



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2012년03월09일  
(11) 등록번호 10-1114189  
(24) 등록일자 2012년02월01일

(51) 국제특허분류(Int. Cl.)  
*G06Q 10/00* (2006.01) *G06F 9/44* (2006.01)  
*G06F 9/00* (2006.01)

(21) 출원번호 10-2004-0060481  
(22) 출원일자 2004년07월30일  
 심사청구일자 2009년07월27일  
(65) 공개번호 10-2005-0031872  
(43) 공개일자 2005년04월06일

(73) 특허권자  
**마이크로소프트 코포레이션**  
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이

(72) 발명자  
**비츠렌느**  
덴마크 디케이-3200 헬신게 램로스 아레스가드 엘리 8

**핸센자콥스틴**  
덴마크 디케이-2980 코케달 토네하벤 9 비

(74) 대리인  
**제익특허법무법인**

CS0493001 B1  
KR1020020097147 A  
KR1020010112031 A  
전체 청구항 수 : 총 17 항

(74) 대리인  
제일특허법이

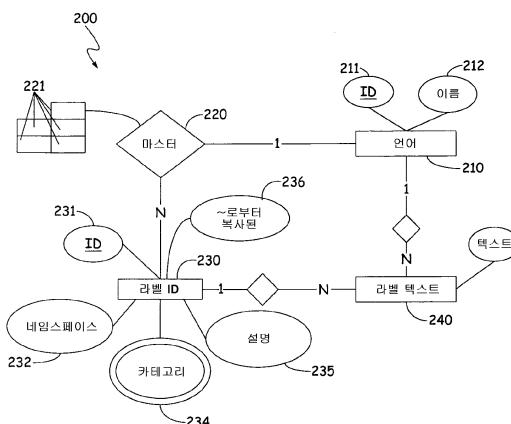
심사관 : 임영국

(54) 발명의 명칭 라벨 시스템 - 실행 및 설계 중 텍스트 번역 및 다중 언어지원

### (57) 요약

본 방법은 다수의 언어들을 갖는 비지니스 솔루션 소프트웨어 프로그램 내에서 라벨을 저장하고 사용한다. 각 언어마다 하나의 리소스 파일을 갖는 대신에, 라벨에 관련된 정보가 데이터베이스에 저장된다. 각 라벨은 라벨 ID 구조 내의 고유 식별자로 할당되고, 시스템 내에서 라벨이 사용되는 방식에 관한 정보 역시 포함한다. 라벨에 대한 텍스트는 별도의 라벨 텍스트 구조에 저장된다. 라벨 텍스트 구조는 사용 가능한 언어들 모두 내의 라벨의 텍스트를 포함한다. 새로운 라벨이 기재되면, 개발자는 의도된 사용에 맞는 라벨을 찾기 위해 현재의 라벨을 검색한다. 만약 원하는 텍스트에 대한 일치가 발견되면, 라벨은 새로운 라벨로 복사된다. 만약 어떠한 일치도 발견되지 않으면, 새로운 라벨이 생성되어야 한다.

## 대 표 도



## 특허청구의 범위

### 청구항 1

컴퓨터로 구현된 비즈니스 솔루션 소프트웨어 시스템(business solution software system)에서 새로운 라벨을 생성하는 컴퓨터 구현 방법으로서 - 상기 새로운 라벨은 상기 비즈니스 솔루션 소프트웨어 시스템과 관련된 사용자 인터페이스 내의 컨트롤을 식별하도록 구성된, 컴퓨터로 구현된 사용자 인터페이스 구성요소임 -,

인터페이스에서, 상기 새로운 라벨을 위한 원하는 텍스트를 표시하는 데이터를 수신하는 단계;

상기 원하는 텍스트와 부합하는(matching) 텍스트를 포함하는 기존 라벨들의 표시들(indications)에 대한 라벨 텍스트 데이터베이스를 검색하는 단계 - 상기 라벨 텍스트 데이터베이스에 표현된 기존 라벨들은 컴퓨터로 구현된 사용자 인터페이스 구성요소임 -; 및

상기 라벨 텍스트 데이터베이스의 상기 검색의 결과에 적어도 부분적으로 기초하여, 상기 원하는 텍스트와 부합하는 텍스트를 포함하는 기존 라벨들의 리스트를 사용자에게 반환하는 단계

를 포함하는 컴퓨터 구현 방법.

### 청구항 2

제1항에 있어서,

상기 새로운 라벨에 대한 새로운 항목(entry)을 상기 라벨 텍스트 데이터베이스 내에 생성하는 단계를 더 포함하는 컴퓨터 구현 방법.

