

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2006-209774

(P2006-209774A)

(43) 公開日 平成18年8月10日(2006.8.10)

(51) Int. Cl.

G06F 9/445 (2006.01)

F I

G06F 9/06 610A

テーマコード (参考)

5B076

審査請求 未請求 請求項の数 20 O L 外国語出願 (全 20 頁)

(21) 出願番号 特願2006-17101 (P2006-17101)
 (22) 出願日 平成18年1月26日 (2006.1.26)
 (31) 優先権主張番号 11/044,354
 (32) 優先日 平成17年1月26日 (2005.1.26)
 (33) 優先権主張国 米国 (US)

(71) 出願人 500046438
 マイクロソフト コーポレーション
 アメリカ合衆国 ワシントン州 9805
 2-6399 レッドモンド ワン マイ
 クロソフト ウェイ
 (74) 代理人 100077481
 弁理士 谷 義一
 (74) 代理人 100088915
 弁理士 阿部 和夫
 (72) 発明者 クリストファー マイケル ハーン
 アメリカ合衆国 98052 ワシントン
 州 レッドモンド ワン マイクロソフト
 ウェイ マイクロソフト コーポレーシ
 ョン内

最終頁に続く

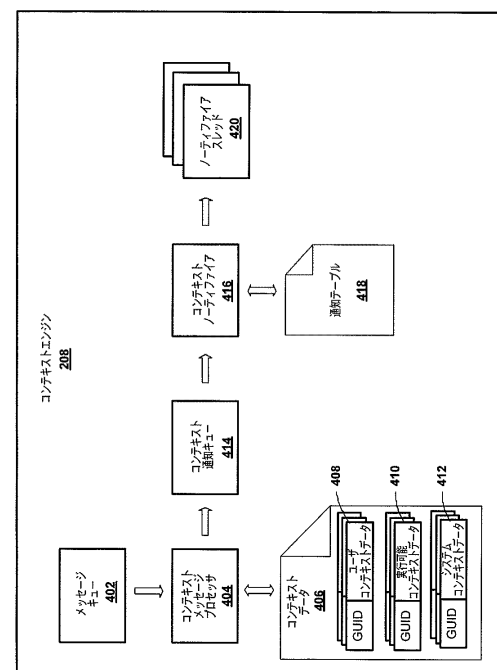
(54) 【発明の名称】 コンテキストウェアネスプラットフォームのためのシステムおよび方法

(57) 【要約】 (修正有)

【課題】 実行環境を適応させるために使用することができるコンテキストウェアネスプラットフォームのためのシステムおよび方法を提供する。

【解決手段】 フレームワークはコンピュータシステムの実行環境についてのコンテキスト情報を設定し、追跡し、取得するインターフェースを備える。フレームワークはプラットフォーム上で稼動している実行可能プログラムによってアクセスするためのコンテキスト情報をプラットフォームレベルで集約するためのコンテキストエンジン208、および統合のデータストアを含む。コンテキストエンジン208は、データストア内のコンテキスト情報の設定、追跡、および取得など、コンテキスト情報オペレーションを実行する旨の要求を実行可能ファイルから受信する。コンテキスト情報の変更の通知を受信すると、1つまたは複数の実行可能プログラムは、システム設定を再構成するなど、実行環境を変える。

【選択図】 図4



【特許請求の範囲】**【請求項 1】**

実行可能プログラムにコンテキスト情報を提供するコンピュータシステムであって、
実行可能ソフトウェアコードと、

前記実行可能ソフトウェアコードに動作可能に結合されたコンテキスト構成要素であって、コンテキスト情報オペレーションを要求するために実行可能ソフトウェアコードによって呼び出し可能なアプリケーションプログラミングインターフェースを有するコンテキスト構成要素と、

前記コンテキスト構成要素に動作可能に結合されたコンテキストエンジンであって、前記実行可能ソフトウェアコードによって要求された前記コンテキスト情報オペレーションを実行するコンテキストエンジンと、

前記コンテキストエンジンに動作可能に結合された統合のデータストアであって、コンテキスト情報を永続的に格納する統合のデータストアと

を含むことを特徴とするコンピュータシステム。

【請求項 2】

コンテキスト情報オペレーションを要求するために前記実行可能ソフトウェアコードによって呼び出し可能な前記アプリケーションプログラミングインターフェースは、コンテキスト情報を設定するよう要求するアプリケーションプログラミングインターフェースを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 3】

コンテキスト情報オペレーションを要求するために前記実行可能ソフトウェアコードによって呼び出し可能な前記アプリケーションプログラミングインターフェースは、コンテキスト情報を取得するよう要求するアプリケーションプログラミングインターフェースを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 4】

コンテキスト情報オペレーションを要求するために前記実行可能ソフトウェアコードによって呼び出し可能な前記アプリケーションプログラミングインターフェースは、コンテキスト情報の変更を追跡するよう要求するアプリケーションプログラミングインターフェースを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 5】

前記コンテキストエンジンは、前記実行可能ソフトウェアコードによって要求された前記コンテキスト情報オペレーションを実行するコンテキストメッセージプロセッサを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 6】

前記コンテキストエンジンは、通知メッセージを前記コンテキスト情報とともに前記実行可能ソフトウェアコードに送信するコンテキストノーティファイアを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 7】

前記コンテキストエンジンは、通知メッセージを前記コンテキスト情報とともに前記実行可能ソフトウェアコードに非同期で送信するノーティファイアスレッドを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 8】

前記統合のデータストアは、ユーザアクティビティ、アプリケーション状態、およびマシン状態のコンテキスト情報を永続的に格納するデータストアを含むことを特徴とする請求項 1 に記載のシステム。

【請求項 9】

請求項 1 に記載の前記システムを含むコンピュータ実行可能構成要素を有することを特徴とするコンピュータ可読媒体。

【請求項 10】

実行可能ソフトウェアコードにコンテキスト情報を提供する方法であって、

10

20

30

40

50

実行可能ソフトウェアコードのコンテキスト情報を設定するアプリケーションプログラミングインターフェースを呼び出すステップと、

前記実行可能ソフトウェアコードの前記コンテキスト情報を設定するステップと、

前記実行可能ソフトウェアコードの前記コンテキスト情報を取得するアプリケーションプログラミングインターフェースを呼び出すステップと、

ユーザアクティビティ、アプリケーション状態、およびマシン状態のコンテキスト情報を含む統合のデータストアから前記実行可能ソフトウェアコードの前記コンテキスト情報を取得するステップと

を含むことを特徴とする方法。

【請求項 11】

前記実行可能ソフトウェアコードのコンテキスト情報を設定する旨の要求を受信するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 12】

前記実行可能ソフトウェアコードのコンテキスト情報を追跡する旨の要求を受信するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 13】

前記実行可能ソフトウェアコードのコンテキスト情報を取得する旨の要求を受信するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 14】

前記コンテキスト情報の更新を実行可能ソフトウェアコードに通知するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 15】

通知メッセージを前記コンテキスト情報とともに前記実行可能ソフトウェアコードに非同期で送信する通知スレッドを作成するステップをさらに含むことを特徴とする請求項 14 に記載の方法。

【請求項 16】

前記コンテキスト情報を、ユーザアクティビティ、アプリケーション状態、およびマシン状態のコンテキスト情報を含む統合のデータストアに永続的に格納するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 17】

前記実行可能ソフトウェアコードの命令を実行するために取得された前記コンテキスト情報を使用して、前記実行可能ソフトウェアコードの実行環境を変更するステップをさらに含むことを特徴とする請求項 10 に記載の方法。

【請求項 18】

請求項 10 に記載の前記方法を実行するコンピュータ実行可能命令を有することを特徴とするコンピュータ可読媒体。

【請求項 19】

実行可能プログラムにコンテキスト情報を提供するコンピュータシステムであって、コンテキスト情報オペレーションを実行するためにアプリケーションプログラミングインターフェースを呼び出す手段と、

