズに対処する。図27Aは、本発明の好適な実施形態によるディスプレイを表している。このディスプレイは、ショッピングセンタ内を散策する時に買い物客が使用するブラウズモードで動作している。ブラウズモードでは、システムは、現在買い物客に最も近い店舗において販売されている関心のある品目を示す。品目は、目的スクリーンに入力されたカテゴリと一致する場合、関心のあるものとみなされる。関心のある品目がない場合は、特定の品目ではなく、その店舗で販売されている一般的な商品のタイプが表示される。買い物客が歩くにつれ、地図では、その買い物客のショッピングセンタ内での正確な現在の位置が表示される。ブラウジング中に表示された品目を買い物客が選択した場合、システムは買い物客に対して、同じ製品をより安い価格で提供している現地の小売店

50

を警告するか、或いは現地の最安値を知らせる。このサーチは現地のショッピングモールに限定され、これは買い物客が移動する意志を有すると仮定された区域となる。

【0268】

代案：

強調する価値がある点として、本発明のエージェントはショッピングタスクの広範な態様をサポートし、例えば、双方向チャンネルとして動作できる。つまり、買い物客に情報を提供できるだけでなく、買い物客の裁量において、小売店に情報を提供することもできる。この実施形態では、このシステムは買い物客の目的と好みを小売店ベースのエージェントに示し、このエージェントは次に、製品にサービスをバンドルしたカスタマイズされたオファーによって応答する。オファーのカスタマイズを可能にすることは、価格のみで競合したくない小売店の協力を得るために非常に重要であり、価格以外の基準に基づいて購入する顧客にとって価値のあることである。好適な実施形態は、主にショッピングタスクのコンテキストにおける位置ベースのフィルタリングに注目しているが、本発明は、物理世界において様々なタスクに従事する人に情報チャンネルを提供する「物理的タスクサポート」エージェントの基盤を提供する。

10

【0269】

位置の予測値：

ユーザの物理的な位置は、ユーザのタスクを予測するのに非常に有効である場合が多いため、本発明は位置ベースではないエージェントに比べて大きな利点がある。誰かがボウリング場又は郵便局にいることが分かったなら、現在の活動を合理的に推測できる。ユーザの現在のタスクに関する知識は、そのユーザが有益と思う可能性が高い情報のタイプをほぼ決定する。ボウリングをしている人が郵便料金を、或いは切手を買っている人が最適なボウリングの球の重さを気にする可能性は低い。加えて、特定の場所でのリソース及び障害に関する知識は、その場所にいる人物にとって可能であり実行すると思われる行動の範囲を示す。ユーザにとって可能であり実行すると思われる行動に関するこうした認識は、ユーザが有益と思う可能性が高い情報のタイプに更に制約を加えるために使用できる。例えば、レストランのワインリストの知識は、提示するワインのアドバイスを制約を加えるために、推奨システムによって使用することができる。

20

【0270】

ショッピングセンタでの買い物客の正確な位置に関する知識は、買い物客の近辺の店舗の特定を可能にするため価値がある。買い物客に最も近い店舗の提供内容は、その買い物客がすぐに選択可能なものを意味する。その買い物客が、商品を直接調べることが重視し、他のストアへ行くことに伴うコストがある場合、最も近い周辺の店舗の商品は、その買い物客にとって最も可能性の高い当面の選択肢となる。結果として、任意の一定の時間に提供される最も有益な情報は、周辺の店舗における以前に述べた目的と一致する商品の入手可能性である。

30

【0271】

人間は自分の多くのタスクを実行する間に異なる場所へ移動する傾向がある。これは、その人物の近辺が、入手できる可能性のあるあらゆる範囲の選択肢を完全に捕らえていないことを示す。実際、ある場所を立ち去る大きな理由の1つは、現在の場所では不可能な行動を実行することである。それにもかかわらず、人間には比較的近くのエリア内でほとんどのタスクに対処する傾向がある。したがって、自分の近辺は一定の時点で利用可能な選択肢を示し、更に広範な場所の観点はタスクを行う過程で考慮する可能性の高い選択肢を捕らえている場合が多い。ショッピングセンタでの買い物客の場合、例えば、買い物客の近辺の店舗は、その時点で利用可能な選択肢を表す。しかしながら、ショッピングセンタの買い物客は、一般にショッピングセンタ内の任意の店舗に移動する意志を有する。そのため、買い物全体での可能な選択肢には、ショッピング内のすべての店舗が含まれる。したがって、関心のある提供物に関する情報を近辺の店舗のみから提示するのは、これらがすぐに利用可能な選択肢となるためである。代案を求められた時、システムはショッピングセンタ内のすべての店舗—つまり、ショッピングタスク全体が実行される可能性が高いエ

40

50

リアにシステム自体を限定する。数百又は数千マイル離れた店舗が、最も安価な現地での代案よりも数ドル安く同じ商品を販売していることを警告されても、買い物客が問題の商品を直接調べる必要がある場合、或いは出荷を待つ意志がない場合にはほとんど価値がない。

【0272】

物理的ショッピング対オンラインショッピング：

位置ベースではないエージェントに対する大きな利点に加えて、本発明はオンライン（又はWeb）ショッピングの欠点を克服する。更なるホームコンピュータの浸透、オンラインでの提供物の拡大、及び優れたオンラインショッピングツールのみによって、やがてオンラインショッピングが支配的なショッピング形態になるという主張は魅力的である。そのため、一見したところでは、後にバーチャル化される活動をサポートするために場所を使用し始めるのは誤りのように思える。既に、オンラインショッピングをサポートする多数のソフトウェアエージェントが出現している。例えば、ユーザが音楽CDの最も安価なソースを特定することを可能にするプログラムに名称が与えられている。同様のプログラムは、BargainBot等、書籍の購入に関しても開発されている。こうしたシステムは、特定の商品に関する完全な市場を創り出す電子商取引Webエージェントの可能性を実証している。こうしたエージェントの成功は、更に多様な商品に関する同様のWebショッピングエージェントの開発を促進する。

10

【0273】

オンラインショッピングの限界：

間違いなく、オンラインショッピングの成長は続き、更に強力なオンラインショッピングエージェントに向かう傾向は継続する。それにもかかわらず、Webエージェントがどれだけ精巧になったとしても、予測可能な未来において、従来の物理的ショッピングが市場を支配し続けることは明らかであると思える。オンラインショッピングに内在するいくつかの難点は、物理的ショッピングの継続的な信頼を確かなものにする。

20

【0274】

代替不能物：

Webベースのショッピングエージェントは通常、書籍及び音楽CDといった代替可能製品に関する最も安い価格をユーザが特定することを可能にする。こうした商品に関する「完全な市場」を創り出す能力は消費者にとって大きな恩恵だが、こうしたアプローチを任意の製品に応用することを複雑にするいくつかの難点が存在する。

30

【0275】

商品がショッピングエージェントに特に最適なのは、競合するオファーとの間での比較が容易なためである。商品は代替可能であるため、差異がある非常に数少ない次元は価格となる。そのため、価格は、唯一ではないものの、購入の判断を行う主要な基準となる。

【0276】

しかしながら、商品を超えて考えるとすぐに、他のいくつかの基準が重要になる。例えば、セータ、マットレス、又はテーブルといった品目をどのように比較するだろう？ 価格に加え、使用されている材質と、色と、フィット感及び使い心地と、出来映えとに注意する。同様の問題は他のほとんどの製品に当てはまる。

40

【0277】

不正確な目的指定：

第二に、関連する難点は、自分の欲求をエージェントに伝えることにある。ショッピングエージェントは、自分の欲しい商品をユーザが正確に知っている場合には素晴らしい働きをする。この場合、ユーザは製品の名前を単純に入力すればいい。残念ながら、買い物する時に特定の品目を思いついていない場合、何が欲しいかをエージェントに伝える問題はずっと困難になる。例えば、ユーザは自分の居間に置きたいランプの種類をエージェントにどのように伝えればいいだろう？

【0278】

不明確な好み：

50

買い物客が、価格帯、色、オプション、ブランドその他といった特徴の説明を含めることを可能にするインタフェースは、上の問題に対処するのに役立つが、これは十分ではない。たいていの場合、買い物客は、ショッピングを開始し、目的の製品の様々な例を調べる機会を持つまで、好みを確立していないか、或いは欲しい物を正確に表現することができない。

【0279】

ショッピングは娯楽である：

人間は買い物が好きであり、特定の購入品が頭になくとも買い物をする。ある研究では、消費者の42%は、レジャーとしてのウィンドウショッピング及び社交活動のためにショッピングセンタを訪れた「目的のない買い物客」である。

10

【0280】

ショッピングは感覚的である：

ユーザが、こうした詳細を効果的に提供できるとしても、大部分のユーザが購入の決断をこうしたエージェントに任せる可能性は低い。結局、多くの人は、信頼する配偶者であっても、自分の代わりに適切な購入を行うかどうかを不安に感じる。ほとんどの人は、購入の決断をする前に、考慮しているものを直接目で見て、手で触れたいと考える。エージェントに提供可能な僅かな好みを、この豊かな直接的な体験に置き換えることはできない。こうした好みは、買い物客が考慮する一連の候補を生成するのに使用できる程度のものである。

【0281】

20

瞬間的な満足感：

ショッピングは往々にして非常に感情的な活動となる。人間は購入品に満足し、家に帰って試してみるのが待ちきれない場合が多い。オンライン購入と受領との間の本質的な遅延は、確認後すぐに家へ持ち帰る必要がある人にとって大きな問題となる。

【0282】

結局、消費者は上で挙げた限界から、物理的ショッピングに従事し続けることになる。しかしながら、タスクを完全にソフトウェアエージェントに任せることができないという事実は、こうしたエージェントの役割を排除するものではない。第一に、欲しい物が分かっている時、商品の購入に関して、エージェントはユーザにとって有益である。しかしながら、第二の役割は、人間が物理的なショッピングタスクに従事する時間全体を通じて、これ自体をサポートすることである。これは当然ながら、ショッパーズアイプロジェクトにおいて用いられるアプローチである。

30

【0283】

ショッパーズアイ：

一見すると、本発明は、純粋なWebベースエージェントと同じ限界のいくつかによって影響を受けるように思える。結局のところ、自分の目的をPDAに伝えることは、Webベースエージェントにつたえることよりも容易になるだろうか？ PDAシステムによってサポートされる購入では、Webベースエージェントの場合よりも自分の好みは明確になるだろうか？

