

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4917744号
(P4917744)

(45) 発行日 平成24年4月18日(2012.4.18)

(24) 登録日 平成24年2月3日(2012.2.3)

(51) Int.Cl.

F I

G 0 6 F 9/44 (2006.01)

G 0 6 F 9/06 6 2 0 K

請求項の数 6 外国語出願 (全 20 頁)

(21) 出願番号 特願2004-219244 (P2004-219244)
 (22) 出願日 平成16年7月27日(2004.7.27)
 (65) 公開番号 特開2005-135377 (P2005-135377A)
 (43) 公開日 平成17年5月26日(2005.5.26)
 審査請求日 平成19年7月27日(2007.7.27)
 (31) 優先権主張番号 10/674,834
 (32) 優先日 平成15年9月30日(2003.9.30)
 (33) 優先権主張国 米国(US)

(73) 特許権者 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 レネ ビス
 デンマーク ディーケー-3200 ヘル
 サイン ラムルーズ アレソガーズ アレ
 8

最終頁に続く

(54) 【発明の名称】 ランタイムおよび設計でのテキスト変換と多言語サポートをするラベルシステム

(57) 【特許請求の範囲】

【請求項 1】

ラベルテキストデータベースを備えたビジネスソリューションシステムにおいて、新しいラベルを作成する方法であって、前記ラベルテキストデータベースは、ラベル毎に、異なる言語バージョンの1つまたは複数のラベルテキストを含み、前記方法は、

ユーザによって入力された、前記新しいラベルに対する所望のテキストの少なくとも一部と、前記新しいラベルを使用すべきプログラムを示す名前空間の指示と、前記新しいラベルを前記プログラムで使用するとき、前記新しいラベルを適用すべき前記システムの特定のコントロール機能を示すカテゴリの指示とを受け取るステップと、

前記所望のテキストの少なくとも一部にマッチするラベルテキストを有するラベルを、
前記ラベルテキストデータベース中で検索するステップと、

前記ラベルテキストデータベース中で見つかった前記所望のテキストの少なくとも一部にマッチする前記ラベルテキストを有する前記ラベルを、マッチのリストとして前記ユーザに返すステップと、

前記マッチのリスト内の前記ラベルのうちの少なくとも1つが選択されると、前記選択されたラベルのカテゴリおよび名前空間を、前記ユーザによって指示された前記新しいラベルのカテゴリおよび名前空間とそれぞれ比較するステップと、

前記選択されたラベルのカテゴリまたは名前空間のいずれか一方が前記新しいラベルのカテゴリまたは名前空間と異なるとき、前記選択されたラベルを複製することによって前記新しいラベルを作成するステップであって、前記選択されたラベルを複製することは、

10

20

前記選択されたラベルのラベルテキストを複製して前記新しいラベルのラベルテキストとすることを含む、ステップと

を備えることを特徴とする方法。

【請求項 2】

前記新しいラベルのための新しいエントリを前記ラベルテキストデータベースに作成するステップと、

前記新しいラベルにグローバル一意識別子 (G U I D) を割り当てるステップと、

前記新しいラベルの前記ラベルテキストのための記録を前記ラベルテキストデータベース中に作成するステップと、

前記新しいラベルを前記ラベルテキストデータベースに記憶するステップと

をさらに含むことを特徴とする請求項 1 に記載の方法。

10

【請求項 3】

前記選択されたラベルが、異なる言語バージョンで表される複数のラベルテキストを有するとき、

前記選択されたラベルのラベルテキストを複製して前記新しいラベルのラベルテキストとすることは、各言語バージョンのラベルテキストをそれぞれ複製して前記新しいラベルのラベルテキストとすることを含み、

前記新しいラベルを前記ラベルテキストデータベースに記憶するステップは、各言語バージョンの前記ラベルテキストを、前記新しいラベルのラベルテキストとして前記ラベルテキストデータベース中の記録に記憶するステップをさらに含むことを特徴とする請求項 2 に記載の方法。

20

【請求項 4】

前記マッチのリストをユーザインタフェース上に表示するステップをさらに含むことを特徴とする請求項 1 に記載の方法。

【請求項 5】

前記選択されたラベルのカテゴリおよび名前空間が前記新しいラベルのカテゴリおよび名前空間とそれぞれ同じである場合は、前記選択されたラベルを前記新しいラベルとして使用するステップをさらに含むことを特徴とする請求項 1 に記載の方法。

【請求項 6】

前記新しいラベルを作成するステップは、

前記新しいラベルの複製元である前記選択されたラベルのグローバル一意識別子 (G U I D) を示すエントリを、前記新しいラベル中に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 1 に記載の方法。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ビジネスソリューションソフトウェアプログラムにおいてラベルをマージおよび開発することに関する。より詳細には、本発明は、ビジネスソリューションソフトウェアのためのモジュール中のラベルを開発およびマージする際に使用されるラベルデータベースおよびラベルダイアログに関する。

40

【背景技術】

【0002】

ビジネスソリューションソフトウェアプログラムは、典型的には企業であるエンドユーザに、ユーザの様々なビジネスパートナーとの接続性をサポートするカスタマイズ可能かつスケーラブルかつグローバルなエンタープライズリソースプランニングソリューションを提供するものである。多くのビジネスソリューションソフトウェアプログラムは、それを実行する企業のニーズをさらに満たすために、ソフトウェアの基本機能を元々の製品からさらに拡張する能力を提供する。この新しいまたは追加の機能は、ビジネスソリューションソフトウェアに含まれる既存の機能および既存のデータを利用するように書かれた追加のモジュールを介して提供される。これらの追加モジュールはしばしば、ビジネスソリ

50

ューションソフトウェアおよび既存のデータを、ソフトウェアの古い機能と新しい機能の両方と自動的に同期させる。

【 0 0 0 3 】

いくつかのビジネスソリューションソフトウェアは、様々な国で複数の言語にまたがって複数の通貨でビジネスを実施する能力を提供する。ビジネスソリューションソフトウェアで提供される多言語能力を用いることにより、送り状などの文書を、受取り側に好まれる言語で送ることが可能である。しかし、現在のシステムで文書の言語を変更するには、新しい言語をビジネスソリューションソフトウェアにロードして、システムの動作言語全体を変更する必要がある。

【 0 0 0 4 】

ビジネスソリューションソフトウェアの様々な利用可能言語による通信は、複数のラベルを使用することによって対処される。ラベルは、ユーザインタフェースコンポーネント上に、または印刷された文書に現れるテキストである。ラベルは、メニュー、ボタン、ダイアログボックスなどの上で使うことができる。さらに、ラベルは、「ラベル」「ヘルプ」「キャプション」「ツールチップ」などのラベルプロパティを有するコントロール上で使用することもできる。現在のビジネスソリューションソフトウェアにおけるラベルは、別々のリソースファイルに記憶されており、ビジネスソリューションソフトウェアによって使用される各言語につき、1つのリソースファイルが専用になっている。さらに、ビジネスソリューションソフトウェアにおける各モジュールは、他のモジュールともビジネスソリューションソフトウェア自体とも共有されないそれ自体のリソースファイルを有する。

【 0 0 0 5 】

ビジネスソリューションソフトウェアの開発全体を通して、開発者の間には、既存のラベルを再利用したいという強い欲求があった。しかし、既存のラベルの再利用は、好都合ではないと言われてきた。というのは、ラベルの語句に関連する様々なプロパティは異なるユーザ間で変化する可能性があるからであり、あるいは、ラベルの意味は異なる開発者間でばらつきがある可能性があるからである。このことは、語句が変更されたときに問題を生じる恐れがある。例えば、あるラベルテキストは、あるアプリケーション上ではメニュー中で使用することができ、別のアプリケーション上ではボタン中で使用することができ、そのため各ラベルごとにプロパティが違ってくる。ほとんどの開発者が望んできたのは、ラベルを構成する語句またはテキストを再利用して、システム上の様々なモジュール用にラベルを開発することに関連するコストを削減することであった。