前記コンテキスト情報オペレーションを実行する手段と、

実行可能ソフトウェアコードの前記コンテキスト情報を取得するアプリケーションプログラミングインターフェースを呼び出す手段と、

前記実行可能ソフトウェアコードの前記コンテキスト情報を取得する手段と

を含むことを特徴とするコンピュータシステム。

【請求項 20】

ユーザアクティビティ、前記実行可能ソフトウェアコードの状態、および前記コンピュータシステムの状態のコンテキスト情報を含む統合のデータストア内に前記コンテキスト情報を永続的に格納する手段をさらに含むことを特徴とする請求項 19 に記載のシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は一般にコンピュータシステムに関し、より詳細には、実行環境を適応させるために使用することができるコンテキストウェアネスプラットフォームの改良されたシステムおよび方法に関する。

【背景技術】

【0002】

コンピュータユーザは、コンピュータシステムが新しいまたは変更されたコンピューティング環境および/または動作環境に知的に適応することができないことで、長い間苛立ちを感じていた。この苛立ちは、しばしば複数の物理的な場所で、また様々な環境下で異なる大量のアクティビティを処理する可能性のあるモバイルコンピュータのユーザにとってはなおさら深刻となる。例えば、1日の初めに、モバイルコンピュータのユーザは、家にいて、電子メールやインスタントメッセージングを使用して、家族や友達と対話するかもしれない。次いでその後、モバイルコンピュータのユーザは、職場で会議中にプレゼンテーションを行うかもしれない。モバイルコンピュータのユーザは、夜に、翌日の商談のために飛行機で移動しながら取引申込書を作成しているかもしれない。

【発明の開示】

【発明が解決しようとする課題】

【0003】

残念ながら、ある環境での構成は別の環境ではすでに適切ではなく、変更する必要がある場合でさえ、コンピュータは、こうした異なる環境および使用ごとに同じように構成されたままである場合がある。例えば、ユーザは、家のネットワークと職場のネットワークとの間を移動する場合、物理的な場所を変えるようにモバイルコンピュータを構成する必要がある場合がある。この場合、ユーザは、デフォルトのプリンタ設定、デフォルトのインターネットブラウザホームページ、インターネットブラウザプロキシ設定、および他の関連の設定を変更する必要がある場合がある。また、ユーザは、アクティビティを変更するようコンピュータを構成する必要がある場合もある。例えば、プレゼンテーションを行う準備をしているユーザは、コンピュータのボリュームを調整し、インスタントメッセージ通知をオフにし、スクリーンセーバのタイムアウトをオフにし、他の設定調整を行うかもしれない。また、ユーザは、職場では家で対話する人とは異なる人と対話するようコンピュータを構成する必要がある場合もある。例えば、家では、友人や家族と通信するために、職場で使用する電子メールアカウントとは異なるデフォルトの電子メールアカウントを使用するかもしれない。その結果、モバイルコンピュータのユーザは、コンピューティング環境が変わるたびにモバイルコンピュータを再構成する必要がある場合がある。

【0004】

コンテキスト情報を使用してコンピュータをコンピューティング環境に知的に適応させることができるように、コンピュータシステムが、物理的な場所、その物理的な場所でユーザが対話する可能性のある人物、ユーザのアクティビティなど、コンピューティング環境に関するコンテキスト情報を収集し、提示する方法が必要である。こうしたシステムおよび方法は、任意の実行可能プログラム(executable)がコンテキスト情報を提供し、または使用して、コンピュータシステムを知的に適応させるように、均一の体系的な方法でコンテキスト情報を収集し、提示するべきである。

【課題を解決するための手段】

【0005】

簡潔に言えば、本発明は、実行環境を適応させるために使用することができるコンテキストウェアネスプラットフォームの改良されたシステムおよび方法を提供する。このため、本発明は、コンピュータシステムの実行環境についてのコンテキスト情報を設定し、追跡し、取得するインターフェースを備える構成要素およびフレームワークを提供する。提供される構成要素の中には、例えば、様々な実行可能プログラム、および統合のデータ

10

20

30

40

50

ストアを含むデータストレージ設備に動作可能に結合されるコンテキストプラットフォームがある。データストア（データ記憶装置）は、プラットフォーム上で稼動する実行可能プログラムによってアクセスするためのコンテキスト情報の集約をプラットフォームレベルで行うことによって、コンテキストプラットフォームのコンテキスト情報の統合されたストレージを提供する。それだけには限定されないが、ユーザコンテキストデータ、実行可能コンテキストデータ、およびシステムコンテキストデータを含めて、任意のタイプのコンテキスト情報を（データストアに）格納することができる。

【 0 0 0 6 】

コンテキストプラットフォームは、コンテキストプラットフォームの様々な部分と他の実行可能プログラムとの間のプログラムに基づいた対話およびデータ交換を調整するために、コンテキストアプリケーションプログラミングインターフェース、および以下コンテキストエンジンと呼ぶ実行可能コードを含む。コンテキストアプリケーションプログラミングインターフェースは、実行可能プログラムがコンテキスト情報を設定し、追跡し、または取得するために、アクセスを提供する。コンテキスト情報を設定し、追跡し、または取得するために、任意の実行可能プログラムを、コンテキストアプリケーションプログラミングインターフェースに動作可能に結合する。コンテキストエンジンは、データストア内のコンテキスト情報の設定、追跡、および取得など、コンテキスト情報オペレーションを実行する旨の要求を実行可能プログラムから受信する。例えば、コンテキストエンジンは、コンテキスト情報を更新する旨の要求を受信したり、コンテキスト情報の変更について通知されるよう要求した実行可能プログラムを探したり、コンテキスト情報の変更の通知を実行可能プログラムに送信する。コンテキスト情報の変更の通知を受信した結果、1つまたは複数の実行可能プログラムは、システム設定を再構成するなど、実行環境を変えることができる。このように、実行可能プログラムは、コンテキスト情報を使用して、その実行環境を適応させる。

【 0 0 0 7 】

コンテキストプラットフォームは、コンテキスト情報を使用してコンピュータを実行環境に知的に適応させることができるように、コンピュータシステムが、物理的な場所、その物理的な場所でユーザが対話する可能性のある人物、ユーザのアクティビティなど、実行環境に関するコンテキスト情報を有利に収集し、提示する。さらに、コンテキストプラットフォームは、拡張可能であり、コンテキスト情報を提供し、使用し得る任意の数の実行可能プログラムをサポートする。さらに、コンテキストプラットフォームは、任意のサイズおよび/またはフォーマットのデータのコンテキスト情報をサポートする。他の利点は、以下の詳細な説明を図面と併せ読めば、理解できよう。

【 発明を実施するための最良の形態 】

【 0 0 0 8 】

動作環境の例

図1は、本発明を実施可能な適したコンピューティングシステム環境100の例を示している。コンピューティングシステム環境100は、適したコンピューティング環境の一例にすぎず、本発明の使用または機能の範囲に関する限定を示唆するものではない。また、コンピューティング環境100を、動作環境100の例に示した構成要素のいずれか1つ、またはその組合せに関連する任意の依存性または必要条件を有していると解釈すべきではない。

【 0 0 0 9 】

本発明は、他の多くの汎用または専用コンピューティングシステム環境または構成で動作可能である。本発明との使用に適したよく知られているコンピューティングシステム、環境、および/または構成の例には、それだけには限定されないが、パーソナルコンピュータ、サーバコンピュータ、ハンドヘルドまたはラップトップ装置、タブレット装置、ヘッドレスサーバ、マルチプロセッサシステム、マイクロプロセッサベースのシステム、セットトップボックス、プログラム可能家庭用電化製品、ネットワークPC、ミニコンピュータ、メインフレームコンピュータ、上記の任意のシステムまたは装置を含む分散コンピ