【0284】

40

純粋なWebベースエージェントと現在の「物理的タスクサポートエージェント」（つまり、物理的な場面でタスクに従事するユーザをサポートするエージェント）との間の主な違いは、Webベースエージェントが、ユーザによって考慮されるあらゆる情報を伝えることに完全に責任を有する点にある。一方、好適な実施形態による「物理的タスクサポート」エージェントは、ユーザの環境の態様を参照することで、Webベースエージェントのアプローチを補強することができる。例えば、30フィート（9m）離れた店舗に特定のセータがある場合、そのセータの使い心地を十分に伝えることは極度に重要な事柄とはならない。必要なのは買い物客にそのセータを紹介することのみである。買い物客は、そのセータを試着してみることで、システムによって伝えることが可能ないかなる情報よりも遥かに優れたセータの認識を得ることができる。Webベースエージェントで不正確に

50

指定された目的に一致する製品が多すぎる時は、更に限定的なサーチを行う必要がある。しかしながら、多くの一致は単純に、買い物客が大きな関心を持つ可能性が高い店舗が存在しているため訪問するべきであることを示しているに過ぎない。店舗に入ってしまうと、その人が関心を持つ商品の範囲を狭めることは、Webベースエージェント上で目的を精緻なものにすることより遥かに容易である場合が多い。そのため、物理的タスクサポートエージェントは、タスクの過程全体を通じて情報を伝達する手段として、物理的な環境の態様にユーザの注意を促すことで、ユーザが自分の好みを形にして、具体的な目的を特定するのを支援することができる。

【0285】

物理的ショッピングエージェントの約束：

10

物理的なショッピングがエージェントのコミュニティによって無視されてきたのは驚くべきことではない。結局、ごく最近になるまで、離れた場所にいる個々の買い物客にカスタマイズされた情報を届けるための信頼性の高い方法はまったく存在しなかった、しかしながら、PDAの爆発的な発展と、更に高度になる通信能力とは、PDAがユーザの居場所に関係なく「ジャストインタイム」に情報を伝える効果的なチャネルになることを約束している。本発明は、ショッピングタスクのあらゆる段階をサポートするチャネルの発展の約束を活用して、物理的ショッピングをサポートする直観的な新しいエージェントを提供し、以下を含む前記の問題を解決する。

【0286】

目的の指定：

20

買い物客は最初に、関心を持っている商品の少なくとも一般的なカテゴリを示す。ショッピングエージェントは、様々な特定性の度合いにおいて、目的の指定を可能にする必要がある。本発明では、こうした目的はタスクの進捗と共に精緻なものにすることができる。

【0287】

製品スペースの調査：

買い物客は、選択を行えるようになる前に、利用可能なものについて知識を得る必要がある。ショッピングエージェントは、様々な提供内容のクラス、レビュー、実演その他を提示することで、このタスクを支援できる。本発明の物理的ショッピングエージェントは、現地で利用可能な提供内容のツアーを買い物客に提供することで、これを補強することができる。

30

【0288】

好みの精緻化：

買い物客が利用可能なものを学習し、提供内容を吟味するにつれ、買い物客の好みは発展していく。エージェントは、買い物客が経時的に好みを精緻化するのを可能にする必要がある。本発明により、ユーザは好みを精緻化することができる。

【0289】

候補製品の特定及び比較：

買い物客は、自分の欲しいもの及び入手可能なものを理解し始めるにつれ、通常、更に慎重に考慮する候補のリストを編集する。本発明のエージェントは、こうしたリストの構築及び維持をサポートし、様々な基準によるリスト内の候補の比較を容易にする。

40

【0290】

オファの交渉：

本ショッピングエージェントは、買い物客に情報を提供することに限定されない。小売店との価格及びサービスオプションの交渉が可能となる。

【0291】

製品の選択、購入、及び製品サポート：

本発明は、取引自体を促進し、製品サービスを提供可能なチャネルとして使用することができる。

【0292】

図27Bは、モバイルポータル2412と電子バレット2713とを含むモバイルポータ

50

プラットフォーム 2710 の図である。電子バレット 2713 は、無線 PDA 等のサポートハードウェア装置 2716 と、シンククライアントオペレーティングシステム 2718 の上で実行されるモバイルポータルシンククライアント規格 2714 とを含む。モバイルポータルは、暗号化及び解読要素 2720 と、インテリジェントエージェント 2724 を含むモバイルポータルサーバ 2722 と、顧客インテリジェント要素 2726 と、顧客データベース 2728 とで構成される。

【0293】

シンククライアントとは、削減されたハードウェアコストと、削減された維持及びサポートコストと、削減された LAN/WAN 帯域幅要件と、短縮されたダウンタイムと、改善された性能と、強化されたセキュリティとの組み合わせによって、全体的な所有コストの削減を提供する、急速に現れつつあるテクノロジーグループを表すために使用される一般的な用語である。シンククライアントの「シン（薄い）」という用語は、クライアントオペレーティングシステムの（非常に小さな）サイズを指す。対照的に、従来の PC オペレーティングシステム（DOS、Windows 95、その他）は、その大きなサイズ及びリソース要件から「ファット（太い）」クライアントとみなされる。シンククライアントオペレーティングシステムがシンである事実にもかかわらず、シンククライアントの能力は堅牢である。クライアントソリューションは現在、ミッションクリティカルな環境に配備されており、無数のアプリケーションに信頼性及び反応性に優れたアクセスを提供している。モバイルポータルシンククライアント 2714 は、処理の大部分がモバイルポータルサーバ 2722 と、関連するサードパーティコンテンツ及びサービスプロバイダ 2730 とで実行されるシンククライアントである。ユーザはモバイルポータルシンククライアントアプリケーション 2714 を利用して、モバイルポータルプラットフォーム 2710 が提供するサービスを選択し、情報をレビューする。モバイルポータルシンククライアントアプリケーション 2714 は、シンククライアントオペレーティングシステム 2718 を使用して、多くの装置を独立させる。シンククライアントオペレーティングシステム 2718 は、モバイルポータルシンククライアントアプリケーション 2714 とサポートハードウェア 2716 との間のメッセンジャとして働く。シンククライアントオペレーティングシステム 2718 により、モバイルポータルシンククライアント 2714 は、ディスプレイ呼び出し及びユーザ入力クエリといった低レベルなハードウェア動作に関して、シンククライアントオペレーティングシステム 2718 に対して関数呼び出しを行うことができる。電子バレット 2713 用のサポートハードウェアとして使用されるそれぞれのハードウェア装置 2716 に関して、別個のシンククライアントオペレーティングシステム 2718 を開発することができる。これにより、モバイルポータルシンククライアントアプリケーション 2714 は、低レベル設計の修正を必要とせず、異なるサポートハードウェア 2716 上で実行できる。

【0294】

モバイルポータル 2712 は、パケット交換無線ネットワーク 2732 を介して、電子バレット 2713 からデータを受領する。パケット交換無線ネットワークを通じて受領された情報は、その後、モバイルポータル 2712 の暗号化及び解読要素 2720 によってデコードされる。データがデコードされると、モバイルポータルサーバ 2722 はインテリジェントエージェント 2724、顧客インテリジェンス 2726、及び顧客データ 2728 を利用して、要求されたデータをサードパーティコンテンツ及びサービスプロバイダ 2730 から取得する。モバイルポータルサーバ 2722 は、インテリジェントソフトウェアエージェントを利用して、顧客のニーズに応答する。ソフトウェアエージェント 2722 は、顧客データ 2728 を利用して、個々のユーザの目的、習慣、及び好みに合わせてタスクをパーソナライズするための判断を行う。次に顧客データ 2728 は、顧客及び顧客の行動によって通常通りに更新される。ユーザがモバイルポータル 2712 を使用するたびに、ユーザによるクエリ及びその他のモバイルポータルプラットフォーム 2710 の使用に関してログが記録される。これにより、ソフトウェアエージェント 2724 は、ユーザの過去の習慣を利用して、タスクをパーソナライズすることができる。

【0295】

ソフトウェアエージェント 2724 に加え、モバイルポータルサーバ 2722 は、顧客インテリジェンス 2726 を利用して、ユーザのニーズに応答する。ユーザは、データの掘り出し及びパターン認識を利用して、望みの情報を見つけることができる。ここでも、顧客のデータ 2728 は更新され、ユーザによるデータの掘り出し及びパターン認識の使用が反映される。サードパーティコンテンツ及びサービスプロバイダ 2730 は、ユーザによって要求されたサービス及び情報を提供するために、モバイルポータル 2712 によって利用される。サードパーティコンテンツ及びサービスプロバイダには、インターネット又はモバイルポータルエクストラネットを通じてアクセスすることができる。インテリジェントエージェントソフトウェア 2712 は、サードパーティプロバイダをサーチし、顧客データ 2728 に含まれる顧客のプロフィールを考慮して、ユーザにとって最適なものを判断する。これにより、ユーザはクエリにおける特定性を、ユーザプロフィールがない場合に行わなければならないものに比べて減らすことができる。例えば、ユーザはモバイルポータルプラットフォーム 2710 を利用してジャケットを要求することができる。インテリジェントエージェントは、その後、顧客データ 2728 を利用して、顧客が実際に望んでいるものを更に具体的に決定する。この場合、顧客データ 2728 は、この特定のユーザがレザージャケットではなくデニムジャケットを好むことを知らせる。インテリジェントエージェント 2724 は、その後、デニムジャケットをサーチすることになる。当然ながら、ユーザプロフィールに格納されているものとは反対の情報を取得するために、ユーザはユーザプロフィールを無効にすることができる。提供される代表的なサービスの一部には、地理的位置情報、音声及び画像編集、個人向けニュース及び娯楽、個人向けショッピング、個人向け健康及び安全、個人用オーガナイザ、個人向け金融、及び個人用通信が含まれる。