【 0 0 0 6 】

前述のように、ビジネスソリューションソフトウェアモジュールでは通常、ラベルはリソースファイルに保持される。しかし、現在のビジネスソリューションソフトウェアは、SQL (Structured Query Language) テーブルなどのデータベースメタデータストアを介して利用可能な、またはウェブサービスを介して利用可能な包括的なリソースファイルを使用しない。通常、ビジネスソリューションソフトウェアでは、これらのラベルはモジュール特有であり、独自仕様のリソースファイルに記憶され、モジュール中に存在する各言語につき、1つのリソースファイルが専用になっている。独自仕様のリソースファイルを使用することに関する問題の1つは、開発者が、あるモジュール中のラベルの一部を、別のリソースファイルに含まれる新しい情報またはプロパティで置換または編集したいと思っても、開発システムは、ビジネスソリューションソフトウェアの他のモジュール中で、所望のラベルと同じラベルプロパティおよび/または用語法を有する別のラベルを探すことをしないことである。さらに、リソースファイルを使用すると、ソフトウェアソリューション中に多数のラベルが存在するため、ラベルの管理が極度に難しくなる。しかし、ビジネスソリューション環境では、ビジネスソリューションソフトウェアは、複数のベンダによって開発された多くの様々なソリューションを扱うことが必要とされる。しばしば、これらのモジュールの開発者は、ビジネスソリューションソフトウェア中の他のモジュール用に開発されたラベルと重複するラベルを開発していることがある。ラベルの開発お

10

20

30

40

50

よびラベルの変換（多言語サポートが望まれるとき）に関連するコストは高価で時間がかかり、特に、ラベルおよびその変換がすでにビジネスソリューションソフトウェア中のどこかに存在するときはそうである。したがって、各ベンダが所望の各モジュールごとにそれ自体のラベルを開発する必要のある完全に独自仕様のシステムとは反対に、様々なモジュールベンダの寄与の結合を用いるラベルシステムを有することが望ましい。さらに、既存のラベルテキストを検索することを可能にし、テキストを再利用する能力を提供するシステムであって、ラベルが使用されるコンテキストがその元々の作成時の使用法と異なるかもしれない場合にも対処するシステムが望ましい。

【発明の開示】

【発明が解決しようとする課題】

10

【0007】

従来のシステムには上述したような種々の問題があり、さらなる改善が望まれている。

【0008】

本発明は、このような状況に鑑みてなされたもので、その目的とするところは、効率的な処理を可能とする、ランタイムおよび設計でのテキスト変換と多言語サポートをするラベルシステムを提供することにある。

【課題を解決するための手段】

【0009】

本発明は、ビジネスソリューションシステムにおけるモジュールの開発者のニーズに対処するものである。以前のシステムとは異なり、ラベル（すなわち表示テキスト）は共通のリソースファイル中に保持され、システムの各モジュールは、各モジュール用の独自仕様のリソースファイルではなく、同じリソースファイルにアクセスする。本発明の効率をさらに高めることに、すべての言語テキストは、元の言語と同じリソースファイルまたはデータベースに記憶される。モジュールによって使用される各ラベルは、ラベルテーブル中で、GUID（globally unique identifier）などの一意識別子、その使用法の説明、関連するカテゴリおよび名前空間で識別される。この情報は、開発者が、新しいラベルまたは新しいモジュールを作成するときにデータベース中の既存のラベルが有用かどうかを判定するのに役立つ。

20

【0010】

各ラベルはまた、ラベルテキストテーブルデータベース中にエントリのセットを有し、エントリのセットは、ラベルテキストの複数言語のバージョン、ならびにラベルテキストの管理に有用なその他の情報を表す。ラベルテキストの元の言語は、マスタ言語として識別される。ユーザがビジネスソフトウェアソリューションシステムの言語を変更したいとき、モジュールの言語を変更したいとき、または文書を別の言語で印刷したいときは、新しい言語セットをシステムにロードしてシステムの好まれる言語を一時的な言語で置換する必要なしに、ラベルテキストテーブルが利用可能言語へのアクセスを可能にする。これにより、システム全体が新しい言語で機能する必要があるせいでユーザが混乱するということなく、言語を素早く変更することができる。さらに、このシステムは常に、現在のアクティブ言語を表示しようとする。ラベルに対する現在のアクティブ言語が見つからない場合は、フォールバック言語を使用することができる。

30

40

【0011】

本発明の一実施形態はまた、開発者がビジネスソリューションソフトウェアのために新しいモジュールを開発している状況、または既存のモジュールを編集している状況にも対処する。本発明では、開発者は、ラベルダイアログ開発ツールを介して既存のラベルを識別することができる。まず開発者は、開発表示内でユーザインタフェースを開く。この表示は、ラベルテキストデータベースの検索とすべての利用可能ラベルへのアクセスとを可能にする検索機能を、開発者に対して表示する。次いで開発者は、所望のラベルのテキストを検索エンジンに入力する。このテキストは、通常の表現を使用して、所望のラベルの全体または一部として入力することができる。代替実施形態では、開発者は、現在のテキストラベルに関する所望の使用法、テキストラベルの言語、または、所望のラベルの使用

50

法を識別するのに役立つその他の情報を指示することができる。

【 0 0 1 2 】

入力されたデータに基づいて、検索エンジンはラベルテキストデータベースを検索し、所望のテキストに最もよくマッチする、かつ名前空間およびカテゴリが提供されていればそれらに最もよくマッチするラベルテキストを識別する。識別されたテキストは、検索エンジン表示内で開発者に返されるが、これらはデータグリッド中に表示することができる。結果は、所定の方法に従ってランク付けされる。例えば、所望のテキストに最もよくマッチするカテゴリおよび言語を有するテキストが先に表示され、同じカテゴリを有さないテキストが後に表示されるなどの方法に従う。次いで開発者は、所望のテキストを含む返された結果の中で、結果を見つけることができる。結果のうちの1つが新しいラベルに対する所望のテキストを含む場合、開発者は結果からこの所望のラベルを選択することができる。

10

【 0 0 1 3 】

結果を選択すると、この特定のラベルに関する、データベースに含まれるより多くの情報が開発者に提示される。選択したラベルテキストの使用法が新しいラベルと同じである場合は、開発者は、選択したラベルを所望のターゲットラベルに使用することができる。しかし、選択したラベルの使用法が新しいラベルと同じでない場合は、開発者は、このラベルを新しいラベルに複製することができる。

【 0 0 1 4 】

選択されたラベルが新しいラベルに複製されるときは、新しいラベルのためのGUIDが生成され、新しいラベルの記録の中に、このラベルに複製されたラベルのGUIDを示すエントリが生成される。このエントリは、親ラベルのテキストが変更されたときに新しいラベルのテキストも更新されるようにするために提供される。さらに、ラベルが新しいラベルに複製されるとき、関連する変換があれば、それらも新しいラベルについてのラベルテキストテーブルにコピーされる。これにより、新しいラベルを利用可能言語に変換することに関連するどんな追加コストも被らずに、ビジネスソリューションソフトウェアシステムの完全な言語機能が、新しいラベルに引き継がれる。別の実施形態では、ラベルの変換済みバージョンが更新されると、同じマスタラベルを共有するすべての関連ラベルが、変換済みラベルの新しいバージョンで更新される。

20

【 0 0 1 5 】

選択されたラベルが新しいラベルに複製されるときは、ラベルのテキスト情報、ならびにマスタラベルのラベルID (identification) を除く他のすべての情報が、新しいラベルにコピーされる。ラベルに対する所望のテキストが見つからない場合は、開発者は、所望のラベルのための新しいラベルを作成しなければならない。さらに、開発者は、新しいラベルのラベルテキストについての新しい変換も生成しなければならない。言い換えれば、開発者がシステム中でラベルを見つけた場合、このラベルが複製される。しかし、開発者がラベルに関する情報を変更した場合は、ラベルはシステムにとって新しいラベルとして見えることになる。

30

【 発明を実施するための最良の形態 】

【 0 0 1 6 】

以下、図面を参照して本発明を適用できる実施形態を詳細に説明する。図1に、本発明を実施することのできる適したコンピューティングシステム環境の例(図中100)を示す。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ついてどんな制限も意味しない。またコンピューティングシステム環境100は、この例示的な動作環境であるコンピューティングシステム環境100に示すコンポーネントのいずれか1つまたは組合せに関してどんな依存や要件を有するものとも解釈すべきではない。

40

【 0 0 1 7 】

本発明は、その他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も動作する。本発明で使用するのに適するであろう周知のコンピューティングシステム、

50

環境、および／または構成の例には、限定しないがパーソナルコンピュータ、サーバコンピュータ、ハンドヘルドデバイスまたはラップトップデバイス、マルチプロセッサシステム、マイクロプロセッサベースのシステム、セットトップボックス、プログラム可能な民生用電子機器、ネットワークＰＣ（personal computer）、ミニコンピュータ、メインフレームコンピュータや、これらのシステムまたはデバイスのいずれかを含む分散コンピューティング環境などが含まれる。

【 0 0 1 8 】

本発明は、プログラムモジュールなど、コンピュータによって実行されるコンピュータ実行可能命令の一般的なコンテキストで述べるができる。一般にプログラムモジュールは、特定のタスクを実施するか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、コンポーネント、データ構造などを含む。本発明は分散コンピューティング環境で実施することもでき、その場合、タスクは通信ネットワークを介してリンクされたリモート処理デバイスによって実施される。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は、メモリ記憶デバイスを含めたローカルとリモートの両方のコンピュータ記憶媒体に位置することができる。