ューティング環境などがある。

【0010】

本発明は、コンピュータによって実行されるプログラムモジュールなどのコンピュータ実行可能命令の一般的な文脈で説明することができる。一般にプログラムモジュールは、特定のタスクを実行する、または特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、構成要素、データ構造などを含む。また、本発明は、タスクが通信ネットワークによってリンクされているリモート処理装置によって実行される分散コンピューティング環境でも実施することができる。分散コンピューティング環境では、プログラムモジュールを、メモリ記憶装置を含むローカルおよび/またはリモートのコンピュータ記憶媒体に置くことができる。

10

【0011】

図1を参照すると、本発明を実施するシステムの例は、汎用コンピューティング装置をコンピュータ110の形で含んでいる。コンピュータ110の構成要素は、それだけには限定されないが、処理ユニット120、システムメモリ130、およびシステムメモリを含む様々なシステム構成要素を処理ユニット120に結合するシステムバス121を含む。システムバス121は、様々な任意のバスアーキテクチャを使用するメモリバスまたはメモリコントローラ、周辺バス、およびローカルバスを含むいくつかのタイプのバス構造のうちどんなバス構造でもよい。こうしたアーキテクチャには、それだけには限定されないが一例として、業界標準アーキテクチャ（ISA）バス、マイクロチャンネルアーキテクチャ（MCA）バス、拡張ISA（EISA）バス、ビデオ電子装置規格化協会（VESA）ローカルバス、およびメザニンバスとしても知られている周辺部品相互接続（PCI）バスなどがある。

20

【0012】

コンピュータ110は、一般に様々なコンピュータ可読媒体を含む。コンピュータ可読媒体は、コンピュータ110からアクセス使用可能な任意の媒体とすることができ、揮発性および不揮発性媒体、取外式および固定式の媒体を含む。コンピュータ可読媒体は、それだけには限定されないが一例として、コンピュータ記憶媒体および通信媒体を含む。コンピュータ記憶媒体には、コンピュータ可読命令、データ構造、プログラムモジュール、他のデータなど、情報を記憶するための任意の方法または技術で実施される揮発性および不揮発性の取外式および固定式媒体がある。コンピュータ記憶媒体には、それだけには限定されないが、RAM、ROM、EEPROM、フラッシュメモリまたは他のメモリ技術、CD-ROM、デジタル多用途ディスク（DVD）または他の光ディスク記憶装置、磁気カセット、磁気テープ、磁気ディスク記憶装置または他の磁気記憶装置、または所望の情報の格納に使用でき、コンピュータ110からアクセスできる他の任意の媒体などがある。通信媒体は一般に、コンピュータ可読命令、データ構造、プログラムモジュール、または他のデータを搬送波または他の移送機構などの変調されたデータ信号に組み込む。これには任意の情報配送媒体がある。「変調されたデータ信号」という用語は、信号に情報を符号化するように1つまたは複数のその特性が設定または変更された信号を意味する。通信媒体には、それだけには限定されないが一例として、有線ネットワーク、直接配線された接続などの有線媒体、および音響、RF、赤外線、その他の無線媒体などの無線媒体

30

40

【0013】

システムメモリ130は、読み取り専用メモリ（ROM）131やランダムアクセスメモリ（RAM）132など、揮発性および/または不揮発性メモリの形のコンピュータ記憶媒体を含む。基本入出力システム133（BIOS）は、例えば起動中など、コンピュータ110内の要素間での情報の転送を助ける基本ルーチンを含み、一般にROM131に格納されている。RAM132は一般に、処理ユニット120から直接アクセス可能な、および/または処理ユニット120が現在処理しているデータおよび/またはプログラムモジュールを含む。図1は、それだけには限定されないが一例として、オペレーティン

50

グシステム 1 3 4、アプリケーションプログラム 1 3 5、他のプログラムモジュール 1 3 6、およびプログラムデータ 1 3 7を示している。

【 0 0 1 4 】

コンピュータ 1 1 0 は、他の取外式 / 固定式、揮発性 / 不揮発性コンピュータ記憶媒体を含むこともできる。一例にすぎないが、図 1 は、固定式不揮発性磁気媒体から読み取り、あるいはそこに書き込むハードディスクドライブ 1 4 1、取外式不揮発性磁気ディスク 1 5 2 から読み取り、あるいはそこに書き込む磁気ディスクドライブ 1 5 1、および C D - R O M や他の光媒体など、取外式不揮発性光ディスク 1 5 6 から読み取り、あるいはそこに書き込む光ディスクドライブ 1 5 5 を示している。動作環境の例で使用する他の取外式 / 固定式、揮発性 / 不揮発性コンピュータ記憶媒体には、それだけには限定されないが、磁気テープカセット、フラッシュメモリカード、デジタル多用途ディスク、デジタルビデオテープ、半導体 R A M、半導体 R O M などがある。ハードディスクドライブ 1 4 1 は一般に、インターフェース 1 4 0 などの固定式メモリアインターフェースを介してシステムバス 1 2 1 に接続され、磁気ディスクドライブ 1 5 1 および光ディスクドライブ 1 5 5 は一般に、インターフェース 1 5 0 などの取外式メモリアインターフェースによってシステムバス 1 2 1 に接続される。

10

【 0 0 1 5 】

上述し、図 1 に示したドライブおよびその関連のコンピュータ記憶媒体は、コンピュータ可読命令、データ構造、プログラムモジュール、およびコンピュータ 1 1 0 の他のデータの記憶域を提供する。図 1 では例えば、ハードディスクドライブ 1 4 1 は、オペレーティングシステム 1 4 4、アプリケーションプログラム 1 4 5、他のプログラムモジュール 1 4 6、およびプログラムデータ 1 4 7 を記憶するように示されている。これらの構成要素は、オペレーティングシステム 1 3 4、アプリケーションプログラム 1 3 5、他のプログラムモジュール 1 3 6、およびプログラムデータ 1 3 7 と同じであっても、異なってもよいことに留意されたい。オペレーティングシステム 1 4 4、アプリケーションプログラム 1 4 5、他のプログラムモジュール 1 4 6、およびプログラムデータ 1 4 7 は少なくとも異なるコピーであることを示すために、ここではそれらに異なる番号を付している。ユーザは、タブレットまたは電子デジタイザ 1 6 4、マイクロフォン 1 6 3、キーボード 1 6 2、および一般にマウス、トラックボール、またはタッチパッドと呼ばれるポインティング装置 1 6 1 などの入力装置を介してコマンドおよび情報をコンピュータ 1 1 0 に入力することができる。図 1 には示していない他の入力装置には、ジョイスティック、ゲームパッド、衛星パラボラアンテナ、スキャナ、またはバイオメトリクスセンサ、環境センサ、位置センサ、または他のタイプのセンサを含む装置を含む他の装置などがある。これらおよび他の入力装置は、しばしばシステムバスに結合されているユーザ入力インターフェース 1 6 0 を介して処理ユニット 1 2 0 に接続されるが、パラレルポート、ゲームポート、ユニバーサルシリアルバス (U S B) など他のインターフェースおよびバス構造で接続してもよい。モニタ 1 9 1 または他のタイプの表示装置もまた、ビデオインターフェース 1 9 0 などのインターフェースを介してシステムバス 1 2 1 に接続される。モニタ 1 9 1 は、タッチ画面インターフェース 1 9 2 を介してシステムバス 1 2 1 に接続されるタッチ画面パネルなどに一体化することもできる。モニタおよび / またはタッチ画面パネルは、タブレット型パーソナルコンピュータなど、コンピューティング装置 1 1 0 が組み込まれるハウジングに物理的に結合することができることに留意されたい。さらに、コンピューティング装置 1 1 0 などのコンピュータは、出力周辺インターフェース 1 9 3 などを介して接続できるスピーカ 1 9 4、プリンタ 1 9 5 などの他の周辺出力装置を含むこともできる。