10

20

【0296】

地理的位置サービスは通常、統合されたグローバルポジショニングシステムから受領した情報に基づく。GPS データは、ユーザに位置固有の情報を提供するために特定のユーザ要求データと組み合わせる。例えば、ユーザがサンフランシスコにおり、この都市での良質な食事の情報を取得したいとする。このユーザは、電子バレット 2713 を利用して良質な食事の情報を要求する。統合された GPS レシーバから取得した位置データは、ユーザの良質な食事に関する要求と組み合わせられ、その後、この組み合わせ済みメッセージがモバイルポータル 2712 に伝送される。受領したデータに基づき、モバイルポータルは適切なサービスを選択し、この場合はサンフランシスコでの良質な食事について、要求を伝送する。モバイルポータルは次に、受領した応答を電子バレット 2713 に返送する。その後、ユーザには要求した情報が提示され、電子バレット 2713 のディスプレイ装置上でフォーマット及び表示される。

30

【0297】

音声及び画像編集サービスは通常、統合されたカメラ及びマイクロフォンから受領したデータに基づく。ユーザは通常、統合されたデジタルカメラを利用して、画像を取り込む。しかしながら、ユーザは、スキャナ、電子メール、及び Web ページといった他のソースからデジタル画像を取得することもできる。加えて、ユーザは通常、統合されたマイクロフォンを利用して音声ファイルを取り込む。しかしながら、音声ファイルは、電子メール、Web ページ、及び CD といった他のソースから取得することもできる。画像及び又は音声データは、ユーザに画像及び音声編集能力を提供するために、特定のユーザ要求と組み合わせる。例えば、ユーザは、統合されたデジタルカメラによって画像を取り込み、その後、特定の写真エディタを使用して画像を編集することを要求できる。統合されたデジタルカメラによって取り込まれた画像は、次に、写真の編集に関するユーザの要求と組み合わせられ、その後、この組み合わせ済みメッセージがモバイルポータル 2712 に伝送される。受領したデータに基づき、モバイルポータル 2712 は適切なサービスを選択し、この場合は画像編集の要求を伝送する。モバイルポータルは次に、受領した応答を電子バレット 2713 に返送する。その後、ユーザには要求した情報が提示され、電子バレット 2713 のディスプレイ装置上でフォーマット及び表示される。この場合、ユーザは画像

40

50

編集用のインタフェースを受領することになる。ユーザはその後、この画像編集用ユーザインタフェースを使用して画像を編集する。画像の変更は要求として扱われ、モバイルポータル2712はこれを、ローカル又は独立したサーバで実行される画像編集アプリケーションに渡す。

【0298】

好適な実施形態による生物医学センサの統合：

本発明の1実施形態は、圧力トランスデューサ、呼吸センサ、ボリユメトリックセンサ、及び除細動器等の統合された生物医学センサを含む電子バレットである。

【0299】

血圧を測定するために統合圧力トランスデューサは、浸透性および非浸透性の2つのタイプであり得る。浸透性統合圧力トランスデューサでは、ユーザは血流内にユニットの一部を埋め込む必要があり、非浸透性統合圧力トランスデューサは血流にアクセスする必要がない。圧力トランスデューサは、患者の血圧を測定し、この場合は電子バレットである受信ユニットにこれをレポートする。その後、電子バレットは、モバイルポータルシンククライアント及びモータルポータルサーバを利用して、圧力トランスデューサから受領したデータを分析及び経路指定することができる。

10

【0300】

歪みゲージ及びボリユメトリックセンサ等の呼吸センサも電子バレットに統合することができる。胸部の周りに装着した歪みゲージは、呼吸中の胸の拡張及び収縮によるゲージの拡張及び収縮に従ってインピーダンスを変化させる。ボリユメトリックセンサは、患者がボリユメトリックセンサに息を吹き込んだ時等に、センサを通過する気圧の量を検知する。歪みゲージとボリユメトリックセンサとは両方とも、対応するデータを電子バレットユニットに無線伝送することができるため、ユーザには活動における大きな自由が与えられる。圧力トランスデューサから受領したデータと同様に、歪みゲージ及びボリユメトリックセンサから受領したデータは、モバイルポータルプラットフォームを利用して分析及び経路指定することができる。

20

【0301】

電子バレットに統合された除細動器は、心臓機能を検知するために利用することができる。除細動器は、生理食塩水ベースのゲルを利用して患者に取り付けられ、R、T、及びP波によって鼓動を検知する。歪みゲージ及びボリユメトリックセンサと同様に、除細動器はデータを電子バレットユニットに無線伝送することが可能であり、電子バレットユニットはその後、モバイルポータルプラットフォームを利用してデータを分析及び経路指定する。

30

【0302】

前記の生物医学センサは、特定のユーザのニーズに応じて、個別に統合すること、或いは環境センサ、他の生物医学センサ、又はGPSレシーバ等のその他のセンサと組み合わせて統合することが可能である。例えば、心臓の問題に関する病歴を有する高齢のユーザは、統合された除細動器及びGPSレシーバを含む電子バレットを有することができる。

【0303】

モバイルポータルプラットフォームを利用することで、ユーザは、有害である恐れがある様々な食品及び薬品を含め、自分の状態に関する最新のニュース及び情報を常に知ることができる。

40

【0304】

加えて、電子バレットは、ユーザの位置に関係なく、心臓の状態から発生する恐れがある問題を感知することができる。公園を散歩中、ユーザが胸の痛みを感じた場合、電子バレットは、ユーザの心臓の状態から発生した問題によって痛みが生じていることを感知する。これは、統合された除細動器と電子バレットの分析能力とを利用することで達成される。この場合、統合された除細動器から受領したデータが事前に定められた安全な閾値を超えたため、電子バレットは緊急事態が発生したことを警告される。モバイルポータルプラットフォームを利用することで、電子バレットはその後、適切な救急対応ユニットに通知

50

を行い、その心臓に関するデータをユーザの医師に転送し、ユーザの家族に連絡する。

【0305】

加えて、電子バレットは、統合されたGPSレシーバから受領した位置座標を、救急対応ユニットに転送し、ユーザの位置の特定及び救助を可能にする。病院での治療の後、モバイルポータルプラットフォームは、ユーザの食事及び栄養摂取、更には運動の日課及び投薬を含め、ユーザのアフターケアプログラムを調整することができる。

【0306】

好適な実施形態によるサポートコード：

以下のコードは、好適な実施形態によるMicrosoft Active Server Pages環境において書かれ実行される。これは主に、データベースの情報のクエリ及び格納を行うためにコード内に埋め込まれたいくつかのデータベース呼び出しを有するMicrosoft Jscriptで構成される。

10

【0307】

意図中心インタフェース

【0308】

意図ASPページを作成する（「intention_create.asp」）

【0309】

【表8】

```
<%@ LANGUAGE = "JScript" %>
<%
Response.Buffer = true;
Response.Expires = 0;
%>

<html>
<head>
    <title>Create An Intention</title>
</head>

<body bgcolor="#FFE9D5" style="font-family: Arial" text="#000000">

<%
//Define some variables

upl = Server.CreateObject("SoftArtisans.FileUp")
intention_name = upl.Form("intention_name")
intention_desc = upl.Form("intention_desc")

//intention_name = Request.Form("intention_name")
//intention_desc = Request.Form("intention_desc")

//intention_icon = Request.Form("intention_icon")
submitted = upl.Form("submitted")
items = new Enumerator(upl.Form)
%>
```

20

30

40

```

<%
//Establish connection to the database
objConnection = Server.CreateObject("ADODB.Connection")
objConnection.Open("Maelstrom")
%>

<%
//Check to see if the person hit the button and do the appropriate thing
if (submitted == "Add/Delete")
{
    flag = "false"

    //loop through all the inputs
    while(!items.atEnd())
    {
        i = items.item()

        //if items are checked then delete them
        if(upl.Form(i) == "on")
        {
            objConnection.Execute("delete from user_intention where intention_id =" +
i);
            objConnection.Execute("delete from intentions where intention_id =" + i);
            objConnection.Execute("delete from tools_to_intention where intention_id
=" + i)

            flag = "true"
        }
        items.moveNext()
    }

    // if items were not deleted then insert whatever is in the text field in the database
    if(flag == "false")
    {
        intention_name_short = intention_name.replace(/ /gi,"")
        objConnection.Execute("INSERT INTO intentions
(intention_name,intention_desc,intention_icon) values('" + intention_name + "','" +
intention_desc + "','" + intention_name_short + ".gif" + "')")
        Response.write("the intention short name is " + intention_name_short);
        upl.SaveAs("E:development/asp_examples/" + intention_name_short + ".gif")
    }
}

// Query the database to show the most recent items.
rsCustomersList = objConnection.Execute("SELECT * FROM intentions")
%>

<input type="Submit" name="return_to_mcp" value="Go to Main Control Panel"
onclick="location.href='default.asp'">

<form method="post" action="intention_create.asp" enctype="multipart/form-data" >

```

10

20

30

40

```

<TABLE border=0>
<tr><td colspan="2"><font face="Arial" size="+1"><b>Enter in a new intention</b></font></td></tr>

<tr><td><font face="Arial">Name:</font></td> <td><INPUT TYPE="text"
name="intention_name"></td></tr>
<tr><td><font face="Arial">Description:</font></td><td><TEXTAREA
name="intention_desc"></TEXTAREA></td></tr>
<tr><td><font face="Arial">Icon Image:</font></td><td><INPUT TYPE="file" NAME="intention_icon"
size=40></td></tr>
<tr><td colspan="2"><INPUT type="submit" name="submitted" value="Add/Delete"></td></tr>
</TABLE>
<HR>
<font face="Arial" size="+1"><b>Current Intentions</b></font>
<TABLE>
    <tr bgcolor=E69780 align="center">
        <td>
            <FONT color="white">Delete</FONT>
        </td>
        <TD>
            <FONT color="white">Intention</FONT>
        </TD>
        <TD>
            <FONT color="white">Description</FONT>
        </TD>
        <TD>
            <FONT color="white">Image</FONT>
        </TD>
    </tr>

<%
// Loop over the intentions in the list
counter = 0;
while (!rsCustomersList.EOF)
{
%>
    <tr bgcolor="white" style="font-size: smaller">
        <td align=center>
            <INPUT type="checkbox" name="<%=rsCustomersList("intention_id")%>">
        </TD>
        <td>
            <%= rsCustomersList("intention_name")%>
        </td>
        <td>
            <%= rsCustomersList("intention_desc")%>
        </td>
        <td>
            ">
        </td>
    </tr>