【 0 0 1 9 】

図１を参照すると、本発明を実施するための例示的なシステムは、コンピュータ１１０の形の汎用コンピューティングデバイスを含む。コンピュータ１１０のコンポーネントには、限定しないがプロセッサ１２０と、システムメモリ１３０と、システムメモリを含めた様々なシステムコンポーネントをプロセッサ１２０に結合するシステムバス１２１とを含めることができる。システムバス１２１は、様々なバスアーキテクチャのいずれかを用いた、メモリバスまたはメモリコントローラ、周辺バス、ローカルバスを含めて、いくつかのタイプのバス構造のいずれかとすることができる。限定ではなく例として、このようなアーキテクチャには、ＩＳＡ（Industry Standard Architecture）バス、ＭＣＡ（Micro Channel Architecture）バス、ＥＩＳＡ（Enhanced ISA）バス、ＶＥＳＡ（Video Electronics Standards Association）ローカルバス、およびＰＣＩ（Peripheral Component Interconnect）バス（メザニンバスとも呼ばれる）が含まれる。

【 0 0 2 0 】

コンピュータ１１０は通常、様々なコンピュータ可読媒体を備える。コンピュータ可読媒体は、コンピュータ１１０からアクセスできる任意の利用可能な媒体とすることができる。揮発性と不揮発性の媒体、リムーバブルとノンリムーバブルの媒体の両方が含まれる。限定ではなく例として、コンピュータ可読媒体には、コンピュータ記憶媒体および通信媒体を含めることができる。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、その他のデータなどの情報を記憶するための任意の方法または技術で実現された、揮発性と不揮発性、リムーバブルとノンリムーバブルの両方の媒体が含まれる。コンピュータ記憶媒体には、限定しないがＲＡＭ（random access memory）、ＲＯＭ（read only member）、ＥＥＰＲＯＭ（electrically erasable programmable read-only memory）、フラッシュメモリまたは他のメモリ技術、ＣＤ（compact disc）-ＲＯＭ、ＤＶＤ（digital versatile disc）または他の光ディスク記憶装置、磁気力セット、磁気テープ、磁気ディスク記憶装置または他の磁気記憶デバイスが含まれ、あるいは、所望の情報を記憶するのに使用できコンピュータ１１０からアクセスできるその他の任意の媒体が含まれる。通信媒体は通常、搬送波や他のトランスポート機構などの変調されたデータ信号中に、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、またはその他のデータを組み入れたものであり、任意の情報送達媒体が含まれる。「変調されたデータ信号」という語は、情報が信号中に符号化される形で１つまたは複数の特性が設定または変更された信号を意味する。限定ではなく例として、通信媒体には、有線ネットワークや直接配線式接続などの有線媒体と、音響、ＲＦ（radio frequencies）、赤外線、その他の無線媒体などの無線媒体とが含まれる。以上の任意の組合せもコンピュータ可読

10

20

30

40

50

媒体の範囲に含めるべきである。

【 0 0 2 1 】

システムメモリ 1 3 0 は、読取り専用メモリ (R O M) 1 3 1 やランダムアクセスメモリ (R A M) 1 3 2 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。 R O M 1 3 1 には通常、起動中などにコンピュータ 1 1 0 内の要素間で情報を転送するのを助ける基本ルーチンを含む B I O S (b a s i c i n p u t / o u t p u t s y s t e m) 1 3 3 が記憶されている。 R A M 1 3 2 は通常、プロセッサ 1 2 0 がすぐにアクセス可能な、かつ / またはプロセッサ 1 2 0 が現在作用している、データおよび / またはプログラムモジュールを含む。限定ではなく例として、図 1 には、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、その他のプログラムモジュール 1 3 6、プログラムデータ 1 3 7 を示す。

10

【 0 0 2 2 】

コンピュータ 1 1 0 は、その他のリムーバブル / ノンリムーバブル、揮発性 / 不揮発性コンピュータ記憶媒体を備えることもできる。例にすぎないが図 1 には、ノンリムーバブル不揮発性の磁気媒体に対して読み書きするハードディスクドライブ 1 4 1 と、リムーバブル不揮発性の磁気ディスク 1 5 2 に対して読み書きする磁気ディスクドライブ 1 5 1 と、 C D R O M や他の光媒体などリムーバブル不揮発性の光ディスク 1 5 6 に対して読み書きする光ディスクドライブ 1 5 5 を示す。この例示的な動作環境で利用できる他のリムーバブル / ノンリムーバブル、揮発性 / 不揮発性コンピュータ記憶媒体には、限定しないが磁気テープカセット、フラッシュメモリカード、 D V D、デジタルビデオテープ、固体 R A M、固体 R O M などが含まれる。ハードディスクドライブ 1 4 1 は通常、インタフェース 1 4 0 などのノンリムーバブルメモリインタ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 は通常、インタフェース 1 5 0 などのリムーバブルメモリインタフェースでシステムバス 1 2 1 に接続される。

20

【 0 0 2 3 】

以上に論じ図 1 に示したドライブおよびそれらに関連する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、その他のデータの記憶域をコンピュータ 1 1 0 に提供する。例えば図 1 には、ハードディスクドライブ 1 4 1 が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、その他のプログラムモジュール 1 4 6、プログラムデータ 1 4 7 を記憶しているのが示されている。これらのコンポーネントは、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、その他のプログラムモジュール 1 3 6、プログラムデータ 1 3 7 と同じものとするのもでき、異なるものとするのもできることに留意されたい。ここでは、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、その他のプログラムモジュール 1 4 6、プログラムデータ 1 4 7 が少なくとも異なるコピーであることを示すために、異なる番号を付けてある。

30

【 0 0 2 4 】

ユーザは、キーボード 1 6 2、マイクロホン 1 6 3、マウスやトラックボールやタッチパッド等のポインティングデバイス 1 6 1 などの入力デバイスを介して、コンピュータ 1 1 0 にコマンドおよび情報を入力することができる。その他の入力デバイス (図示せず) には、ジョイスティック、ゲームパッド、衛星受信アンテナ、スキャナなどを含めることができる。これらおよび他の入力デバイスは、システムバスに結合されたユーザ入力インタフェース 1 6 0 を介してプロセッサ 1 2 0 に接続されることが多いが、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、その他のインタフェースおよびバス構造で接続されてもよい。モニタ 1 9 1 または他のタイプの表示デバイスも、ビデオインタフェース 1 9 0 などのインタフェースを介してシステムバス 1 2 1 に接続される。モニタに加えて、コンピュータは通常、スピーカ 1 9 7 やプリンタ 1 9 6 など他の周辺出力デバイスも備えることができ、これらは出力周辺インタフェース 1 9 5 を介して接続することができる。

40

50

【 0 0 2 5 】

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用いて、ネットワーク化された環境で動作することができる。リモートコンピュータ 1 8 0 は、パーソナルコンピュータ、ハンドヘルドデバイス、サーバ、ルータ、ネットワーク P C、ピアデバイス、またはその他の一般的なネットワークノードとすることができ、通常、パーソナルコンピュータ 1 1 0 に関して上述した要素の多くまたはすべてを備える。図 1 に示す論理接続は、ローカルエリアネットワーク (L A N) 1 7 1 およびワイドエリアネットワーク (W A N) 1 7 3 を含むが、その他のネットワークを含むこともできる。このようなネットワーキング環境は、オフィス、企業全体のコンピュータネットワーク、イントラネット、インターネットでよくみられる。

10

【 0 0 2 6 】

L A N ネットワーキング環境で使用されるときは、コンピュータ 1 1 0 は、ネットワークインタフェースまたはアダプタ 1 7 0 を介して L A N 1 7 1 に接続される。W A N ネットワーキング環境で使用されるときは、コンピュータ 1 1 0 は通常、インターネットなどの W A N 1 7 3 を介した通信を確立するためのモデム 1 7 2 または他の手段を備える。モデム 1 7 2 は内蔵でも外付けでもよく、ユーザ入力インタフェース 1 6 0 または他の適切な機構を介してシステムバス 1 2 1 に接続することができる。ネットワーク化された環境では、コンピュータ 1 1 0 に関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のメモリ記憶デバイスに記憶することができる。限定ではなく例として、図 1 には、リモートアプリケーションプログラム 1 8 5 がリモートコンピュータ 1 8 0 上にあるものとして示す。図示のネットワーク接続は例示的なものであり、コンピュータ間で通信リンクを確立するための他の手段を使用してもよいことは理解されるであろう。