20

30

40

【 0 0 1 6 】

コンピュータ 1 1 0 は、リモートコンピュータ 1 8 0 など 1 つまたは複数のリモートコンピュータへの論理接続を使用してネットワーク式環境で動作することができる。リモートコンピュータ 1 8 0 は、パーソナルコンピュータ、サーバ、ルータ、ネットワーク P C、ピア装置、または他の一般のネットワークノードでよく、一般にコンピュータ 1 1 0 に

50

関連して上述した多くまたはすべての要素を含むが、図 1 にはメモリ記憶装置 181 のみを示している。図 1 に示した論理接続は、ローカルエリアネットワーク (LAN) 171 および広域ネットワーク (WAN) 173 を含むが、他のネットワークを含んでいてもよい。こうしたネットワーキング環境は、オフィス、全社規模のコンピュータネットワーク、イントラネット、およびインターネットではごく一般的である。LAN ネットワーキング環境で使用する場合、コンピュータ 110 は、ネットワークインターフェースまたはアダプタ 170 を介して LAN 171 に接続される。WAN ネットワーキング環境で使用する場合、コンピュータ 110 は一般に、モデム 172、またはインターネットなど WAN 173 を介して通信を確立する他の手段を含む。モデム 172 は、内蔵でも外付けでもよく、ユーザ入力インターフェース 160 または他の適切な機構を介してシステムバス 121 に接続することができる。ネットワーク式環境では、コンピュータ 110 に関連して示したプログラムモジュール、またはその一部をリモートメモリ記憶装置に格納することができる。図 1 は、それだけには限定されないが一例として、リモートアプリケーションプログラム 185 をメモリ装置 181 上に存在するように示している。図示したネットワーク接続は例であり、コンピュータ間の通信リンクを確立する他の手段を使用してもよいことは理解されよう。

10

【0017】

コンテキストウェアネスプラットフォーム

本発明は一般に、コンテキストウェアネスプラットフォームのシステムおよび方法を対象とする。コンテキストウェアネスプラットフォームは、コンテキスト情報を使用してコンピュータを実行環境に知的に適応させることができるように、コンピュータシステムが、物理的な場所、その物理的な場所でユーザが対話する可能性のある人物、ユーザのアクティビティなど、実行環境に関するコンテキスト情報を有利に収集し、提示できるようにする。実行環境は、本明細書で使用する場合、インストールまたは構成の設定、実行可能プログラム命令、オペレーティングシステムサービスなど、実行可能設定を意味する。本発明のアーキテクチャは、コンピュータシステムの実行環境についてのコンテキスト情報を設定し、追跡し、取得するインターフェースを備える構成要素およびフレームワークを提供することがわかる。提供される構成要素の中には、プラットフォーム上で稼動する実行可能プログラムによってアクセスするためのコンテキスト情報の集約をプラットフォームレベルで行うコンテキストエンジンおよび統合のデータストアがある。ユーザコンテキストデータ、実行可能コンテキストデータ、およびシステムコンテキストデータを含めて、任意のタイプのコンテキスト情報を格納することができる。本明細書に示した様々なブロック図、フローチャート、およびシナリオは例にすぎず、本発明を適用するシナリオが他にも多数あり得ることは理解されよう。

20

30

【0018】

図 2 を参照すると、コンテキストウェアネスプラットフォームのシステム構成要素のアーキテクチャ例の概略を表すブロック図を示している。図面に示したブロック内で実施される機能を個別の構成要素として実施したり、いくつかのまたはすべてのブロックの機能を単一の構成要素内で実施したりすることができることを当業者であれば理解されよう。一例として、コンテキストプラットフォーム 204 内のコンテキスト API 206 の機能は、個別の構成要素で実施することができる。

40

【0019】

図 2 に示したコンテキストプラットフォーム 204 は、様々な実行可能プログラム 202、およびデータストア 210 などのストレージ設備に動作可能に結合することができる。コンテキストプラットフォーム 204 は、コンテキストアプリケーションプログラミングインターフェース (API) 206 およびコンテキストエンジン 208 を含む。コンテキスト API 206 は、アプリケーションプログラミングインターフェース (API)、オブジェクト、コントロールまたは他の実行可能コードとすることができる。コンテキスト API 206 は、例えば、データストア 210 にコンテキスト情報を追加する Set Context Data、データストア 210 内のコンテキスト情報の更新が通知される T

50

rackContextData、データストア210内のコンテキスト情報を取り出すGetContextDataなど、呼び出し可能なインターフェースを含めることによって、コンテキスト情報を設定し、追跡し、または取得するアクセスを提供することができる。コンテキスト情報を操作（コンテキスト情報の情報処理の意、ここではコンテキスト情報のオペレーションを実行すること）するために、他のコンテキストAPIを使用することもできる。

【0020】

コンテキストエンジン208は、カーネル構成要素、アプリケーション構成要素、リンクライブラリの構成要素、オブジェクトなどを含む、任意の実行可能ソフトウェアコードとすることができる。コンテキストエンジン208は、データストア210内のコンテキスト情報の設定、追跡、および取得など、コンテキスト情報オペレーションを実行する旨の要求を実行可能ソフトウェアコードから受信することができる。例えば、コンテキストエンジン208は、コンテキスト情報の更新または変更が通知される旨の要求を追跡し、追跡されるよう要求したコンテキスト情報の更新または変更を受信すると、実行可能プログラムに通知することができる。コンテキスト情報を永続的に格納するために、コンテキストエンジン208を、データストア210などの様々なストレージ設備に動作可能に結合することができる。コンテキストエンジンは、コンテキスト情報を格納するために使用することができる、コンテキスト情報を表すコンテキストデータのブロックに、GUIDなどの一意の識別子を関連付けることができる。

【0021】

データストア210は、プラットフォーム上で稼動する実行可能プログラムによってアクセスするためのコンテキスト情報の集約をプラットフォームレベルで行うことによって、コンテキストプラットフォーム204のコンテキスト情報の統合されたストレージを提供することができる。データストアは、コンテキストタイプを格納することができる。コンテキストタイプは、データブロックのビットによって表されるコンテキスト情報の識別子として働くGUIDとすることができる。コンテキスト情報は、本明細書で使用する場合、それだけには限定されないが、場所、アクティビティ、および状態の情報を含む論理的および物理的な実行環境についての任意の情報を含む。実際に、コンテキスト情報は、任意のサイズ、任意のフォーマット、または任意のタイプのデータとすることができる。例えば、コンテキスト情報は、バイナリデータ、フォーマット済みデータ、音声またはビデオなどのメディアデータ、人間可読データ、XML、暗号化されたデータなどを含む。データストア210に格納されている様々なコンテキスト情報の中には、ユーザコンテキストデータ212、実行可能コンテキストデータ214、およびシステムコンテキストデータ216がある。ユーザコンテキストデータ212は、Webをサーフィンする、プレゼンテーションプログラムを使用するなど、ユーザのアクティビティについての情報を含む。実行可能コンテキストデータ214は、プログラム設定、開いたファイルなど、実行可能プログラムの状態についての情報を含む。また、システムコンテキストデータ216は、バッテリー電力レベルのようなハードウェア装置の状態、ネットワーク接続など、システム状態についての情報を含む。