```

10

20

30

40

```

<%
counter++
rsCustomersList.MoveNext()
%>
</TABLE>
<hr>
Available Tools
</form>
</BODY>
</HTML>

```

10

【0310】

意図リストASPページを検索する（「intention_list.asp」）

【0311】

【表9】

```

<!-- #include file="include/check_authentication.inc" -->

```

```

<HTML>

```

```

<HEAD>

```

```

<TITLE>mySite! Intentions List</TITLE>

```

```

<SCRIPT LANGUAGE="JavaScript">

```

```

function intentionsList () {

```

```

    this.internalArray = new Array();

```

```

    <%

```

```

    // establish connection to the database

```

```

    objConnection = Server.CreateObject("ADODB.Connection");

```

```

    objConnection.Open("Maelstrom");

```

```

    // create query

```

```

    intentionsQuery = objConnection.Execute("SELECT * FROM intentions ORDER BY
intention_name asc");

```

```

    %>

```

```

    // write out the options

```

```

    <%

```

```

    numOptions = 0

```

```

    while (!intentionsQuery.EOF) {

```

```

        intentionName = intentionsQuery("intention_name");

```

```

        intentionIcon = intentionsQuery("intention_icon");

```

```

    %>

```

```

        this.internalArray[<%= numOptions%>] = new Array(2);

```

```

        this.internalArray[<%= numOptions%>][0] = "<%= intentionName %>";

```

```

        this.internalArray[<%= numOptions%>][1] = "images/<%= intentionIcon %>";

```

```

    <%
        numOptions++; intentionsQuery.MoveNext();    %>

```

```

    <%
        }
    %>

```

```

}

```

20

30

40

```

numIntentions = <%= numOptions%>;
intentionArray = new intentionsList().internalArray;
function selectIntention () {
    for (i=0;i<numIntentions;i++) {
        if (IntentionsListSelect.options[i].selected) {
            intentionNameTextField.value = intentionArray[i][0];
            //intentionPicture.src = intentionArray[i][1];
            break;
        }
    }
}
</SCRIPT>

</HEAD>

<BODY BGCOLOR="<%=Session("main_background")%>" style="font-family: Arial">

<CENTER>
<!-- <FORM NAME="intention_list"> -->
<TABLE FRAME="BOX" border=0 CELLPADDING="2" CELLSPACING="2">

<TR><TD COLSPAN="3" STYLE="font: 20pt arial" ALIGN="CENTER"><B>Add a mySite!
Intention</B></TD></TR>

<TR><TD COLSPAN="3">&nbsp;</TD></TR>

<TR>
    <TD width="100"><font size="-1">Please Select An Intention You Would Like to Add to Your
List</font></TD>
    <TD colspan=2>
        <SELECT ID="IntentionsListSelect" NAME="IntentionsListSelect" SIZE="10"
style="font: 9pt Arial;" onClick="selectIntention()">
            <%
            intentionsQuery.moveFirst();
            for(j=0;j<numOptions;j++) { %>
                <OPTION VALUE="<%= intentionsQuery("intention_id") %>" <% if (j == 0) { %>
SELECTED <% } %>>
                <%= intentionsQuery("intention_name") %>
                <% intentionsQuery.moveToNext()
            }
            intentionsQuery.moveFirst();
            %>
            </SELECT>

        </TD>

</TR>

<TR><TD COLSPAN="3">&nbsp;</TD></TR>

```

10

20

30

40

```
<TR>  
    <TD width="100"><font size="-1">Customize the Intention name</font></TD>  
    <TD COLSPAN=2*><INPUT TYPE="text" NAME="intentionNameTextField"  
ID="intentionNameTextField" SIZE="30" VALUE="<% intentionsQuery("intention_name") %>"></TD>  
</TR>  
  
<TR><TD COLSPAN="3">&nbsp;&nbsp;&nbsp;</TD></TR>  
  
<TR>  
    <TD COLSPAN="3" ALIGN="CENTER">  
        <INPUT TYPE="button" NAME="intentionOKButton" VALUE="      CK      " SIZE="10"  
ID="intentionOKButton" onClick="javaScript:top.opener.top.navframe.addAnIntention();">  
  
        &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~  
        <INPUT TYPE="button" NAME="intentionCancelButton" VALUE="Cancel" SIZE="10"  
ID="intentionCancelButton" onClick="self.close();">  
  
    </TD>  
</TR>  
</TABLE>  
  
<!--- </FORM> --->  
</CENTER>  
  
<% objConnection.Close(); %>  
</BODY>  
</HTML>
```

【 0 3 1 2 】

ユーザ意図リストASPページを表示する（「navigation.asp」から抜粋）

【 0 3 1 3 】

【表 10】

```
<DIV ID="intentionlist" style="position: absolute; width:210; height:95; left: 365pt; top: -5;
visibility: hidden; font-family: Arial; font-color: #000000; font: 8pt Arial ; " >
<DIV style="position: absolute; top:7; left:7; height:78; width:210; z-index:2; background:
<%=Session("main_background")%>; border: solid 1pt #000000; padding: 3pt; overflow: auto; alink:
black; link: black;">
<body LINK="#000000" ALINK="#000000" vlink="black">
    <%
        // create query

        intentionsQuery = objConnection.Execute("SELECT user_intention.* FROM
user_intention, user_intention_to_persona WHERE user_intention_to_persona.user_persona_id = " +
Session("currentUserPersona") + " AND user_intention_to_persona.user_intention_id =
user_intention.user_intention_id" );

        numintentions = 0;

        Response.Write("<SCRIPT>numintentions=" + intentionsQuery.RecordCount +
"</SCRIPT><TABLE cellpadding='0' width='100%' cellspacing='0'>");

        while (!intentionsQuery.EOF)
        {
            %>
```

40

```

<TR><TD><a href="javascript:changeIntention('<%=
intentionsQuery("user_intention_id") %>', '<%=numintentions%>')" onmouseover="mouseOverTab()"
onmouseout="mouseOutOfTab()" "><font color="Black" face="arial" size="-2"><%=
intentionsQuery("intention_custom_name") %></font></a></TD><TD><IMG align="right"
SRC="images/delete.gif" alt="Delete this intention" onClick="confirmDelete(<%=
intentionsQuery("user_intention_id") %>)" "></TD></TR>

<%=numintentions++; intentionsQuery.moveToNext(); %>

<%
    }
    Response.Write("<SCRIPT>numintentions="+numintentions + "</SCRIPT>");
    %>
    <tr><td colspan="2"><hr></td></tr>
    <TR><td colspan="2"><a href="javascript:changeIntention('add
...','<%=numintentions%>');" onmouseover="mouseOverTab()" onmouseout="mouseOutOfTab()" "><font
color="Black" face="arial" size="-2">add ...</font></a></td></TR>
    </table>

</body>
</DIV>
<DIV style="position: absolute; top:0; left:-5; width: 230; height:105; z-index:1; "
onmouseout="intentionlist.style.visibility='hidden'"
onmouseout="intentionlist.style.visibility='hidden'"
onmouseover="intentionlist.style.visibility='hidden'"></DIV>
</DIV>
</DIV>

```

10

20

【0314】

以上様々な実施形態を説明してきたが、これらの実施例は、限定のためではなく例示のために説明されたものと理解されるべきである。したがって、好適な実施形態の幅及び範囲は、上で説明した各実施形態例のいずれによっても制限されず、以下の請求項及びそれらの等価物によってのみ画定されるべきである。

【図面の簡単な説明】

【図1】 好適な実施形態による代表的なハードウェア環境を示すブロック図である。

【図1A】 Webポータルネットワークに基づく従来技術のモバイルコンピューティングソリューションを示す図である。

【図2】 好適な実施形態によるシステムのフローチャートである。

30

【図3】 好適な実施形態によるシステムのパーシングユニットのフローチャートである。

【図4】 好適な実施形態によるパターンマッチングに関するフローチャートである。

【図5】 好適な実施形態によるサーチユニットに関するフローチャートである。

【図6】 好適な実施形態による全体的なシステム処理に関するフローチャートである。

【図7】 好適な実施形態によるトピック処理のフローチャートである。

【図8】 好適な実施形態によるミーティングレコード処理のフローチャートである。

【図9】 好適な実施形態によるポケットバーゲンファインダのプロセスフローを示すブロック図である。

【図10A】 好適な実施形態によるカスタマイズコンテンツWebページの作成に関連するロジックを示すブロック図及びフローチャートである。

40

【図10B】 好適な実施形態によるカスタマイズコンテンツWebページの作成に関連するロジックを示すブロック図及びフローチャートである。

【図11】 好適な実施形態によるユーザ中心コンテンツの検索に関連する詳細なロジックを示すフローチャートである。

【図12】 好適な実施形態によるユーザプロフィールのデータモデルを示す図である。

【図13】 好適な実施形態によるペルソナデータモデルを示す図である。

【図14】 好適な実施形態によるインテンションデータモデルを示す図である。

【図15】 好適な実施形態によるエージェントの現行統計の作成に関する処理のフローチャートである。

50

【図 16】 好適な実施形態によるユーザのための個人化された製品評点を決定するロジックのフローチャートである。

【図 17】 好適な実施形態による中央で保存されたプロフィールにアクセスするロジックのフローチャートである。

【図 18】 好適な実施形態によるユーザと特定のサプライヤのインテグレータとの間の対話ロジックのフローチャートである。

【図 19】 好適な実施形態による口頭サマリを生成するエージェント処理のフローチャートである。

【図 20】 好適な実施形態によるディスプレイログインを示す図である。

【図 21】 好適な実施形態による日常ロジスティックス管理ディスプレイを示す図である。

10

【図 22】 好適な実施形態によるユーザメインディスプレイを示す図である。

【図 23】 好適な実施形態によるエージェント対話ディスプレイを示す図である。

【図 24】 好適な実施形態によるアクティブナレッジ管理システムを示すブロック図。

【図 25】 好適な実施形態によるバックエンドサーバを示すブロック図である。

【図 26】 本発明の 1 実施形態のハードウェア及びソフトウェアがどのように動作するかを表すフローチャートである。

【図 27A】 好適な実施形態によるブラウザモードのディスプレイを示す図である。

【図 27B】 好適な実施形態によるモバイルポータルプラットフォームを示す図である。

20

【符号の説明】

- 110 … CPU
- 116 … ROM
- 114 … RAM
- 112 … バス
- 118 … 入出力アダプタ
- 120 … ディスク記憶ユニット
- 124 … キーボード
- 126 … マウス
- 128 … スピーカ
- 132 … マイクロフォン
- 134 … 通信アダプタ
- 135 … ネットワーク
- 136 … ディスプレイアダプタ
- 138 … ディスプレイ装置
- 1010 … 顧客の Web ブラウザ
- 1020 … インテグレータの Web サーバ
- 1030 … 意図データベース
- 1040 … コンテンツデータベース
- 1050 … サプライヤプロフィールデータベース
- 1060 … 顧客プロフィールデータベース
- 1070 … サプライヤの Web サーバ
- 1080 … 製品情報データベース
- 1090 … 製品注文データベース
- 1091 … 顧客情報データベース
- 1710 … プロフィールデータベース
- 1720 … プロフィールゲートウェイサーバ
- 1730 … パブリックインターネット
- 1740 … テレビ
- 1750 … 業者