### 청구항 3

제2항에 있어서,

상기 새로운 항목을 생성하는 단계는,

상기 새로운 라벨에 대한 전역 고유 식별자(GUID)를 할당하는 단계; 및

상기 새로운 라벨과 연관된 텍스트에 대한 레코드를 상기 라벨 텍스트 데이터베이스 내에 생성하는 단계를 포함하는 컴퓨터 구현 방법.

### 청구항 4

제3항에 있어서,

상기 인터페이스에서, 상기 새로운 라벨에 대한 카테고리 코드(category code)를 표시하는 데이터를 수신하는 단계; 및

상기 인터페이스에서, 상기 새로운 라벨에 대한 설명(description)을 표시하는 데이터를 수신하는 단계

를 더 포함하는 컴퓨터 구현 방법.

### 청구항 5

제4항에 있어서,

상기 인터페이스에서 설명을 표시하는 데이터를 수신하는 단계는 네임스페이스(namespace)를 수신하는 단계를 포함하는 컴퓨터 구현 방법.

### 청구항 6

제4항에 있어서,

상기 인터페이스에서, 상기 새로운 라벨에 대한 원래의 언어를 표시하는 데이터를 수신하는 단계를 더 포함하는 컴퓨터 구현 방법.

### 청구항 7

제2항에 있어서,

상기 라벨 텍스트 데이터베이스 내에 상기 새로운 라벨을 저장하는 단계를 더 포함하는 컴퓨터 구현 방법.

### 청구항 8

제1항에 있어서,

상기 인터페이스에서, 상기 새로운 라벨이 어떻게 이용되어야 하는지 표시하는 데이터를 수신하는 단계를 더 포함하는 컴퓨터 구현 방법.

### 청구항 9

제8항에 있어서,

상기 원하는 텍스트와 부합하는 텍스트를 포함하는 기존 라벨들의 표시들에 대한 상기 라벨 텍스트 데이터베이스를 검색하는 단계;

상기 선택된 레코드가 어떻게 이용되어야 하는지에 대한 표시를 상기 선택된 레코드를 내에서 식별하는 단계;

상기 선택된 레코드들의 상기 표시된 이용법과 상기 새로운 라벨의 상기 표시된 이용법을 비교하는 단계; 및

상기 원하는 텍스트와 관련하여 부합하는 정도 및 상기 새로운 라벨의 표시된 이용법에 기초하여, 상기 선택된 레코드를 정렬하는 단계

를 더 포함하는 컴퓨터 구현 방법.

### 청구항 10

제9항에 있어서,

상기 기존 라벨들의 리스트를 상기 사용자에게 반환하는 단계는 상기 기존 라벨들의 리스트를 디스플레이하는 단계를 포함하는 컴퓨터 구현 방법.

### 청구항 11

제10항에 있어서,

상기 리스트를 디스플레이하는 단계는, 상기 원하는 텍스트와 가장 근접하게 부합하는 기존 라벨들이 가장 먼저 디스플레이되도록 상기 기존 라벨들의 리스트를 디스플레이하는 단계를 포함하는 컴퓨터 구현 방법.

### 청구항 12

제9항에 있어서,

상기 기존 라벨들의 리스트로부터의 라벨 선택(a selection of label)을 수신하는 단계; 및

상기 선택된 라벨의 상기 표시된 이용법과 상기 새로운 라벨의 상기 표시된 이용법을 비교하는 단계

를 더 포함하는 컴퓨터 구현 방법.

### 청구항 13

제12항에 있어서,

상기 선택된 라벨의 상기 표시된 이용법이 상기 새로운 라벨의 상기 표시된 이용법과 동일하지 않다고 결정하는 단계; 및

상기 라벨 텍스트 데이터베이스 내에서 상기 선택된 라벨을 상기 새로운 라벨로 복제하는 단계

를 포함하는 컴퓨터 구현 방법.

#### 청구항 14

제13항에 있어서,

상기 선택된 라벨에 대한 라벨 텍스트 데이터베이스 내의 임의의 번역(transliteration)을 상기 새로운 라벨에 대한 상기 라벨 텍스트 데이터베이스 내의 레코드로 복제하는 단계를 더 포함하는 컴퓨터 구현 방법.

#### 청구항 15

제13항에 있어서,

상기 선택된 라벨의 ID를 상기 새로운 라벨에 관련시키는 단계를 더 포함하는 컴퓨터 구현 방법.