【0022】

実行可能プログラムは、アプリケーションプログラム、カーネル構成要素、またはコンテキスト情報を提供または取得し、および/またはコンテキスト情報の変更に従ってそのコンピューティング環境を適応させることができる他の実行可能プログラムとすることができる。コンテキスト情報を設定し、追跡し、取得するために呼び出し可能なインターフェースにアクセスするように、実行可能プログラム202をコンテキストAPI206に動作可能に結合することができる。例えば、Microsoft（登録商標）Windows（登録商標）XPオペレーティングシステムのNetwork Location Awareness（NLA）サービスは、SetContextData APIを呼び出して、ネットワークのコンテキスト情報を新しいネットワーク接続に設定することによって、ネットワークの変更を監視し、ネットワーク情報をコンテキストプラットフォーム

10

20

30

40

50

ムに提供することができる。コンテキストエンジンは、ネットワーク接続を関連付ける新しい値を受信して、ネットワーク接続のコンテキスト情報の変更について通知されるべき `TrackContext API` を呼び出した実行可能プログラムを検索することができる。次いでコンテキストエンジンは、ネットワーク接続のコンテキスト情報の変更の通知をこれらの実行可能プログラムに送信することができる。ネットワーク接続の変更の通知を受信した結果、1つまたは複数の実行可能プログラムは、デフォルトのプリンタ設定、デフォルトのインターネットブラウザホームページ、インターネットブラウザプロキシ設定など、システム設定を変更することができる。このように、実行可能プログラムは、コンテキスト情報の更新の通知があり次第、コンテキスト情報を使用してその実行環境を適応させることができる。

10

【0023】

図3は、代替実施形態におけるコンテキストウェアネスプラットフォームのシステム構成要素のアーキテクチャ例の概略を表すブロック図を示している。図2とともに説明したシステム構成要素に加えて、管理下のコンテキストアプリケーションプログラミングインターフェース304をコンテキストAPI206に動作可能に結合して、管理下のコードを使用してコンテキストプラットフォーム302を実施することができる。管理下のコードとは、Microsoft（登録商標）、.NET Platformのために開発されたソフトウェアコードを指し、Microsoft（登録商標）Common Language Runtime（CLR）によって使用することができるメタデータを含む。有利には、.NETランタイムは、メモリの割り振り、メモリの再利用、型チェック、および他のデータ関連のタスクなど、データ関連のタスクを管理することができる。管理下のコンテキストAPI304は、2つの静的メソッド `SetContext` および `GetContext` を備える `ContextClient` オブジェクト、およびウィンドウメッセージと直接対話するより、.NET相互運用アセンブリを使用して実行可能プログラムがコンテキスト情報イベントを受信できるようにする1組の非静的メソッドを含む。管理下のコンテキストAPI304は、コンテキストAPI206に動作可能に結合し、上述したコンテキストAPI206を使用して、.NET相互運用アセンブリを用いてコンテキストエンジン208と通信する。

20

【0024】

`SetContext` メソッドは、コンテキストのタイプについての新しいコンテキスト情報を提供するために実行可能プログラムによって使用することができる。コンテキストのタイプは、メソッドに渡されるオブジェクトのタイプによって決定することができる。コンテキストエンジンは、メソッドに渡されるオブジェクトに対応するGUIDにコンテキスト情報を関連付けることによって、コンテキスト情報を格納することができる。実行可能プログラムは、`GetContext` メソッドを使用して、特定のタイプのコンテキストの現在のコンテキスト情報を取得することができる。コンテキストエンジンは、コンテキストのタイプのGUIDを使用して、`SetContext` メソッドへの前回の呼び出しによってデータストアに書き込まれる、格納済みコンテキスト情報を検索する。

30

【0025】

管理下のコンテキストAPI304は、`ContextClient` と呼ばれるオブジェクトによってアクセス可能なイベント機構も含む。`ContextClient` は、`OldContext`、`NewContext`、`ContextType` および `Stale` などのプロパティとともに `ContextChangedEventArgs` を使用することができる `ContextChanged` という名前のイベント、および `ContextChangedEventHandler` という名前のイベントハンドラを提示することができる。`OldContext` プロパティおよび `NewContext` プロパティは、イベントの受信側が適切な行動を起こすことができるように、コンテキストの状態が変更される前または後に、イベントの受信側がそれを決定する。`ContextType` は、このイベントにおけるオブジェクトのタイプを表す。最後に、`Stale` フラグは、このコンテキスト情報が依然として揮発性メモリに存在しているという条件で、実行可能プ

40

50

ログラムであるかどうかを示すために使用する。

【0026】

図4は、コンテキストプラットフォームの様々な実装形態に含むコンテキストエンジンのシステム構成要素のアーキテクチャ例の概略を表すブロックエンジンを示している。コンテキストエンジン208は、メッセージキューからコンテキスト情報オペレーションを実行する旨の要求を受信し、コンテキスト情報の更新でデータストア210を更新し、実行可能プログラム202に変更を通知する。コンテキストエンジン208は、メッセージキュー402、コンテキストメッセージプロセッサ404、コンテキストデータ406、コンテキスト通知キュー414、コンテキストノーティファイア416、通知テーブル418、および1つまたは複数のノーティファイアスレッド420を含む。一実施形態において、コンテキスト情報オペレーションの実行可能プログラムによって送信された要求がメッセージキュー402で受信されるたびに、オペレーティングシステムは、コンテキストメッセージプロセッサ404をディスパッチして、メッセージを処理する。コンテキストエンジン208によって受信されるメッセージは、コンテキスト情報をコンテキストデータ406に追加するSetContextData要求、コンテキストデータ406内のコンテキスト情報への更新が通知されるTrackContextData要求、およびコンテキストデータ406からコンテキスト情報を取り出すGetContextData要求を含む。

【0027】

コンテキストメッセージプロセッサ404は、コンテキスト情報406によって示されるように、コンテキスト情報をメモリに格納する。コンテキストデータ406に格納されている様々なコンテキスト情報の中には、ユーザコンテキストデータ408、実行可能コンテキストデータ410、およびシステムコンテキストデータ412がある。コンテキストエンジン208は、コンテキスト情報を含む関連のデータブロックとともに、コンテキスト情報をGUIDとして格納する。GetContextData要求を受信するたびに、コンテキストメッセージプロセッサ404は、コンテキストデータ406から要求されたコンテキスト情報を取得し、コンテキスト情報のコピーを共有メモリバッファやメモリマップファイル、またはディスク上のファイルやシステムレジストリなどの他の共有場所に入れて、通知を受信すると実行可能プログラムがそれを読み取り可能とする。SetContextData要求が受信されるたびに、コンテキストメッセージプロセッサ404は、SetContextData要求内のコンテキスト情報をコンテキストデータ406に追加することによってコンテキストデータ406を更新し、コンテキストメッセージプロセッサ404は、コンテキスト通知キュー414にコンテキスト情報を入れる。一実施形態において、コンテキスト情報は、古いコンテキスト情報、新しいコンテキスト情報、およびこのコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストを含む更新オブジェクトとすることができる。別の実施形態において、実行可能プログラムのリストを1つまたは複数のコンテキストタイプまたはGUIDに関連付ける通知テーブル418に、コンテキスト情報内の更新が通知されるよう要求した実行可能プログラムのリストを別々に格納してもよい。コンテキスト情報をコンテキスト通知キュー414に入れた後、コンテキストノーティファイア416を起動させる。一実施形態において、コンテキストノーティファイア416は、コンテキストエンジン208によって生じたスレッドとしてもよい。コンテキストノーティファイア416は、コンテキスト通知キュー414からコンテキスト情報の更新を読み取り、このコンテキスト情報の更新が通知されるよう要求した各実行可能プログラムに通知メッセージを送信する。

【0028】

一実施形態において、コンテキストノーティファイア416は、通知メッセージをコンテキスト情報とともに非同期で送信する1つまたは複数のノーティファイアスレッド420を作成して、個々の実行可能プログラムに、コンテキスト情報の更新を通知することができる。したがって、ノーティファイアスレッド上の任意の実行可能プログラムによる同期ブロックは、他の通知が送信されるのを防止せず、任意の1つの実行可能プログラムに