30

40

50

2 0 0 0、2 2 2 0、2 3 1 0	… エージェント	
2 0 3 0	… ロゴ	
2 0 4 0	… ボタン	
2 1 0 0	… アニメーションエージェント	
2 1 1 0	… 旅行	
2 1 2 0	… 家庭の雑用	
2 1 3 0	… 金融	
2 1 4 0	… 市場活動	
2 1 4 2	… アイコン	
2 1 4 6	… ダイレクトリンク	10
2 1 5 0	… 許可	
2 1 8 0	… 項目	
2 2 0 0	… 世界のニュース	
2 2 1 0	… ローカルニュース	
2 2 3 0	… 不動産	
2 3 0 0	… 情報	
2 3 7 0	… チャート	
2 3 8 0	… ボタン	
2 3 9 0	… ボタンのセット	
2 3 9 5	… ヒント	20
2 4 0 0	… バックエンド	
2 4 1 0、2 4 2 0	… マジックウォール	
2 4 3 0、2 4 3 6	… アウェアネスマシン	
2 4 4 6、2 4 4 8	… Webクライアント	
2 4 5 0、2 4 6 0	… 電子メールクライアント	
2 7 1 0	… モバイルポータルプラットフォーム	
2 7 1 2	… モバイルポータル	
2 7 1 3	… 電子バレット	
2 7 1 4	… シンククライアント	
2 7 1 6	… ハードウェア装置	30
2 7 1 8	… オペレーティングシステム	
2 7 2 0	… 暗号化及び解読要素	
2 7 2 2	… モバイルポータルサーバ	
2 7 2 4	… インテリジェントエージェント	
2 7 2 6	… 顧客インテリジェント要素	
2 7 2 8	… 顧客データベース	
2 7 3 0	… サードパーティコンテンツ及びサービスプロバイダ	
2 7 3 2	… パケット交換無線ネットワーク	

【図 1】

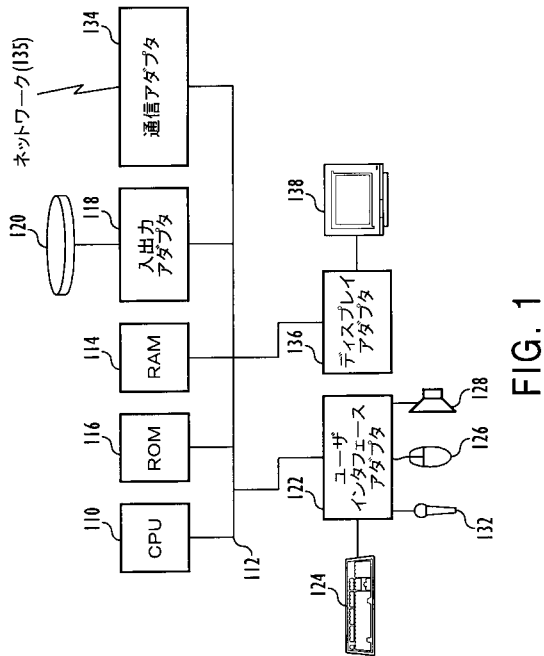
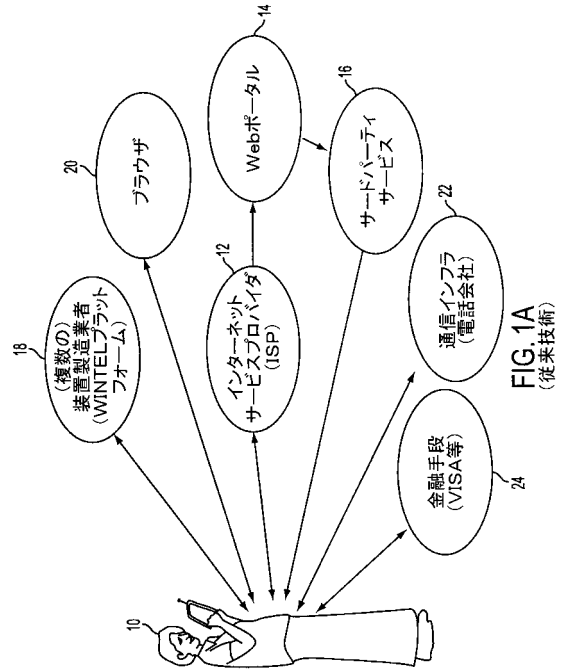


FIG. 1

【図 1 A】

FIG. 1A
(従来技術)