#### 청구항 16

제12항에 있어서,

상기 선택된 라벨의 상기 표시된 이용법이 상기 새로운 라벨의 상기 표시된 이용법과 동일하다고 결정하는 단계; 및

상기 새로운 라벨에 대해 상기 선택된 라벨을 사용하는 단계

를 포함하는 컴퓨터 구현 방법.

#### 청구항 17

컴퓨터로 구현된 비즈니스 솔루션 소프트웨어 시스템에서 새로운 라벨을 생성하는 컴퓨터 구현 방법으로서 - 상기 새로운 라벨은 상기 비즈니스 솔루션 소프트웨어 시스템과 관련된 사용자 인터페이스 내의 컨트롤을 식별하도록 구성된, 컴퓨터로 구현된 사용자 인터페이스 구성요소임 -,

인터페이스에서, 상기 새로운 라벨이 어떻게 이용되어야 하는지 표시하는 데이터를 수신하는 단계;

기준 라벨들의 표시들에 대한 라벨 텍스트 데이터베이스를 검색하는 단계 - 상기 검색하는 단계는 상기 새로운 라벨이 어떻게 이용되어야 하는지를 표시하는 상기 데이터에 적어도 부분적으로 기초하여 검색하는 단계를 포함하고, 상기 라벨 텍스트 데이터베이스에 표현된 상기 기준 라벨들은 상기 비즈니스 솔루션 소프트웨어 시스템 내에서의 컨트롤을 식별하도록 구성된, 컴퓨터로 구현된 사용자 인터페이스 구성요소임 -; 및

상기 라벨 텍스트 데이터베이스의 상기 검색의 결과들에 적어도 부분적으로 기초하여, 기준 라벨들의 리스트를 사용자에게 반환하는 단계

를 포함하는 컴퓨터 구현 방법.

#### 청구항 18

삭제

#### 청구항 19

삭제

#### 청구항 20

삭제

#### 청구항 21

삭제

#### 청구항 22

삭제

#### 청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

## 명세서

### 발명의 상세한 설명

#### 발명의 목적

#### 발명이 속하는 기술 및 그 분야의 종래기술

[0007] 본 발명은 비즈니스 솔루션 소프트웨어 프로그램에 있어서 라벨들의 병합 및 개발에 관한 것이다. 특히, 본 발명은 비즈니스 솔루션 소프트웨어용 모듈에 있어서 라벨들의 개발 및 병합에 사용되는 라벨 데이터베이스 및 라벨 대화에 관한 것이다.

[0008] 비즈니스 솔루션 소프트웨어 프로그램은 최종 사용자, 일반적으로는 기업에 사용자의 다양한 비즈니스 파트너와의 연계성을 지원하는 맞춤식, 가변식 및 글로벌 기업 리소스 계획 솔루션을 제공한다. 수많은 비즈니스 솔루션 소프트웨어 프로그램들은 원 제품을 초월하여 소프트웨어의 기본적인 기능을 확장할 수 있도록 하여, 이행 기업의 욕구를 보다 충족시키게 된다. 이러한 새로운 또는 추가적인 기능은 비즈니스 솔루션 소프트웨어 내에 포함된 현존 데이터 및 현존 특징을 이용하도록 기록되어진 추가적인 모듈을 통하여 제공된다. 종종 이러한 추가적인 모듈은 소프트웨어 및 현존 데이터를 비즈니스 솔루션 소프트웨어의 신구 기능 모두와 자동으로 동기화 시킨다.

[0009] 일부 비즈니스 솔루션 소프트웨어는 다양한 언어 및 다양한 통화(currency)로 다른 나라에서 비즈니스를 행할 수 있게 한다. 비즈니스 솔루션 소프트웨어에 제공되는 다중 언어 능력을 사용함으로써, 송장(invoice)과 같은 문서를 수령인에게 양호한 언어로 전달할 수 있게 된다. 그러나, 현 시스템에서 문서의 언어를 변환하게 되면, 비즈니스 솔루션 소프트웨어에 새로운 언어를 로딩하여, 상기 시스템의 전반적인 동작 언어를 변환시킬 필요가 있게 된다.

[0010] 비즈니스 솔루션 소프트웨어의 활용가능한 여러 언어들에서의 커뮤니케이션은 다수의 라벨을 사용하여 처리된다. 라벨은 사용자 인터페이스 컴포넌트 또는 인쇄된 문서 상에 제시되는 텍스트이다. 라벨은 메뉴, 버튼, 대화 상자 등의 상에 사용될 수 있다. 또한, 라벨은 "label", "help", "caption" 및 "tool tip"과 같은 라벨 특성을 갖는 제어기(control) 상에 사용될 수 있다. 현 비즈니스 솔루션 소프트웨어에서의 라벨은 상기 비즈니스 솔루션 소프트웨어에 의해 사용되는 각각의 언어에 전용하는 하나의 리소스 파일로 개별 리소스 파일에 저장된다. 또한, 비즈니스 솔루션 소프트웨어에서의 각각의 모듈은 다른 모듈이나 상기 비즈니스 솔루션 소프트웨어 자체와 공유하지 않는 그 자신의 리소스 파일을 갖는다.