以前送信された通知の完了を要求することなく、任意の数の通知を実行可能プログラムに送信することができる。コンテキスト情報の更新の通知を受信すると、実行可能プログラムは、共有メモリバッファやメモリマップファイルなどの共有場所からコンテキスト情報の更新にアクセスする。

【0029】

定期的に、コンテキストエンジン208は、コンテキストデータ406内のコンテキスト情報をデータストア210などの永続ストレージに書き込む。コンテキストエンジン208は、永続ストレージに通知テーブル418を書き込んでよい。一実施形態において、コンテキストエンジン208は、非アクティブ期間の後でスリープになる。スリープになるのに先立って、コンテキストエンジン208は、永続ストレージに、コンテキストデータ406および通知テーブル418を書き込む。デフォルトの非アクティブ期間は、例えば1時間とすることができる。様々な実施形態において、ユーザアクティビティの予想される期間に適した他の期間を使用してもよい。一部の実施形態において、期間は、ユーザによって設定することができる。コンテキストエンジンは、起動される場合、永続ストレージからコンテキストデータを読み込み、それをコンテキストデータ406に書き込んだり、永続ストレージから通知リストを読み込み、それを通知テーブル418に書き込む。

10

【0030】

実行可能プログラムは、他の実行可能プログラムによって使用するためのコンテキスト情報を提供することを望む場合はいつでも、1組の要求をコンテキストエンジンに送信する。図5は、コンテキスト情報を設定する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートを示している。ステップ502で、コンテキストの変更が最初に検出される。次いでステップ504で、コンテキスト情報を設定する旨の要求が送信される。例えば、Microsoft(登録商標)Windows(登録商標)XPオペレーティングシステムのNetwork Location Awareness(NLA)サービスなどの実行可能プログラムは、ネットワークの変更を監視し、新しいネットワーク接続を検出することができる。次いで実行可能プログラムは、SetContextData APIを呼び出して、ネットワークのコンテキスト情報を新しいネットワーク接続に設定することによって、ネットワーク情報をコンテキストプラットフォームに提供することができる。

20

30

【0031】

図6は、コンテキスト情報を追跡する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートを示している。まず、ステップ602で、コンテキスト情報を追跡する旨の要求が送信される。例えば、コンピュータシステムのデフォルトのプリンタ設定の責任を負う実行可能プログラムは、TrackContextData APIを呼び出して、ネットワーク接続についてのコンテキスト情報への更新が通知される旨の追跡要求をコンテキストプラットフォームに送信する。次に、ステップ604で、コンテキスト情報の更新の通知を受信する。次いでステップ606で、コンテキスト情報の更新を取得する。例えば、デフォルトのプリンタ設定の責任を負う前の例の実行可能プログラムは、新しいネットワーク接続についてのコンテキスト情報の更新の通知を受信し、新しいネットワーク接続についてのコンテキスト情報の更新のコピーを共有メモリバッファまたはメモリマップファイルから取り出す。最後に、ステップ608で、実行可能プログラムは、コンテキスト情報の更新に従って適応(adapt)する。したがって、デフォルトのプリンタ設定の責任を負う前の例の実行可能プログラムのような実行可能プログラムは、例えばデフォルトのプリンタ設定を新しいネットワーク接続に使用される設定に変更することによって、コンテキスト情報の更新に従ってその実行環境を適応させることができる。

40

【0032】

図7は、コンテキスト情報を取得する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートを示している。まず、ステップ7

50

02で、コンテキスト情報を取得する旨の要求が送信される。例えば、ブラウザのデフォルトのホームページを設定する責任を負う実行可能プログラムは、GetContextData APIを呼び出して、ネットワークのコンテキスト情報を取得する。次に、ステップ704で、コンテキスト情報の通知を受信する。最後に、ステップ706で、実行可能プログラムは、取得されたコンテキスト情報に従って適応することが。例えば、ブラウザのデフォルトのホームページを設定する責任を負う前の例の実行可能プログラムは、デフォルトのホームページ設定を、ネットワーク接続に使用される設定に変更する。

【0033】

図8は、要求されたコンテキスト情報オペレーションを実行する一実施形態においてコンテキストエンジンによって引き受けられるステップ例の概略を表すフローチャートを示している。実装は、同じ効果を達成しながら、また本発明の範囲から逸脱することなく、これらのステップを異なる順序で実行することを選択したり、効率や柔軟性の目的でこれらのステップの一部のみを実行したりすることを選択できることを当業者であれば理解されよう。ステップ802で、コンテキスト情報オペレーションを受信する。コンテキストエンジン208は、例えば、実行可能プログラムからコンテキスト情報オペレーションを受信する。ステップ804で、受信されたコンテキスト情報オペレーションはコンテキスト情報を設定する旨の要求であるかどうかを決定する。そうである場合、ステップ806で、コンテキスト情報をメモリ内に格納されているコンテキストデータの表現に追加し、ステップ808で、コンテキスト情報をコンテキスト通知キューに入れる。一実施形態において、コンテキストエンジン208は、設定要求内のコンテキスト情報をコンテキストデータ406に追加し、次いでコンテキストエンジン208は、コンテキスト情報をコンテキスト通知キュー414に入れる。

【0034】

ステップ810で、このコンテキスト情報の更新が通知されるよう要求した任意の実行可能プログラムが、このコンテキスト情報の更新について通知される実行可能プログラムのリスト内で検索される。例えば、一実施形態において、実行可能プログラムのリストをコンテキストタイプまたはGUIDに関連付ける通知テーブル418などの通知テーブルに、このコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストを別々に格納する。別の実施形態において、このコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストは、数ある情報の中でも、更新されたコンテキスト情報を含む更新オブジェクト内に格納することができる。

【0035】

ステップ811で、このコンテキスト情報の更新が通知されるべき実行可能プログラムのリストに含まれるよう要求した実行可能プログラムがあるかどうか決定される。ある場合、ステップ812で、このコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストに、通知を送信する。一実施形態において、コンテキストノーティファイア416は、コンテキスト通知キュー414からコンテキスト情報の更新を読み取り、このコンテキスト情報の更新が通知されるよう要求した各実行可能ファイルに通知メッセージを送信することができる。様々な実施形態において、コンテキストノーティファイア416は、通知メッセージをコンテキスト情報とともに非同期で送信する1つまたは複数のノーティファイアスレッド420を作成して、個々の実行可能ファイルにコンテキスト情報の更新を通知する。

【0036】

ステップ812で、このコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストに通知を送信した後、またはステップ811で、このコンテキスト情報の更新が通知されるべき実行可能プログラムのリスト内に含まれるよう要求した実行可能プログラムがないと決定された場合、ステップ814で、コンテキスト情報の更新を永続的に格納する。例えば、コンテキストエンジン208は、コンテキストデータ406内の更新されたコンテキスト情報をデータストア210などの永続ストレージに定期的書き込む。非アクティブ期間の後、コンテキストエンジン208は、一実装形態において、ス

10

20

30

40

50

リープになるのに先立って、コンテキストデータ 406 を永続ストレージに書き込んでもよい。更新されたコンテキスト情報が永続的に格納された後、処理が終了する。

【0037】

ステップ 804 に戻って、受信されたコンテキスト情報オペレーションが設定要求ではなかった場合、ステップ 816 で、受信されたコンテキスト情報オペレーションがコンテキスト情報を追跡する旨の要求であるかどうかを決定する。そうである場合、ステップ 818 で、コンテキスト情報の更新が通知される実行可能プログラムのリストを更新する。次に、ステップ 820 で、コンテキスト情報の更新が通知される実行可能プログラムの更新済みリストを永続的に格納する。例えば、コンテキストエンジン 208 は、通知テーブル 418 内の実行可能プログラムの更新済みリストを永続ストレージに定期的に書き込む。別の実施形態において、コンテキストエンジン 208 は、数ある情報の中でも更新済みのコンテキスト情報を含む更新オブジェクト内の実行可能プログラムの更新済みリストを永続ストレージに定期的に書き込む。非アクティブ期間の後、コンテキストエンジン 208 は、様々な実施形態において、スリープになるのに先立って、実行可能プログラムの更新済みリストを永続ストレージに書き込んでもよい。通知されるべき実行可能プログラムの更新済みリストが永続的に格納された後、処理が終了する。