【図 2】

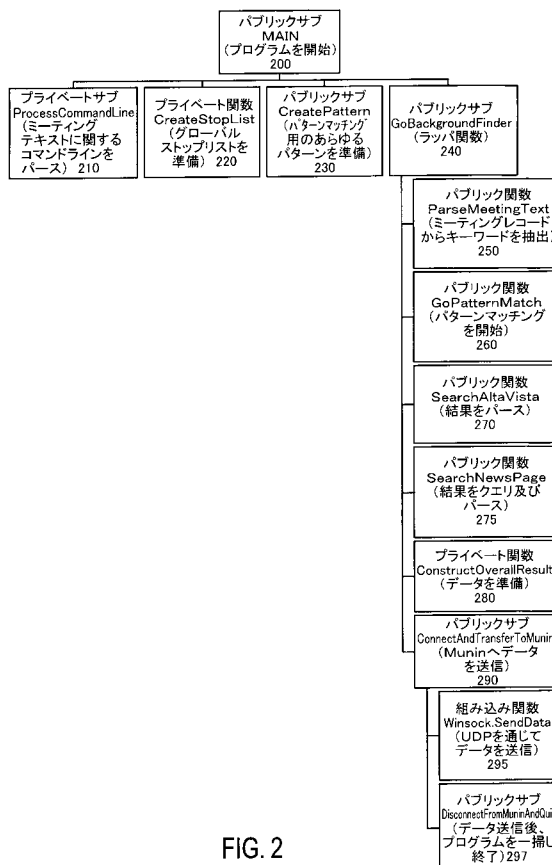


FIG. 2

【図 3】

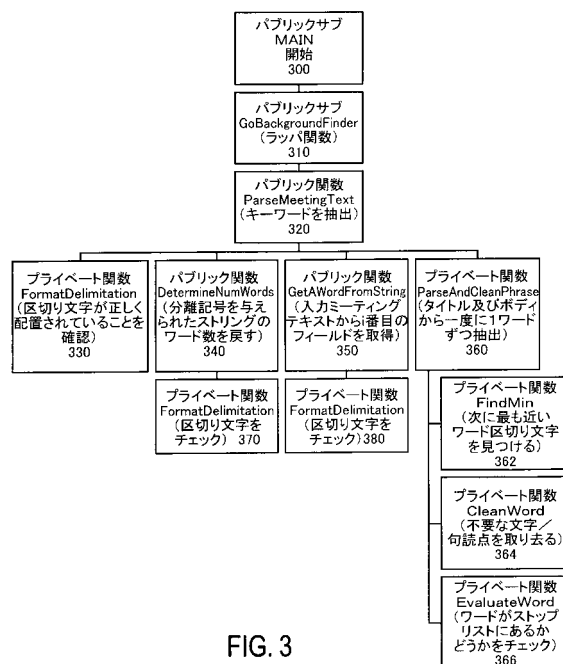


FIG. 3

【図 4】

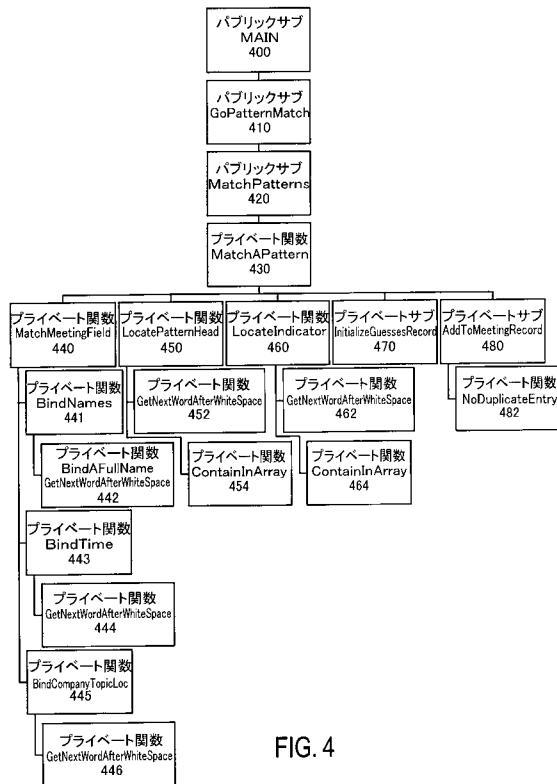


FIG. 4

【図 5】

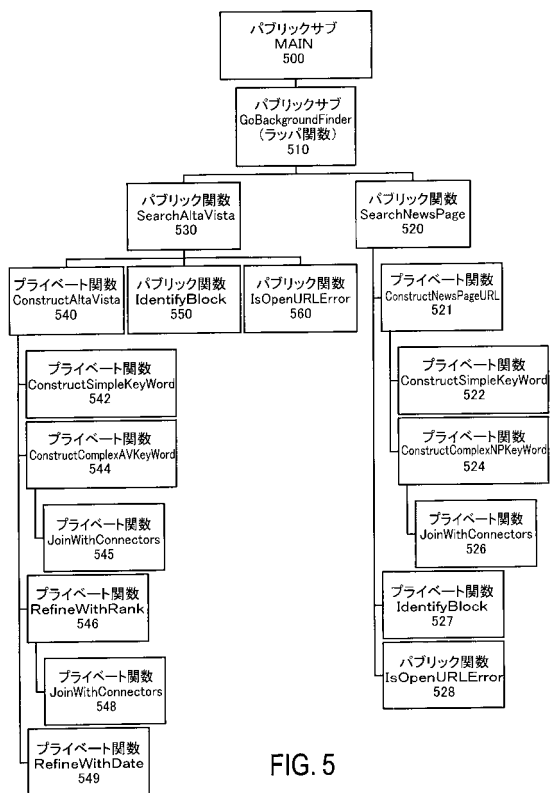


FIG. 5

【図 6】

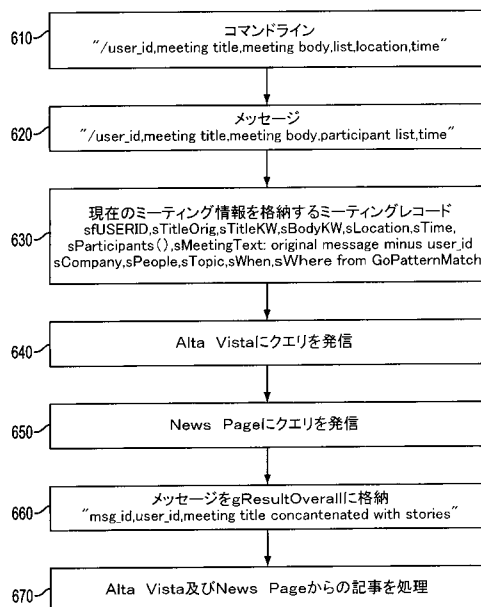


FIG. 6

【図 7】

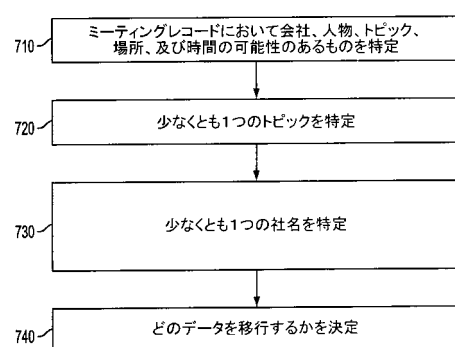


FIG. 7

【図 8】

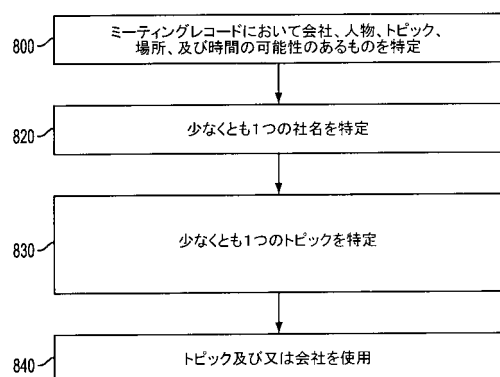


FIG. 8

【図 9】

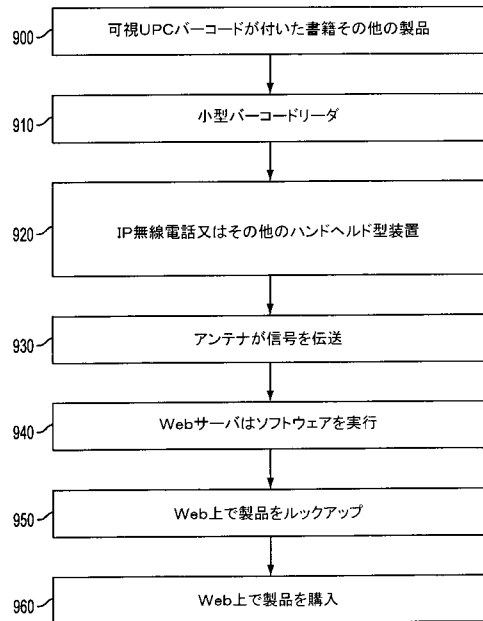


FIG. 9

【図 10 A】

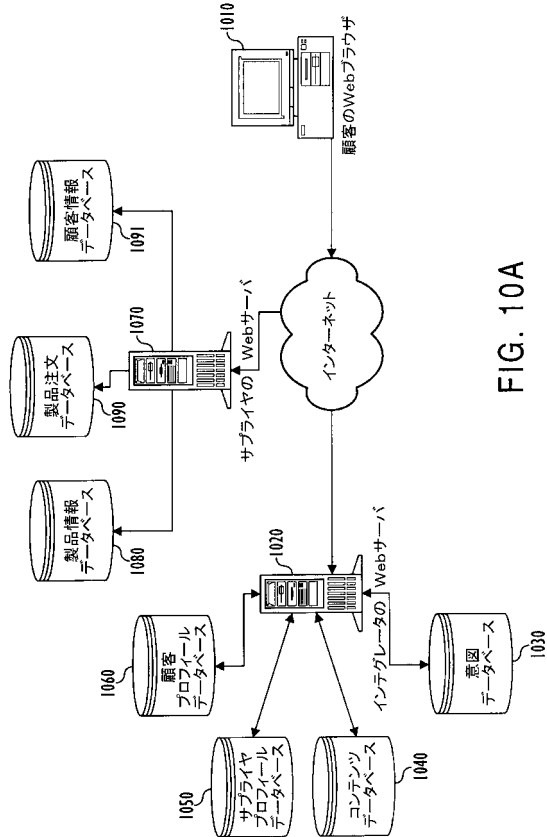


FIG. 10A

【図 10 B】

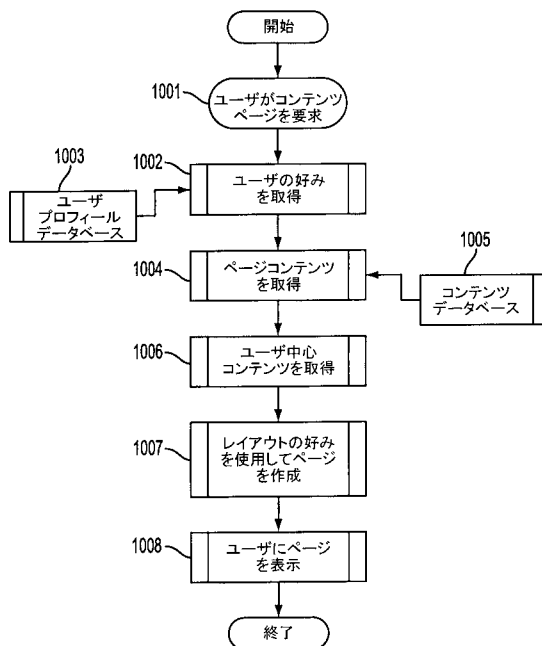


FIG. 10B

【図 11】

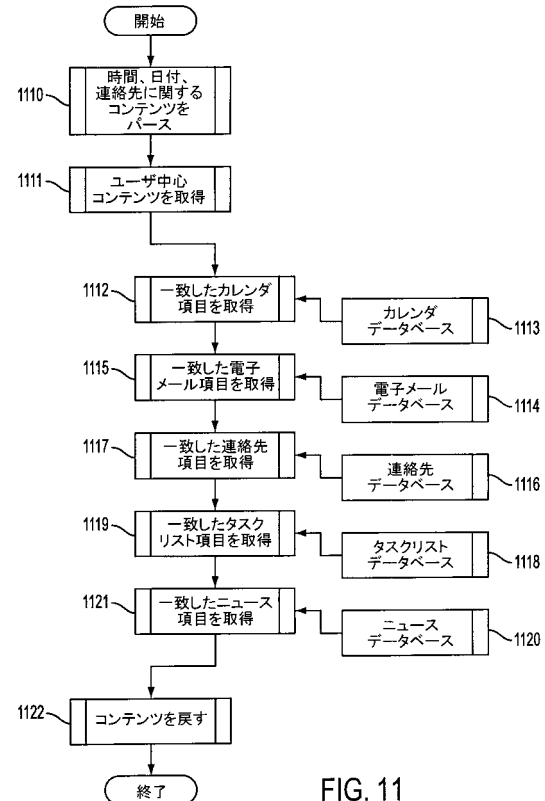


FIG. 11

【図 12】

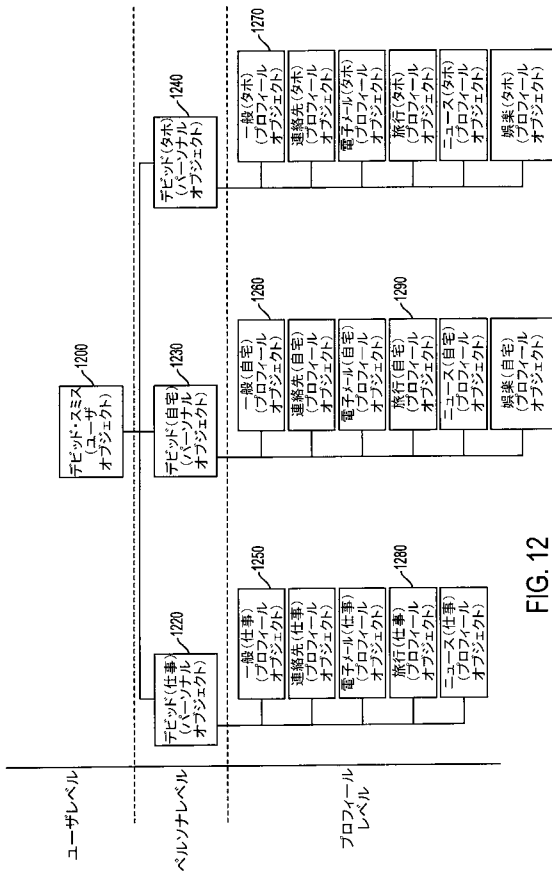


FIG. 12

【図 13】

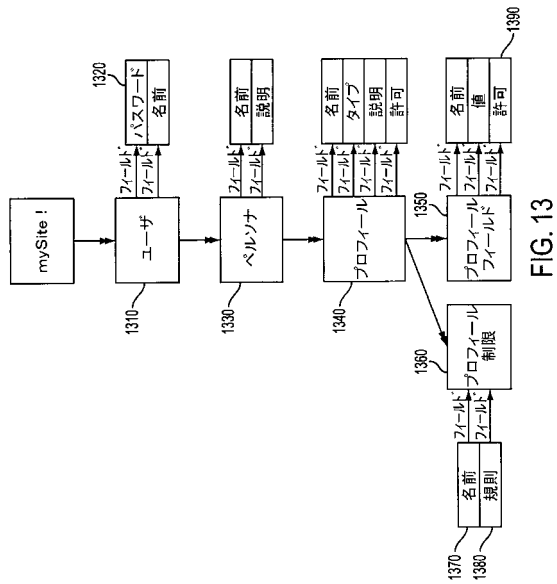


FIG. 13

【図 14】

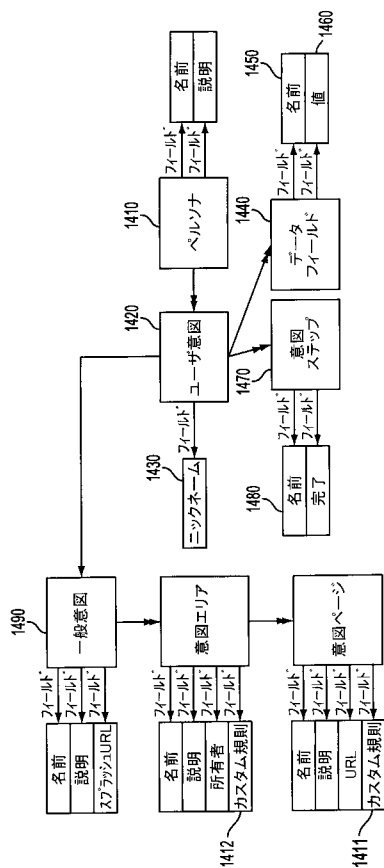


FIG. 14

【図 15】

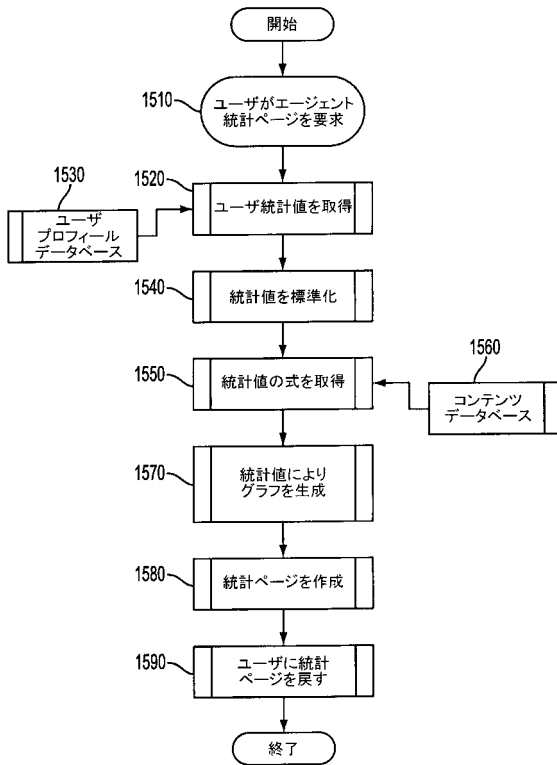