20

【 0 0 2 7 】

一実施形態では、システム 1 0 0 は、一般的な経理およびビジネスの機能を特定のアプリケーションモジュールによって統合することのできるビジネス管理ソリューション生成ソフトウェアを備える。これらのモジュールには、財務、取引、ロジスティクス、製造、顧客サービス、プロジェクト、人的資源のためのモジュールを含めることができる。ただし、その他のモジュールタイプをこのビジネス管理ソリューションで使用することもできる。本発明のソリューションは、複数の通貨 (ユーロ、ドル、円、ウォンなど)、複数の言語 (英語、ドイツ語、フランス語、デンマーク語、ロシア語、日本語、中国語など)、複数の税金形式 (複数の税務当局とやり取りするエンドユーザによって使用される) をサポートするように構成することができる。

30

【 0 0 2 8 】

ビジネス管理ソリューションで使用されるモジュールは、異なる様々なソースから開発することができる。一実施形態では、これらのすべてのモジュール間にある共通のつながりは、ラベルの使用である。ラベルは、識別子で表されるテキストである。ラベルは、ダイアログボックス、テキストストリング、または、ユーザに情報を伝えるのに使用される任意のテキストを含めて、ラベルプロパティを有する任意のオブジェクト上で使用することができる。ラベルは一般に、グラフィカルユーザインタフェースすなわち G U I などのリソースを介してユーザに提示される。ただしラベルは、テキストをユーザに提示する他の手段を介してユーザに提示することもできる。さらに、ラベルは、「ラベル」「ヘルプ」「キャプション」「ツールチップ」などのラベルプロパティを有するコントロール上で使用することもできる。以前のビジネスソリューションソフトウェアでは、各モジュール用のラベルが別々のリソースファイルに保持される。しかし、これらのリソースファイルは一般に、特定モジュールの独自仕様のフラットファイルアーキテクチャで保持されていた。開発者はしばしば、モジュールの開発プロセス中に、現在作成中のラベルの要件を満たすラベルが別のモジュール中にあることを知っていることがある。しかし、従来技術のシステムにおける独自仕様のファイルソース構造では、開発者は、あるモジュールからのラベルを別のモジュール中で使用することができず、ラベルを検索することも不可能であった。

40

50

【 0 0 2 9 】

図 2 は、本発明の一実施形態による、ビジネスソリューションシステム 2 0 0 中のラベルを構成する様々な集まり (collection) の間の関係を示すエンティティ関係図である。図 2 のエンティティ関係図は、言語テーブル 2 1 0 (またはシステム中の利用可能言語を示すその他の指標) と、マスタ 2 2 0 と、ラベル I D テーブル 2 3 0 と、ラベルテキストテーブル 2 4 0 を含む。一実施形態では、これらの集まりは、テーブルに構成された S Q L (Structured Query Language) メタデータストアとして構成される。ただし、これらの集まりのその他の構成およびその他のデータベースを使用してもよい。ラベル I D テーブル 2 3 0 およびラベルテキストテーブル 2 4 0 については、図 3 A および 3 B に関してより詳細に述べる。

10

【 0 0 3 0 】

言語テーブル 2 1 0 は、少なくとも 2 つのサブフィールドを含むテーブルである。言語テーブル 2 1 0 中の 2 つのサブフィールドは、言語 I D フィールド 2 1 1 と言語名フィールド 2 1 2 である。言語 I D フィールド 2 1 1 は、特定言語を示すコードであってビジネスソリューションソフトウェアプログラムによって理解可能なコードを保持する。言語名フィールド 2 1 2 は、言語の名前を保持するテキストフィールドである。例えば、利用可能言語の 1 つが米国英語である場合、言語 I D フィールド 2 1 1 は例示として「 0 1 」とすることができ、 A S C I I (American National Standard Code for Information Interchange) 規格が使用される場合は「 e n - u s 」とすることができる。ただし、その他の I D タイプを言語 I D フィールド 2 1 1 中で使用してもよい。メタデータテーブルのこのエントリについての言語名フィールド 2 1 2 は、例えば「 E n g l i s h - U S 」となる。あるいは、このソリューションが特定の言語によって提供されたものである場合、このエントリは、その特定言語のテキストを有するラベルとしてもよい。

20

【 0 0 3 1 】

言語テーブル 2 1 0 に記憶された情報は、ユーザがシステムの動作言語を確認または変更したいときに、ラベルダイアログ表示内でユーザに対して表示することができる。言語テーブル 2 1 0 は、ラベルテキストテーブル 2 4 0 と 1 : n の関係にある。この関係 (1 : n) が生じるのは、システム 2 0 0 に利用可能な各言語で様々なラベルを表す複数のラベルテキストがある可能性があるからである。

【 0 0 3 2 】

マスタ 2 2 0 は、一実施形態では、システム 2 0 0 中の各ラベルに関する 1 つのバージョンを含むテーブルである。一実施形態では、マスタ 2 2 0 は、元の言語およびフォーマットによる、ラベルの元のバージョンを保持する。ただし、ラベルの他のバージョンをマスタ 2 2 0 に記憶してもよい。例えば、ベストプラクティスガイドラインが使用される場合、システム 2 0 0 は、米国英語のラベルをマスタ 2 2 0 に記憶することができる。ベストプラクティスガイドラインは、特定のカテゴリタイプ、説明、言語を使用してラベル開発を標準化する手順のセットである。システム 2 0 0 中の各ラベルには、マスタ 2 2 0 中の 1 つのエントリが関連する。マスタラベルは、ラベルの変換が更新されている最中や、ラベルが現在選択されている (アクティブな) 言語にまだ変換されていないときや、何らかの理由でシステム 2 0 0 からアクセス不可能になったときに使用される。しかし、代替実施形態では、マスタ 2 2 0 に記憶される情報は、ラベルテーブル 2 3 0 と言語 2 1 0 のどちらかの中に、マスタラベルのラベル I D を示す 1 つまたは複数のフィールドとして記憶することもできる。さらに、マスタ 2 2 0 は、どの言語ラベルが作成されたかに関する情報を保持する単純なフィールドとしてもよい。このフィールドは、システム 2 0 0 中の任意のテーブルに位置するものとしてすることができる。

30

40

【 0 0 3 3 】

各マスタテーブルエントリ 2 2 1 は、ラベル I D テーブル 2 3 0 中の 1 つのエントリに対応する。ラベル I D テーブル 2 3 0 は、変換者 (テキストを変換している人) がラベルを正しく変換するのに役立つ様々なプロパティを含む。これらのプロパティはまた、開発者がラベルを正しく使用するのにも役立つ。複数のラベル I D が同じマスタラベルを共有

50

することができるので、マスタテーブルエン트리 221 とラベル ID テーブル 230 との関係は 1 : n である。

【0034】

ラベルテキストテーブル 240 は、ラベル ID で識別される各ラベルについてのテキストを含むエントリを備える。ラベルテキストテーブル 240 はまた、様々な言語による各ラベル ID についての変換を含むエントリも備える。ラベル ID テーブル 230 とラベルテキストテーブル 240 との関係、ならびに言語テーブル 210 とラベルテキストテーブル 240 との関係は、1 : n である。これは、1 つのラベルに対してラベル ID テーブル 230 中に 1 つのエントリがあるが、ラベルのテキストは複数の言語で存在する可能性があるからである。しかし、本発明の他の実施形態では、ラベル ID とラベルテキストとの関係は 1 : 1 (またはテキストが存在しない場合は 1 : 0) であり、マスタラベルの各変換が、それ自体の固有のラベル ID およびエントリをラベル ID テーブル 230 中に有する。

10

【0035】

図 3A に、本発明の一実施形態による、ラベル ID テーブル 230 を構成するフィールドを示す。ラベル ID テーブル 230 は、ID フィールド 231 と、名前空間フィールド 232 と、カテゴリフィールド 234 と、説明フィールド 235 を含む。他の実施形態では、ラベル ID テーブル 230 は、ラベル ID テーブル中のエントリが別のラベルから複製されたものかどうか、またはテーブル中の別のエントリに複製されたものかどうかを示すフィールド 236 を含むことができる。