【0038】

ステップ 816 に戻って、受信されたコンテキスト情報オペレーションが追跡要求ではなかった場合、ステップ 822 で、受信されたコンテキスト情報オペレーションがコンテキスト情報を取得する旨の要求であるかどうかを決定する。そうである場合、ステップ 824 で、コンテキスト情報が取り出される。例えば、一実施形態において、コンテキストエンジン 208 は、コンテキストデータ 406 からコンテキスト情報を取得する。別の実施形態において、コンテキストエンジン 208 は、データストア 210 からコンテキスト情報を取得する。次に、ステップ 826 で、コンテキスト情報を要求側に送信する。コンテキスト情報を、取得要求を送信した実行可能プログラムに送信した後、処理が終了する。一実施形態において、取得要求の送信に応答して任意の実行可能プログラムがコンテキスト情報を受信するたびに、このコンテキスト情報の更新が通知されるよう要求した実行可能プログラムのリストに通知を送信してもよい。

【0039】

有利には、このシステムおよび方法は、コンテキスト情報を使用してコンピュータを実行環境に知的に適応させることができるように、コンピュータシステムが実行環境についてのコンテキスト情報を収集し、提示する。実行可能プログラムは、コンテキスト情報の単一の項目を使用することによって実行環境を適応させ、またはコンテキスト情報の複数の項目を使用することによって実行環境を適応させる。さらに、実行可能プログラムは、使用可能なコンテキスト情報を使用して、コンテキスト情報を設定し、または実行環境を適応させるのに、他の実行可能プログラムから独立して動作し、またはいくつかの実行可能プログラムとともに動作する。プラットフォームで稼働している任意の実行可能プログラムがコンテキスト情報を提供または使用して、コンピュータシステムを知的に適応させてユーザの生産性を向上させるように、コンテキスト情報を、プラットフォームレベルで統合のデータストアに集約する。

【0040】

上記の詳細な説明からわかるように、本発明は、実行環境を適応させるために使用することができるコンテキストウェアネスプラットフォームの改良されたシステムおよび方法を提供する。本発明のアーキテクチャは、コンテキスト情報を設定し、追跡し、取得するインターフェースを備える構成要素およびフレームワークを提供する。提供されるシステムおよびフレームワークは拡張可能である。ユーザコンテキストデータ、実行可能コンテキストデータ、およびシステムコンテキストデータを含めて、任意のタイプのコンテキスト情報を使用することができる。コンテキスト情報を提供し、または使用する任意の数の実行可能プログラムをサポートすることができる。したがって、このシステムおよび方法は、現代のコンピューティングに必要なかなりの利点および利益を提供することがわか

る。

【 0 0 4 1 】

本発明は、様々な変更および代替の構成が可能であるが、示したいいくつかの実施形態を図面に示し、上記で詳しく説明している。しかし、本発明を開示した特定の形に限定する意図はなく、逆に、本発明は、本発明の意図および範囲内に含まれるすべての変更形態、代替の構造、および均等物をカバーすることを理解されたい。

【図面の簡単な説明】

【 0 0 4 2 】

【図 1】本発明を組み込むことができるコンピュータシステムの概略を表すブロック図である。

10

【図 2】本発明の一態様によるコンテキストウェアネスプラットフォームの一実施形態におけるシステム構成要素のアーキテクチャ例の概略を表すブロック図である。

【図 3】本発明の一態様によるコンテキストウェアネスプラットフォームの別の実施形態におけるシステム構成要素のアーキテクチャ例の概略を表すブロック図である。

【図 4】本発明の一態様によるコンテキストプラットフォームの様々な実施形態に含まれるコンテキストエンジンのシステム構成要素のアーキテクチャ例の概略を表すブロック図である。

【図 5】本発明の一態様によるコンテキスト情報を設定する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートである。

【図 6】本発明の一態様によるコンテキスト情報を追跡する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートである。

20

【図 7】本発明の一態様によるコンテキスト情報を取得する一実施形態において実行可能プログラムによって引き受けられるステップ例の概略を表すフローチャートである。

【図 8】本発明の一態様による要求されたコンテキスト情報オペレーションを実行する一実施形態においてコンテキストエンジンによって引き受けられるステップ例の概略を表すフローチャートである。

【符号の説明】

【 0 0 4 3 】

- 1 2 0 処理ユニット
- 1 2 1 システムバス
- 1 3 0 システムメモリ
- 1 3 4 オペレーティングシステム
- 1 3 5 アプリケーションプログラム
- 1 3 6 他のプログラムモジュール
- 1 3 7 プログラムデータ
- 1 4 0 固定式不揮発性メモリインターフェース
- 1 4 4 オペレーティングシステム
- 1 4 5 アプリケーションプログラム
- 1 4 6 他のプログラムモジュール
- 1 4 7 プログラムデータ
- 1 5 0 取外式不揮発性メモリインターフェース
- 1 6 0 ユーザ入力インターフェース
- 1 6 1 マウス
- 1 6 2 キーボード
- 1 6 3 マイク
- 1 6 4 タブレット
- 1 7 0 ネットワークインターフェース
- 1 7 1 ローカルエリアネットワーク
- 1 7 2 モデム
- 1 7 3 広域ネットワーク

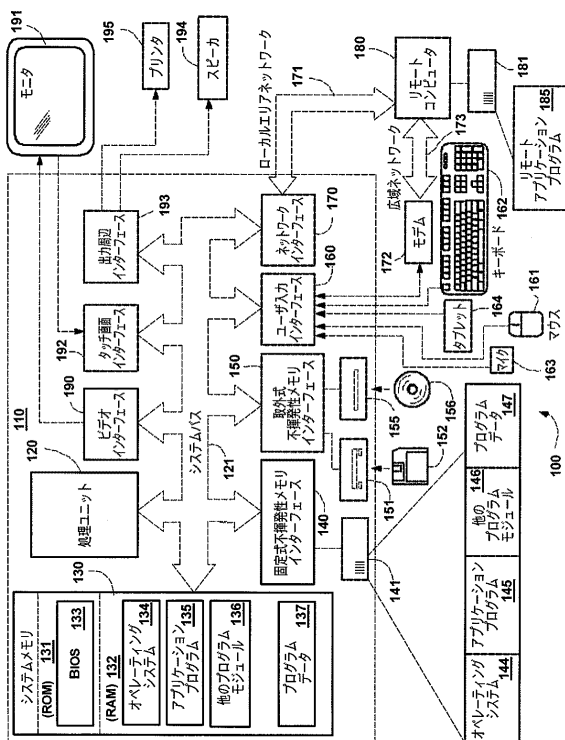
30

40

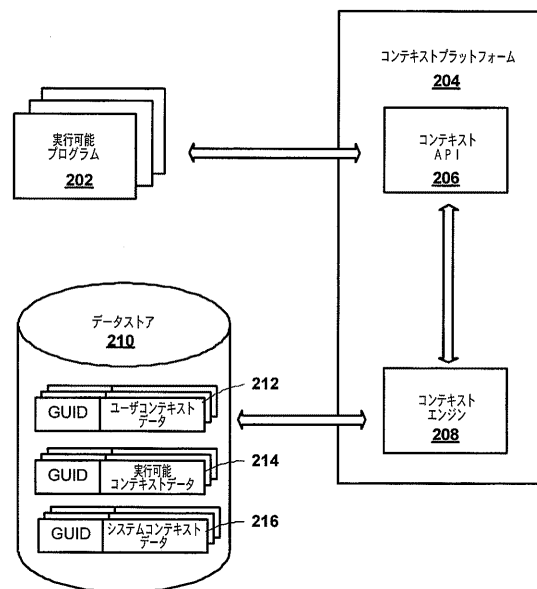
50

- 180 リモートコンピュータ
- 185 リモートアプリケーションプログラム
- 190 ビデオインターフェース
- 191 モニタ
- 192 タッチ画面インターフェース
- 193 出力周辺インターフェース
- 194 スピーカ
- 195 プリンタ