FIG. 15

【図 16】

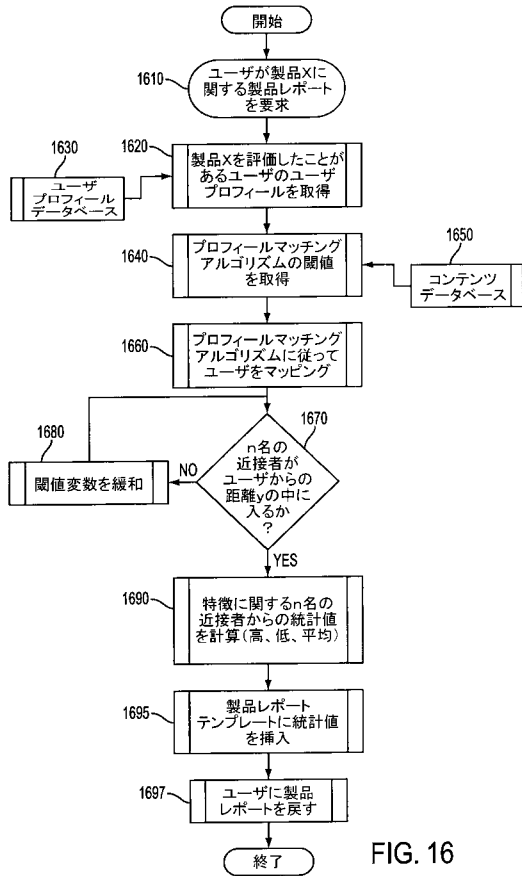


FIG. 16

【図 17】

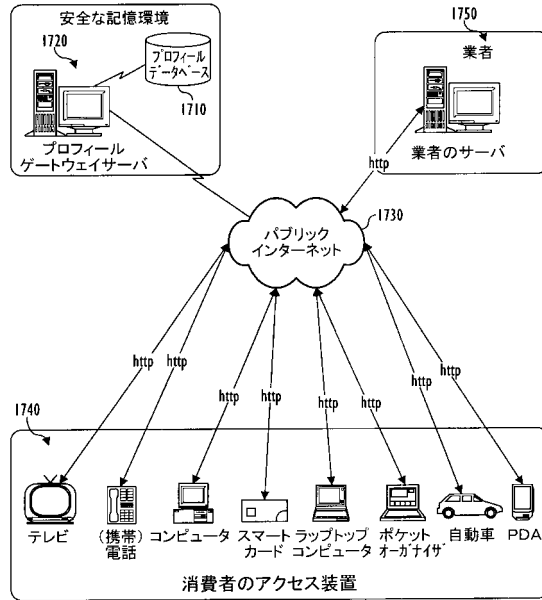


FIG. 17

【図 18】

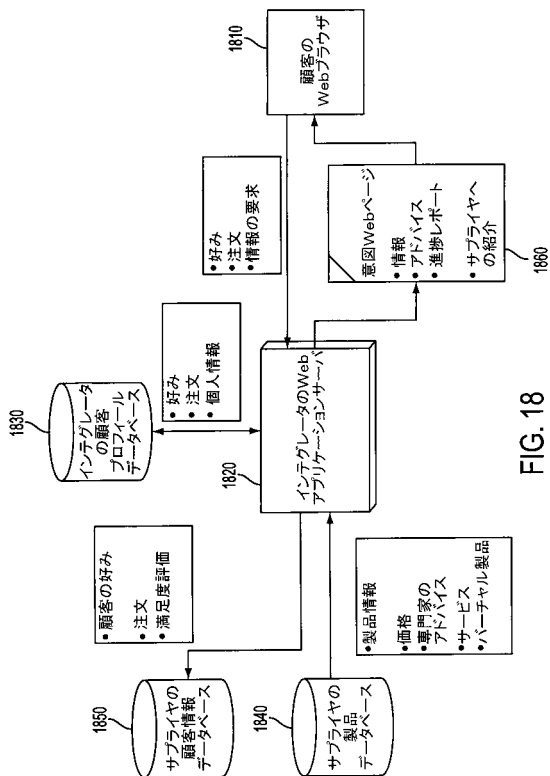


FIG. 18

【図 19】

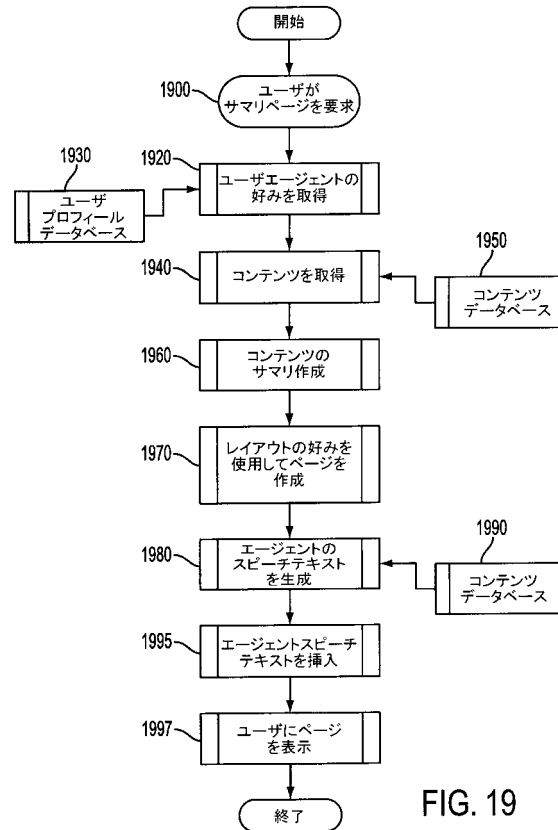


FIG. 19

【 2 0 】

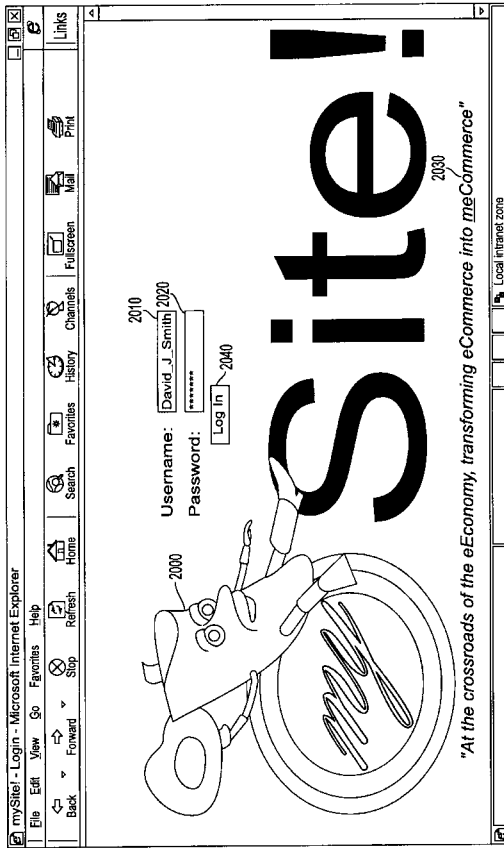


FIG. 20

【 2 1 】

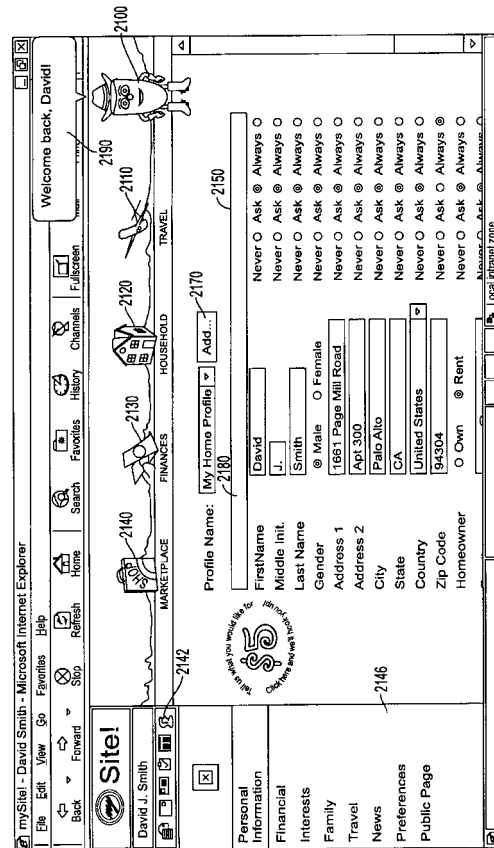


FIG. 21

【 2 2 】

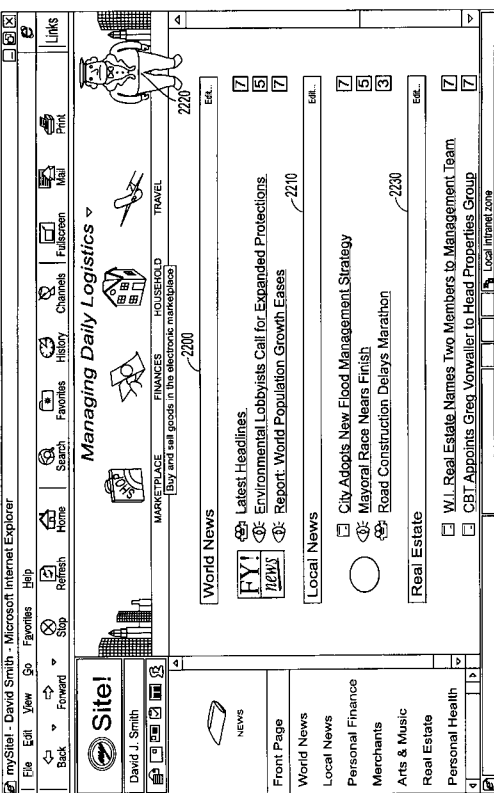


FIG. 22

【 2 3 】

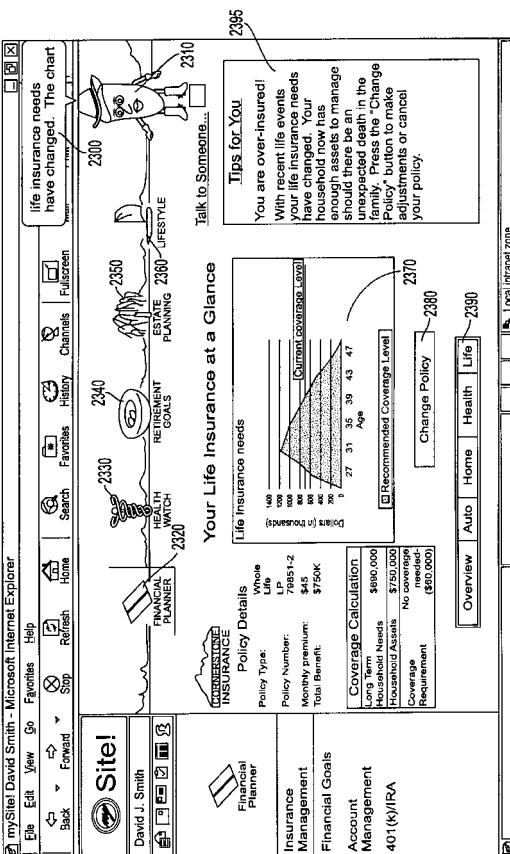


FIG. 23

【図 24】

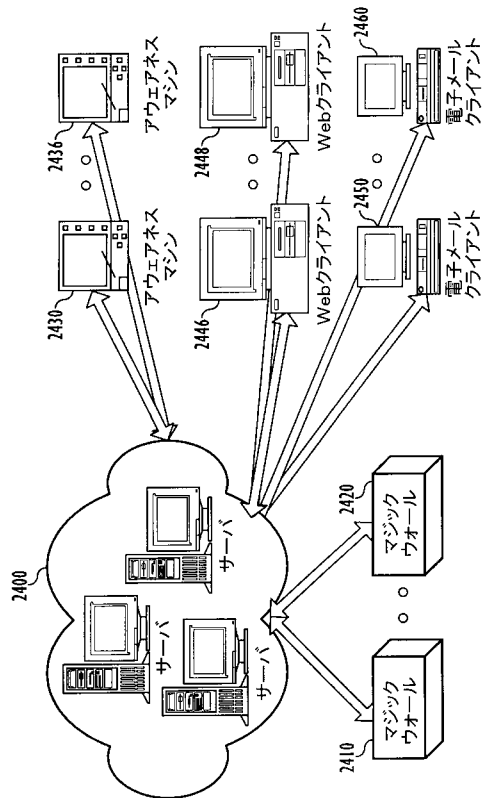


FIG. 24

【図 25】

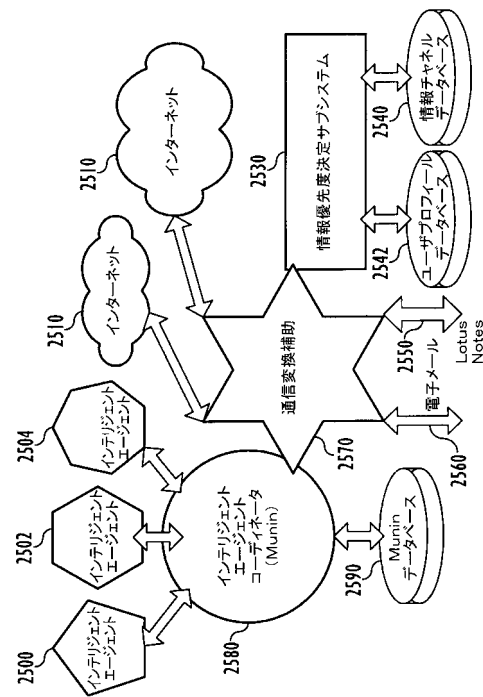


FIG. 25

【図 26】

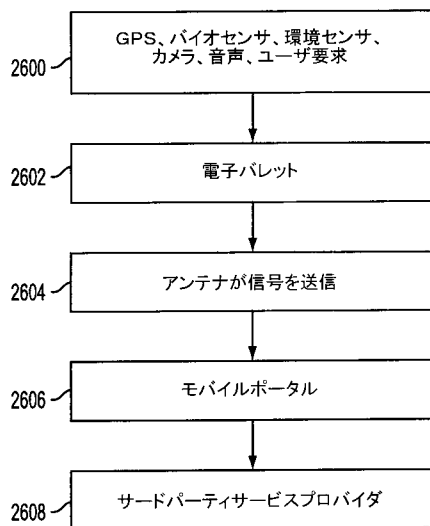


FIG. 26

【図 27 A】

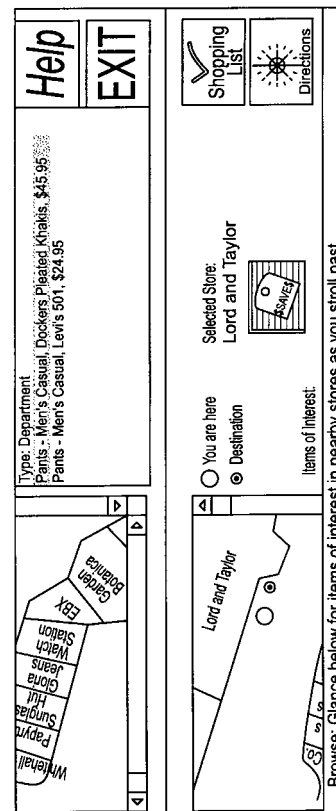


FIG. 27A

フロントページの続き

- (72)発明者 スワミナザン・キショア・サンダラム
アメリカ合衆国 イリノイ州60056 ダウナーズ・グローブ, ステア・ストリート, 6524
- (72)発明者 マイヤーズ・ジェームズ・エル.