20

【0036】

通常、ラベルはリソースファイル中に保持される。現在のビジネスソリューションシステムは、SQL テーブルなどのデータベースメタデータストアを介して利用可能な包括的なリソースファイルを使用しない。現在のビジネスソリューションシステムでは、これらのラベルは、独自仕様のリソースファイルに記憶される。独自仕様のリソースファイルを使用することに関連する問題の 1 つは、開発者がシステム中の 1 つまたは複数のラベルの一部を、このファイルについてのすべてのリソースを有するファイルで置換したいと思っても、システムは、現在のラベルと同じラベルプロパティまたは用語法を有する別のファイルをシステム中で探すことをしないことである。しかし、ビジネスソリューション環境では、ビジネスソリューションシステムは、複数のベンダによって開発された、同じまたは類似する問題に対する多くの様々なソリューションを扱うことが必要とされる。したがって、本発明のラベルシステム 200 は、様々なベンダによってなされたすべての寄与の総和を用いることができる。

30

【0037】

ID 231 フィールドは、システム 200 中の特定のラベルを識別するのに使用される。一実施形態では、この ID は、グローバル一意識別子すなわち GUID である。GUID は、2 つのベンダが 2 つの無関係のラベルに対して同じ識別番号を選択したことによって生じる問題を回避するために使用される。一実施形態の GUID は、一意識別子が必要とされるどんな場合にもすべてのコンピュータおよびネットワークにわたって使用することのできる 128 ビット整数 (16 バイト) である。このような識別子システムを使用することにより、2 つのラベルが同じ ID を有する可能性が減少する。GUID は、一実施形態ではストリングとして表され、以下のパターンに従ってフォーマットされる。

40

x x x x x x x x - x x x x - x x x x - x x x x - x x x x x x x x x x x x

【0038】

このフォーマットで、GUID の値は、一連の 8、4、4、4、12 デジットのグループによる小文字の 16 進数ディジットとして表され、ハイフンで区切られる。例えば、行 301 のエントリに対する GUID 戻り値は、382c74c3 - 721d - 4f34 - 80e5 - 57657b6cb c27 とすることができる。ただし、その他のフォーマットタイプを ID フィールド 231 に使用してもよい。新しいラベルが生成されるたびに、新しい GUID が割り当てられる。一実施形態では、新しいラベルは、システム 200

50

にとって新しいラベルとして定義され、単に既存のラベルの変換ではない。しかし他の実施形態では、既存のラベルの変換に対して、ＩＤフィールド２３１中の新しいエントリが生成される。

【００３９】

ＧＵＩＤの使用は、現在のシステムに勝る付加利益をもたらす。第１に、各ラベルを複数の言語に分ける必要がない。さらに、ＧＵＩＤにより、行番号の使用など他の方法で一意性を制御する必要なしに、各ラベルを一意に識別することができる。第２に、ＧＵＩＤにより、ラベルの物理記憶域を、独自仕様のリソースファイルからメタデータデータベースやウェブサービスなどの共通リソースファイルに変更することができる。

【００４０】

ラベルＩＤテーブル２３０中の名前空間フィールド２３２は、変換者がラベルテキストをマスタ言語からターゲット言語に変換するときにラベルについての正しい語句を得るのを助けるために、ラベルＩＤに入力される特別なフィールドである。現在のラベルシステムでは、特定のラベルがプログラム中のどこで使用されるかを容易に知ることは不可能である。さらに、ラベルがどの領域で使用されるかを知ることは容易ではない。したがって、プログラムを変換者のコンピュータにインストールして、ラベルがどこで使用されるかを知ることができるようにならない限り、変換者がラベルについての正しい語句を得ることは、不可能とは言わないまでも極度に難しい。名前空間フィールド２３２は、プログラムを変換者のマシンにインストールする必要なしに、ラベルが使用される領域を知ることを可能にする。名前空間フィールドに含まれる情報は、新しいラベルが作成されるときに開発者によって提供され（自動または手動で）、特定のラベルの使用法に関する情報をもたらす。例えば、図３Ａの行３０２のエントリ中で、名前空間フィールド２３２は、このラベルが課金モジュール中で使用されることを変換者または開発者に伝える。

【００４１】

ラベルＩＤテーブル２３０中のカテゴリフィールド２３４は、プログラムのどのコンポーネントでラベルが使用されるかを示す特別なフィールドである。さらに、カテゴリフィールド２３４の使用は、開発者がラベルテキストの作成中にラベルを正しい方式で記述するのに役立つ。カテゴリは、ラベルについてのノードタイプとプロパティの組合せである。ノードタイプは、ラベルの特別なプロパティである。これらのラベルプロパティのいくつかには、「ラベル」「ヘルプ」「キャプション」などを含めることができる。したがって、カテゴリフィールド２３４は、ラベルが所望のプログラムのために正しい方式使用されるようにすることを可能にする。さらに、カテゴリは、システム中に存在するすべてのコントロールのマッピングである。したがって、所望のラベルのタイプまたはカテゴリに基づいて、既存のラベルデータベースを検索することが可能である。本発明では、システム２００によって実施される各規則またはコントロールごとに、１つのカテゴリが作成される。モジュールの開発中、特定のラベルテキストに特別な規則が必要な場合は、追加のカテゴリを作成することができる。例えば、ノードフィールド２３４は、エントリ３０３のラベルが、ユーザをモジュール中の別のポイントに向けるためにモジュールのメニューバー上で使用されるものであることを伝えるエントリとすることができる。各カテゴリは、システム２００中の特定の機能にマッピングされる。

【００４２】

説明フィールド２３５は、ラベルがどのように使用されるかをユーザまたは開発者に説明する、テーブル２３０中のエントリである。例えば、説明フィールド２３５は、エントリ３０３のラベルが台帳のケース上で使用されるものであることを伝えるエントリとすることができる。この説明は、言葉（プレーンテキスト）によるものでもよく、所定のコードリストに対してコード化されたものでもよい。

【００４３】

複製元フィールド２３６は、エントリの関連ラベルがラベルシステム中の別のラベルから複製されたものかどうかを示す。ラベルが別のラベルから複製されたものである場合は、複製元フィールド２３６は、親またはマスタラベル２２０からのＩＤを含む。一実施形

10

20

30

40

50

態では、このIDは親のGUIDである。ただし、その他のIDをフィールド236で使用してもよい。ラベルが別のラベルの複製バージョンでない場合は、複製元フィールド236は空欄であるか、ヌルに設定される。さらに、ラベルのテキストがコピー後にマスタラベルのテキストから変更された場合は、複製元フィールド236はヌルに設定され、それにより、マスタラベル220とこの特定の子ラベルとの間のどんなリンクも除去される。ただし、複製元フィールド236は、エントリに対するその他の変更によってリセットされることになる。代替実施形態では、図3AのラベルIDテーブル230は、どのラベルIDが複製済みラベルを含むかを示すフィールドを含む。IDテーブル230中のラベルは、マスタラベルと子のどちらかである。

【0044】

10

図3Bに、特定のラベルのラベルテキストテーブル240を占めるフィールドを示す。ラベルテキストテーブル240のエントリは、少なくとも3つの異なるフィールドを含む。すなわち、ラベルIDフィールド244、ラベルテキストフィールド241、編集日242である。本発明の代替実施形態では、ラベルテキストテーブル240に追加フィールドを加えることもできる。これらのフィールドには、テキストが書かれた言語を示すフィールド243や、ラベルテキストテーブル中のテキストの各バージョンのエントリ識別子を識別するフィールドが含まれる。

【0045】

ラベルテキストテーブル240のテキストフィールド241は、システムのすべての利用可能言語による、ラベルのテキストの最新バージョンを含む。このラベルについてのラベルテキストテーブル240中の第1のエントリ351は、このラベルが書かれた元の言語またはマスタ言語によるものである。このテキストは、マスタテキストと呼ばれる。このラベルがベストプラクティスガイドラインに従って開発される場合、マスタテキストは米国英語で書かれることになる。ただし、その他の言語をマスタ言語として使用してもよく、ベストプラクティスガイドラインに従う必要はない。一般に、マスタ言語はシステムが動作している現在言語に対応することになる。

20

【0046】

ラベルテキストがマスタ言語から別の言語に変換されるとき、新しいラベルテキストエントリ、例えば352や353が、ラベルテキストテーブル中に作成される。これらのエントリは、元のラベルの変換済みバージョンをそれぞれの言語で含む。例えば、ラベルの元のテキストが「Do you want to save the changes you made to」である場合、このテキストは、ラベルテキストテーブル240の第1行351に記憶される。その後、このテキストがドイツ語、デンマーク語、フランス語に変換されるとき、これらの変換が、ラベルテキストテーブル中でマスタテキストエントリの下のエントリに入力される。これらの追加の変換を、エントリ352、353、354で示す。