[0011] 비즈니스 솔루션 소프트웨어의 개발을 통하여 개발자들 사이에서는 현존 라벨들을 재사용하고자 하는 강한 욕구가 있어 왔다. 그러나, 라벨의 용어와 관련된 다양한 특성이 다른 용도 사이에서 변할 수 있고, 또는 라벨의 의미가 다른 개발자들 사이에서 변화할 수 있기 때문에, 현존 라벨을 재사용하는 것은 어렵지 못하다는 사실을 알게 되었다. 이것은 용어가 변할 때 문제를 일으킨다. 예를 들면, 라벨 텍스트는 어떤 애플리케이션에서는 메뉴로, 그리고 다른 애플리케이션에서는 버튼으로 사용될 수 있기 때문에, 각각의 라벨에 대하여 상이한 특성을 나타낸다. 대부분의 개발자들이 원했던 것은 시스템 상의 다양한 모듈에 대한 라벨을 개발하는 것과 관련된 비용을 절감하기 위하여 라벨을 포함하는 텍스트 또는 용어를 재사용하는 것이었다.

### 발명이 이루고자 하는 기술적 과제

[0012] 상술한 바와 같이, 일반적으로, 비즈니스 솔루션 소프트웨어 모듈에서 라벨은 리소스 파일에 보유된다. 그러나, 현 비즈니스 솔루션 소프트웨어는 구조화된 조회 언어(structured query language; SQL) 테이블과 같은 데이터베이스 메타데이터 저장을 통하여 활용가능한 일반적인 리소스 파일들을 사용하지는 않는다. 일반적으로 이러한 라벨은 비즈니스 솔루션 소프트웨어에 있어서 특정 모듈이고, 독점적인 리소스 파일에 모듈에 있는 각각의 언어에 전용하는 하나의 리소스 파일로 저장된다. 독점적인 리소스 파일을 사용하는 것과 관련된 하나의 문제점은 하나의 모듈에서 라벨의 일부분을 다른 리소스 파일에 포함된 특성이나 새로운 정보로 대체하거나 편집하고자 할 때, 개발 시스템이 소망하는 라벨과 동일한 라벨 특성 및/또는 용어를 갖는 비즈니스 솔루션 소프트웨어의 다른 모듈에서 다른 라벨을 찾지 못한다는 것이다. 또한, 리소스 파일을 사용함으로써, 소프트웨어 솔루션에 있는 상당히 많은 라벨들로 인하여 라벨들의 관리가 극히 어렵게 된다. 그러나, 비즈니스 솔루션 환경에 있어서 비즈니스 솔루션 소프트웨어는 수많은 판매자들에 의해 개발되는 수많은 다양한 솔루션을 처리하는 것이 요구된다. 종종 이러한 모듈의 개발자들은 비즈니스 솔루션 소프트웨어에서 다른 모듈에 대하여 개발된 라벨과 겹치는 라벨을 개발하곤 한다. 라벨 개발 및 라벨 번역(다중 언어 지원이 요구될 때)과 관련된 비용은 비싸고, 특히 라벨과 그 번역이 비즈니스 솔루션 소프트웨어의 다른 곳에 이미 존재할 때에는 시간을 소모하는 것이 된다. 따라서, 각각의 판매자가 각각의 소망하는 라벨에 대하여 그 자신의 라벨을 개발하도록 하는 완전히 독점적인 시스템과 달리, 다양한 모듈 판매자들의 기여(contribution)를 서로 결합시킨 라벨 시스템을 갖는 것이 바람직하다. 또한, 시스템이 현존 라벨 텍스트를 검색하는 것을 가능하게 하여, 라벨이 사용되는 문맥(context)이 원래 만들어졌을 때의 라벨의 용도와 다른 문맥을 여전히 처리하는 동안 상기 텍스트를 재사용할 수 있도록 하는 것이 바람직하다.