アメリカ合衆国 イリノイ州60614 シカゴ, アpartment #1, ウエスト・ライトウッド, 1107
- (72)発明者 ファノ・アンドリュー・イー.
アメリカ合衆国 イリノイ州60202 エバンストン, アpartment・1イー, メープル・アベニュー, 1137

審査官 岩間 直純

- (56)参考文献 国際公開第98/056188 (WO, A1)
特開平10-116295 (JP, A)
Fano A E, SHOPPER'S EYE: Using Location-based Filtering for a Shopping Agent in Physical World, Proceedings of the second international conference on Autonomous agents, 米国, ACM, 1998年, pp.416-421
小山 和也, 外, コミュニケーションツール記述のためのマルチフェースエージェント (MFA) の提案, 情報処理学会研究報告, 日本, 社団法人情報処理学会, 1994年 3月 4日, Vol. 94 No. 19, pp. 69-74
藤野 信次, 外, 移動遍在計算機環境の実現, 電子情報通信学会技術研究報告, 日本, 社団法人電子情報通信学会, 1995年 5月23日, Vol. 95, No. 61, pp. 91-96
FUJINO N, et al., Mobile Information Service Based on Multi-Agent Architecture, IEICE TRANSACTIONS ON COMMUNICATIONS, 日本, IEICE, 1997年10月 1日, V E80-B N10, P1401-1406

(58)調査した分野(Int.Cl., DB名)

G06F 17/30
G06Q 10/00
G06Q 30/00
G06Q 50/00