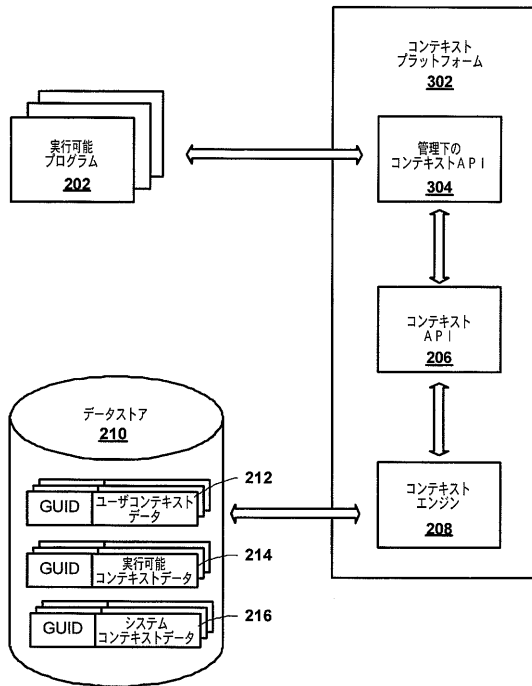
【図1】



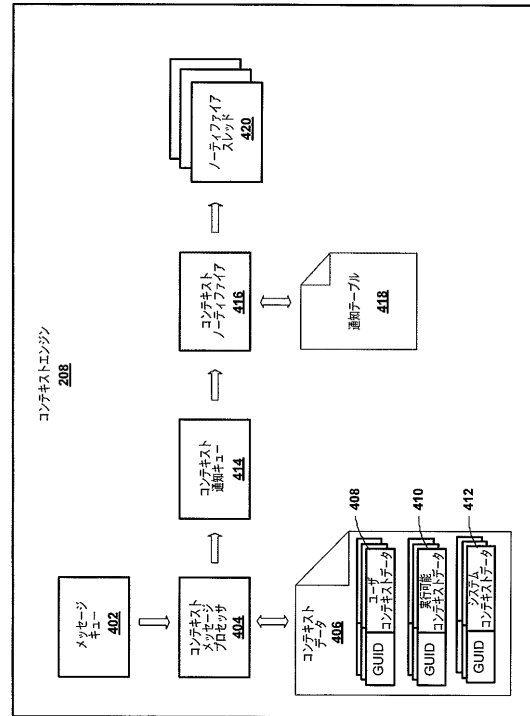
【図2】



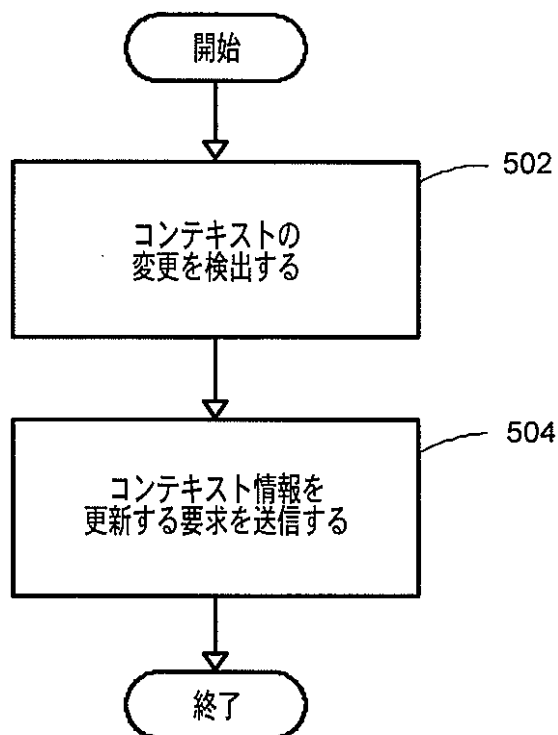
【 図 3 】



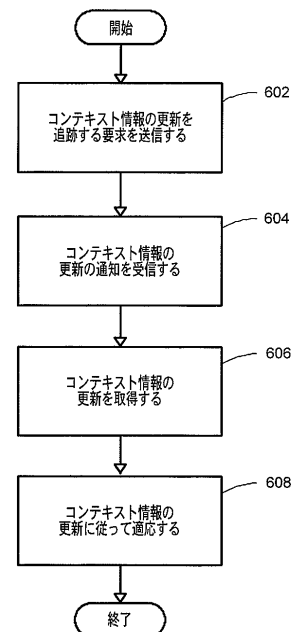
【 図 4 】



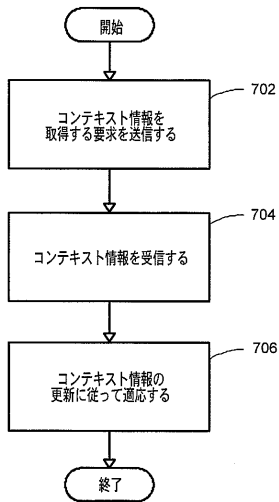
【 図 5 】



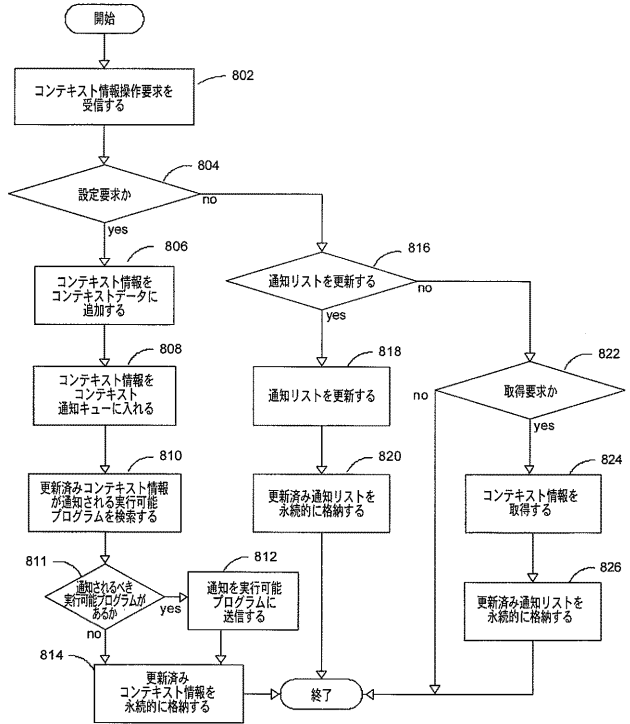
【 図 6 】



【図 7】



【図 8】



フロントページの続き

- (72)発明者 カメシュ チャンダー ツムシ ダヤカール
アメリカ合衆国 9 8 0 5 2 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 ラビパル ソイン
アメリカ合衆国 9 8 0 5 2 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 ライアン エドワード キュキエマン
アメリカ合衆国 9 8 0 5 2 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 ビクラム マダン
アメリカ合衆国 9 8 0 5 2 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内

F ターム(参考) 5B076 AA02

【外国語明細書】

2006209774000001.pdf