30

【0047】

編集日フィールド242は、ラベルテキストテーブル中のその特定のエントリに関連する言語でラベルのテキストが最後に変更されたのはいつかが開発者にわかるように、ラベルテキストテーブル240中に設けられる。さらに、「修正者」フィールドなど、妥当性検査情報を含むその他のフィールドを追加することもできる。この日付は、テキストについての変換が提供された場合に、この変換がそれよりも新しいバージョンを置換しないようにするようにするのに役立つ。変換が自動的にシステムにロードされると、変換日が比較され、エントリ中のバージョンの方が提案エントリよりも新しい場合は、提案エントリはテーブルに入力されない。また、編集日フィールド242により、開発者は、変換がマスタテキストエントリの最新バージョンと同じく最新であるかどうかをチェックすることもできる。一実施形態では、マスタテキストエントリ351の編集日は、特定ラベルについてのラベルテキストテーブル240中で最も古い日付であることが望ましい。

40

【0048】

いくつかの実施形態では、ラベルテキストテーブル240は、言語フィールドエントリ

50

243を含む。言語フィールドエントリ243は、ラベルテキストテーブル中の特定エントリの特定言語を、開発者がその言語を喋ることや理解することがない場合であっても開発者に示す。言語フィールドエントリ243中のこの言語標識は、言語の数値表現としてもよく、言語名として記述してもよく、その他の任意の種類の言語識別子としてもよい。

【0049】

数値表現を使用して言語が識別される場合は、言語の参照番号またはエントリは、例示として、ASCII言語コードやISO (International Organization for Standardization) 639などの知られている規格に準拠するものとすることができる。ただし、その他のコードを使用してテキストエントリの言語を識別してもよい。言語名がフィールド243に入力される場合は、言語名は、例示として、知られている規格(すなわち英語の言語名)に従って記述することができる。ただし、その他のフォーマットを使用してもよい。

【0050】

ラベルテキストテーブル240は、特定のラベルテキストについてのラベルID231を識別するエントリフィールド244も含む。このエントリにより、開発者は、現在のテキストがどのラベルID231に関係するかを知ることができる。ラベルテキストテーブル240の他の実施形態は、テキストID245のためのエントリを含む。このテキストID245は、各テキストラベルをラベルテキストテーブル240中でそれ自体のエントリとして個別に識別するために提供される。テキストID245のエントリは、GUIDとすることもでき、あるいは、ラベルテキストテーブルの構成に適合するその他の任意の識別方法とすることもできる。代替実施形態では、ラベルテキストテーブル240とラベルIDテーブル230を1つのテーブルまたはデータベースにマージすることができる。この実施形態では、ラベルを管理するため、およびモジュールの実行中に正しいラベル言語テキストが表示されるようにするために、追加のフィールドセットが必要になる。さらに、ラベルのロードを管理するためのインデックスも追加される。

【0051】

図4は、ラベルシステムインタフェース400を構成するコンポーネントを示すブロック図である。ラベルシステムインタフェース400は、開発者401をメタデータストア409にリンクして、開発者がシステム200のための新しいモジュールを開発するときに既存のラベルを扱えるようにするインタフェースである。ラベルシステムインタフェース400は、ラベルダイアログ402、ラベルダイアログロジックコンポーネント403、拡張言語インタフェース404、ラベルインタフェース405を備える。

【0052】

開発者は、ラベルダイアログ402を介してラベルシステムインタフェース400と対話する。ラベルダイアログ402は、開発者からラベルシステムインタフェースの諸機能へのアクセスを可能にするユーザインタフェースである。一実施形態では、ラベルダイアログ402は、ユーザが特定のラベルの使用法を閲覧および管理することができるようにするウィンドウである。ラベルダイアログ402は、メタデータストアへのこのインタフェースを使用して、ラベルを扱い、すべての利用可能言語のラベルのために利用できるコマンドの完全なセットにアクセスする。これらの機能を提供するために、ラベルダイアログインタフェース402は、ラベルシステム400中の各ラベルにつき、すべての利用可能言語へのアクセスを必要とする。

【0053】

しかし、ラベルダイアログ402は、ラベルを特定の状況で 사용할 ことができるかどうかを判定するためのロジックは含まない。すべての制御は、ラベルインタフェースまたはラベルダイアログロジックコンポーネント403を制御するクラスに渡される。ラベルダイアログロジックコンポーネント403は、ラベルダイアログ402と共に使用されるように設計されているが、ラベルダイアログ402のためのすべてのロジックを含むので、ラベルダイアログロジックのいくつかの機能は、他の領域で再利用することができる。さらに、ラベルダイアログロジックコンポーネント403は、各ラベルがそのラベルにつ

いての名前空間とカテゴリの正しい組合せに適合するような形で使用されることを確実にする。このロジックは、最初に新しいマスタラベルを作成せずにラベルを別の領域でうっかり使用してしまうことを防止する。

【 0 0 5 4 】

拡張言語インタフェース 4 0 4 は、ラベルシステムインタフェース 4 0 0 からシステム中で利用可能なすべての言語へのアクセスを提供する。拡張言語インタフェース 4 0 4 は、別のクラスを使用して、複数の言語をラベルダイアログロジックコンポーネント 4 0 3 およびラベルダイアログ 4 0 2 に利用可能にする。拡張言語インタフェース 4 0 4 は、システム中の複数の言語に共通のメソッドだけを有する（すべての利用可能言語のリストを返すメソッド、また、現在の言語の ID を返すメソッド）。1 つの言語だけに関係するメソッドは、ラベルクラス 4 0 5 上に直接に位置する。ラベルクラスは、メタデータ中のラベルに接続することを可能にする。利用可能な各言語中に、ラベルクラスのインスタンスが 1 つある。ただし場合によっては、特定のクラスにはインスタンスがないこともある。

10

【 0 0 5 5 】

言語インタフェースコンポーネント 4 0 4 はまた、利用可能言語へのアクセスを制御する。言語インタフェースコンポーネント 4 0 4 は、ラベルのインスタンスのマップを含む。言語インタフェースコンポーネント 4 0 4 中にはまた、現在のラベルについての追加の利用可能言語クラスそれぞれへのインタフェースもある。

【 0 0 5 6 】

図 5 は、本発明の一実施形態による、モジュールの一部の開発中に開発者がモジュール用に新しいラベルを作成するときに実施されるステップを示すフローチャートである。例えば、開発者がモジュール中で「customer」についての入力を表すラベルを作成したい場合を考える。開発者は、このラベルがどのようにモジュール中で使用されることになるかを判定しなければならない。語句「customer」は、多くの様々な意味を持つ。例えば、customer は、自分から物品およびサービスを購入する人を意味する場合もあり、自分が対処する「人」を意味する可能性もある。この単語の使用法は、他の言語のテキストにも影響を与えることになる。図 5 のブロック 5 0 1 にこれを示す。

20

【 0 0 5 7 】

次に、開発者はラベルダイアログプログラムを開く。開かれるラベルダイアログは、例示として、図 6 に示すラベルダイアログ 6 0 0 と同様のものである。ただし、その他のインタフェースを使用してもよい。提示されたインタフェースにより、開発者は、ラベルに対して望む特定のテキストを入力することができる。図 5 のブロック 5 0 2 に、ラベルダイアログを開くことを示す。

30

【 0 0 5 8 】

ラベルダイアログを開くと、次いで開発者は、ラベルに対する所望のテキストを行 6 0 2 に入力する。あるいは開発者は、行 6 0 2 に、ラベルに対する所望のテキストの一部を入力するのでもよい。例えば、開発者は、テキストセクションに「customer」と入力してもよく、「cust」と入力してもよい。さらに、開発者は、新しいラベルがどのように使用されることになるかを示すデータを、インタフェース中の行 6 0 3 および 6 0 4 に入力する。このデータは、ラベルシステム中で既存のラベルを検索するのに使用することもでき、新しいラベルを作成するときに使用することもできる。例えば、開発者が販売管理用のモジュールを作成している最中であり、語句「customer」を使用して購入者についてのラベルを生成したい場合、開発者は、購入者についてのカテゴリコードを行 6 0 4 に入力することになる。このコードは、手動で入力してもよく、プルダウンメニューで入力してもよく、現在のシステム設定を使用して自動的に入力されるものとしてもよい。このカテゴリコードが、この機能によって使用されるプロセスの残りを制御する。一般に、ラベルダイアログ中のカテゴリコードおよび名前空間のエントリは、ラベルシステムの現在設定に従う。図 5 のブロック 5 0 4 に、データの入力を示す。このデータは、通常の表現として入力される。さらに代替実施形態では、開発者は、選択されたカテゴリに検索を限定するチェックボックスを選択することによって、検索を制御することが

40

50

できる。

【 0 0 5 9 】

開発者は、ボタン 6 5 0 を選択することによって検索機能をアクティブにする。ただし、その他の技法を使用してラベル検索機能をアクティブにしてもよい。図 5 のブロック 5 0 5 に、検索エンジンのアクティブ化を示す。

【 0 0 6 0 】

次いで本発明は、データベース中を検索して、入力されたテキストにマッチするエントリをデータベース中で見つける。検索プロセスの間、進行状況インジケータをユーザに対して表示することができる。進行状況インジケータの 1 つを図 6 の参照番号 6 4 0 で示す。検索が完了すると、進行状況の指示は消え、ダイアログは、選択された言語による最初のクエリにマッチしたすべてのラベルの表示を開発者に返す。これをブロック 5 0 6 に示す。

10

【 0 0 6 1 】

インタフェース 6 0 0 は、一実施形態では、図 6 に示すように、識別されたマッチを表示するように拡大する。この実施形態では、表示されるマッチのリストは、ラベル ID テーブル 2 3 0 に含まれる情報、ならびにラベルテキストテーブル 2 4 0 からのいくつかの情報を含むことができる。結果ビューは、選択された言語のテキストを、テーブル 2 3 0 との関係を通して入手可能な追加情報と組み合わせて表示する。検索結果は、GUI による昇順で開発者に提示される。図 6 の領域 6 1 0 に、結果の例を示す。ただし、テキストマッチングによる昇順など、その他の順序付け技法を使用してもよい。次いで開発者は、ラベルテキストのいずれかが所望のテキストにマッチするかどうかをチェックする。これをブロック 5 2 0 に示す。新しいラベルに対する所望のテキストにマッチするテキストが見つかった場合は、開発者は、所望のラベルテキストにマッチするこの特定ラベルを強調表示するか、または他の方法で示す。これをブロック 5 0 7 に示す。望むなら、ラベルの変換を 6 2 0 に表示することもできる。

20

【 0 0 6 2 】

次にシステムは、選択されたラベルカテゴリおよび名前空間が新しいラベルのカテゴリと同じかどうかをチェックする。これをブロック 5 0 8 に示す。選択されたラベルと新しいラベルとでラベルカテゴリおよび名前空間が同じである場合は、選択されたラベルの情報を新しいラベルに使用する。これを図 5 のブロック 5 1 1 に示す。

30

【 0 0 6 3 】

ラベルの名前空間およびカテゴリが新しいラベルの所望の使用法と同じでない場合は、ラベルを複製しなければならない。この複製される情報には、選択されたラベルのすべての変換済みバージョンも含まれる。ラベルが新しいラベルに複製されるとき、新しいラベルの複製元ラベルの GUI D を示すエントリが、新しいラベルについてラベル ID テーブル 2 3 0 中に作成される。これをブロック 5 1 0 に示す。これにより、マスタラベルテキストが変更されたとき、新しいラベルのテキストを容易に更新することが可能になる。この変更時更新は、通常の検索置換プロトコルを実行することによって行うことができ、あるいはその他の任意の自動化方法によって行うことができる。

【 0 0 6 4 】

40

検索中にマッチが見つからなかったときは、開発者は、新しいラベルのための新しいオブジェクトを生成しなければならない。ブロック 5 2 1 に、新しいオブジェクトの生成を示す。ラベルの生成中に、ラベルの特定の特性をラベル ID テーブル 2 3 0 およびラベルテキストテーブル 2 4 0 に記憶する。ラベルのテキストが不完全な場合は、開発者は、所望のテキストの残りの部分をラベルダイアログ 6 0 0 に入力する。名前空間およびカテゴリコードは、ラベルシステムの現在設定に基づいて入力される。開発者はまた、所望の言語があればその言語でラベルテキストについての変換も生成しなければならない。また、ラベルの現在の動作言語の ID も、マスタ言語として記憶する。ブロック 5 2 2 に、これらの変換の追加または生成、およびラベルプロパティの入力を示す。

【 0 0 6 5 】

50

最後に、本発明は、ビジネスソリューションソフトウェアプログラムにおいてラベルを記憶および使用するための新しい方法も対象とする。各言語につき単一のリソースファイルがあるのではなく、情報は1:nの関係に分割される。関係の「1」の側は、ラベル識別子、およびその他の実質的な一般ラベル情報を保持する。関係の「n」の側は、特定言語のラベルテキストを保持する。このようにしてラベルを構成することにより、すべての言語のラベルにランタイムにアクセスすることが可能になる。

【0066】

新しいラベルの開発中は、システム中の既存のラベルを使用する、あるいは新しいラベルを一から作成してラベルを複製することでもできる。ラベルを検索するとき、表現を入力することが可能である（例えば「<Ledger」と入力すると、Ledgerで始まるすべての利用可能ラベルが提供される。ただし、システムがどこで実施されるかに応じて、異なる構文を使用することができる）。検索される特定の名前空間またはカテゴリを選択することによって、ヒットを少なくすることが可能である。検索はまた、語句データベースを使用して実施することでもできる。また、語句データベース（ウェブサービス）を使用して新しいラベルを作成することでもできる。語句データベースに入力するときは、現在の検索基準を、ラベルが使用される実際の状況を記述する追加の基準と組み合わせることが可能である。これは、正しい語句が使用されることを確実にするためである。ラベルが語句データベース中で見つかり、これが新しい識別子に複製される。語句データベースは、サポートされるすべての言語についての変換を有する。ラベルが使用されることになる場合、ラベルは、実際の名前空間およびカテゴリにマッチして検索結果中に存在しなければならない。そうである場合は、この特定のラベルを選択することが可能である。そうでない場合は、ラベルテキストを手動で記述することによって、またはラベルを語句データベースの別の名前空間/カテゴリから複製することによって、新しいラベルを作成する。ラベルを複製すると、現在の名前空間およびカテゴリに関する新しい識別子が提供される。すべてのテキストおよびその他の一般情報が複製される。

【0067】

ラベルの管理はラベルダイアログを使用して行われ、ラベルダイアログは、特定のラベルプロパティから直接に呼び出される。ダイアログは、ラベルを維持すること、および特定のプロパティに対して使用される特定のラベルを選択することを可能にする。必要とされる機能を提供するラベルインタフェースを介して、コードによるラベルシステムの使用がもたらされる。これにより、ラベルの記憶方法を変更することが可能である。記憶域は、SQLメタデータストア、リソースファイル、またはウェブサービスとすることができ、実際、ウェブへの具体的なアクセスに応じて、あらゆるタイプの記憶域の組合せとすることができる。

【0068】

ラベルダイアログは、ラベルが正しい状況で使用されることを確実にする。ラベルシステムにカテゴリを導入することにより、ラベルが正しく使用されることが保証される。ただし、この機能には別の目的もある。ラベルを変換するとき、名前空間情報とカテゴリの組合せを介して、ラベルがどこで使用されることになるかを知ることが可能である。これにより、使用される正しい語句を知ることが可能である。ラベルがどこで使用されるかを知ることにより、より多くのベストプラクティスチェックをベストプラクティスフレームワークに追加することが可能である。本発明では、既知の（または未知の）変換ツールに適合する特定の変換ファイルを生成することができる。日付情報を変換ファイルにエクスポートすることによって、変換をインポートするときに日付をチェックすることが可能である。これは、進行中の変換プロセスが、システム中の最後の変更と同じく最新であることを確実にするために行う。更新される相互参照システムを使用して、特定のラベルがどこで使用されるかを知ることが可能である。また、特定のラベルに対する変更を知ることでも可能である。この機能は変換者に関係があり、特定のラベルに対する変更が単純かどうか（例えば行の最後に「.」が追加されるなど）を変換者が判定できるようにする。

【0069】

本発明を特定の実施形態に関して述べたが、本発明の趣旨および範囲を逸脱することなく形式および詳細に変更を加えることもできることは、当業者なら理解するであろう。

【図面の簡単な説明】

【 0 0 7 0 】

【図 1】本発明を使用することのできる例示的な一環境のブロック図である。

【図 2】本発明のラベルシステムを構成するテーブルを示すブロック図である。

【図 3 A】ラベル I D テーブル中およびラベルテキストテーブル中のフィールドの概略図である。

【図 3 B】ラベル I D テーブル中およびラベルテキストテーブル中のフィールドの概略図である。

【図 4】ラベルダイアログのコンポーネント間の関係を示すブロック図である。

【図5】新しいラベルを作成するときの実行されるステップを示す流れ図である。

【図 6】ラベルデータベースを開発および検索するときにユーザによって呼び出されるユーザインタフェースの例を示す図である。

【符号の説明】

【 0 0 7 1 】

2 1 0 言語

2 1 2 名前

2 2 0 マスタ

2 3 0 ラベル I D

2 3 2 名前空間

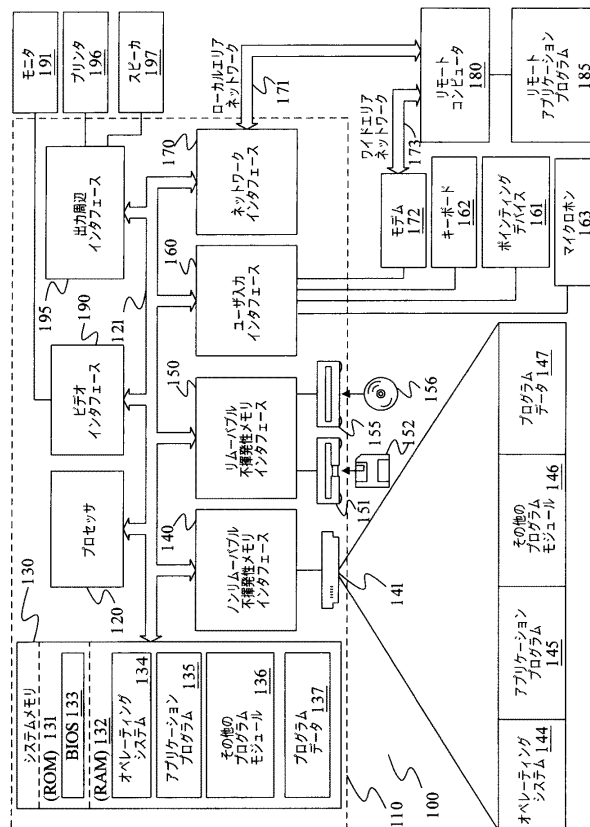
2 3 4 カテゴリ

2 3 5 説明

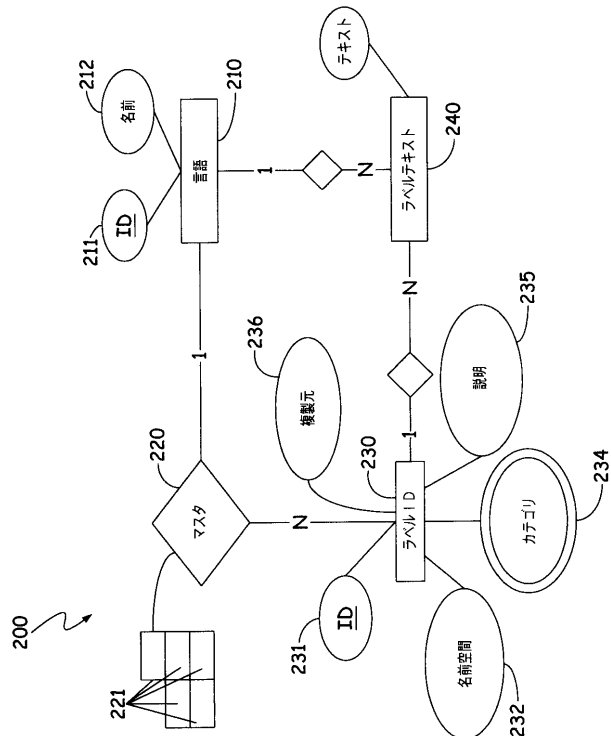
2 3 6 複製元

2 4 0 ラベルテキスト

【 図 1 】



【圖 2】



【図 3 A】

ラベルID 230

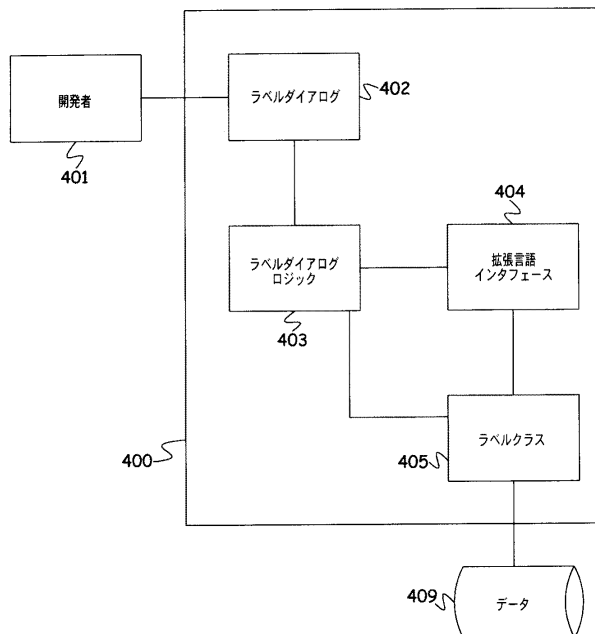
231 GUID	232 名前空間	234 カテゴリ	235 説明	236 複製元
301 1				
302 2	課金モジュール			
303 3		メニューバー	右側のケースで 使用される	1

【図 3 B】

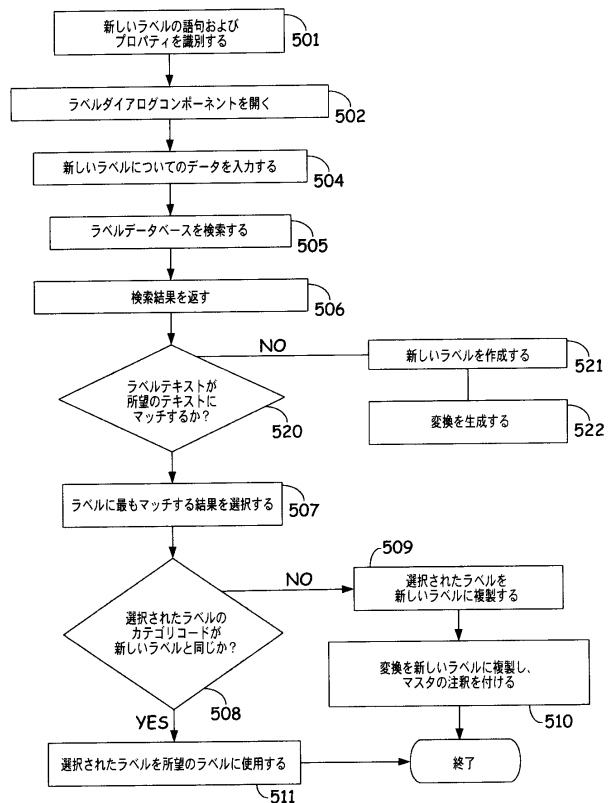
ラベルテキスト 240

244 ラベルID	245 テキストID	241 テキスト	242 編集日	243 言語
1	351	DO YOU WANT TO SAVE THE CHANGES YOU MADE TO	22-AUG-2003	ENGLISH - UNITED STATES
1	352	MÖCHTEN SIE DIE ÄNDERUNGEN GEBILDET AN SPEICHERN	27-AUG-2003	GERMAN - DEUTCH
1	353	LAVE JER VILLE GERNE OPSPARE DEN FORANDIGERJER INDRETTET TIL AT	27-AUG-2003	DANISH - DANSK
1	354	VOUS VOULEZ EN SAUVER LES CHANGEMENT VOUS FAIT A	27-AUG-2003	FRENCH - FRANCAIS

【図 4】



【図 5】



【 6 】

600

602

Find: iCust

Language: en-us

Category:

Namespace:

650 Find now

604 Stop

603

640

Select

New

Duplicate

ID	Language	Text	Description	Name	Space	Category
@GLS.. en-us		CustGroup				
@SYS.. en-us		Custom				
@SYS.. en-us		Custom-made reports creat...				
▶ S56520 en-us		Custom...				
@SYS.. en-us		Customer				
@SYS.. en-us		Customer				
@SYS.. en-us		Customer				
⋮						
ID	Language	Package				Category
▶ @SYS...	da	Bruggerdefineret...				
@SYS...	fr	Personnaliser...				
⋮						
Label(s) found						

610

620

フロントページの続き

(72)発明者 ヤコブ ステーン ハンセン
デンマーク ディーケー - 2 9 8 0 コッケダル トルネヘブン 9 ビー

審査官 稲垣 良一

(56)参考文献 米国特許第 6 4 9 3 6 6 1 (U S , B 1)
特開 2 0 0 1 - 2 2 2 4 1 6 (J P , A)
特開平 5 - 2 9 8 0 8 2 (J P , A)
Ewan Smith et al. , A Multi-Tiered Classification Scheme for Component Retrieval , In Pr
oceedings of 24th Euromicro Conference , IEEE , 1 9 9 8 年 8 月 2 5 日 , Vol.2 , pp.882-88
9

(58)調査した分野(Int.Cl. , D B 名)
G 0 6 F 9 / 4 4
G 0 6 F 1 7 / 3 0