### 발명의 구성 및 작용

[0013] 본 발명은 비지니스 솔루션 시스템에서의 모듈 개발자(developer)의 필요성을 다룬다. 종전 시스템과는 달리, 라벨(즉 디스플레이 텍스트)은 공통 자원 파일에 보관되고, 공통 자원 파일에서는 시스템의 각 모듈들이 각 모듈의 독점 자원 파일 대신에 같은 자원 파일을 액세스한다. 또한, 본 발명의 능률을 향상시키기 위하여, 모든 언어 텍스트는 원어(original language)와 동일한 자원 파일 또는 데이터 베이스에 저장된다. 모듈에 의해 사용되는 각 라벨은, 전역 고유 식별자(Globally Unique Identifier, GUID), 그 이용의 기술, 및 관련 카테고리 및 네임스페이스와 같은 고유 식별자에 의해 라벨 테이블 내에서 식별된다. 이 정보는 새로운 라벨이나 모듈을 생성할 때 데이터 베이스 내에 존재하는 라벨이 유용한지 여부를 개발자가 결정할 때 도움이 된다.

[0014] 각 라벨은 또한 라벨 텍스트 데이터 베이스 내에 엔트리의 집합을 가지며, 라벨 텍스트 데이터 베이스는 복수의

언어로 라벨 텍스트의 버전 및 라벨 텍스트를 관리하는데 유용한 기타 정보들을 나타낸다. 라벨 텍스트의 원어는 마스터 언어로서 식별된다. 사용자가 비지니스 소프트웨어 솔루션 시스템의 언어를 변경하길 원하거나 모듈의 언어를 변경하길 원하거나 다른 언어로 문서를 출력하길 원한다면, 라벨 텍스트 테이블은 새로운 언어의 집합을 시스템에 로딩하거나 선호하는 시스템 언어를 일시적인 언어로 바꾸지 않고도 가용 언어로의 액세스를 가능하게 한다. 이것은 전체 시스템으로 하여금 새로운 언어에서 동작하도록 요구함으로써 사용자를 혼란스럽게 하는 일 없이 언어를 신속하게 변화시키는 것을 가능하게 한다. 또한, 시스템은 현재 활성화된 언어를 디스플레이하도록 항상 시도한다. 만약 현재 활성화된 언어가 라벨에 대하여 실패한다면, 폴 백 언어(fall back language)가 이용될 수 있다.

[0015] 본 발명의 일 실시예는 또한, 개발자가 비지니스 솔루션 소프트웨어에 대한 새로운 모듈을 개발하거나 기존 모듈을 편집하는 경우를 다룬다. 본 발명은 개발자로 하여금 라벨 대화 개발 툴을 통하여 기존 라벨을 식별하는 것을 가능하게 한다. 먼저, 개발자는 개발 디스플레이에서 사용자 인터페이스를 연다. 이 디스플레이에는 라벨 텍스트 데이터베이스를 통한 검색 및 모든 가용 라벨로의 액세스를 가능하게 하는 검색 기능을 개발자에게 디스플레이한다. 개발자는 원하는 라벨 텍스트를 검색 엔진에 기입한다. 이 텍스트는 정규 표현을 이용하여 원하는 라벨의 전체 또는 부분으로서 기입될 수 있다. 대안적인 실시예에 있어서, 개발자는 현재 텍스트 라벨의 원하는 이용, 텍스트 라벨의 언어 또는 원하는 라벨의 사용을 식별하는 데에 도움을 주는 기타 정보를 표시할 수 있다.

[0016] 기입된 데이터에 기초하여, 검색 엔진은 라벨 텍스트 데이터베이스를 검색하고, 제공된 네임스페이스 및 카테고리 뿐만 아니라, 원하는 텍스트에 가장 근접한 라벨 텍스트들을 식별한다. 식별된 텍스트는 검색 엔진 디스플레이에서 개발자에게 반환되고, 데이터 그리드에서 디스플레이 될 수 있다. 결과는 카테고리를 갖는 라벨 텍스트와 같이 미리 선정된 방법에 따라 순위가 매겨지며, 원하는 텍스트에 가장 근접한 언어가 먼저 디스플레이되고, 동일한 카테고리를 갖지 않는 것들은 나중에 디스플레이된다. 그 이후에 개발자는 원하는 텍스트를 포함하는 반환된 결과에서 결과를 찾을 수 있다. 만약에 결과 중의 하나가 새로운 라벨에 대한 원하는 텍스트를 포함한다면, 개발자는 결과로부터 원하는 라벨을 선택할 수 있다.

[0017] 결과를 선택할 때에, 개발자는 데이터베이스에 포함된 특정 라벨에 대한 정보를 더 제공받는다. 선택된 라벨 텍스트의 이용이 새로운 라벨과 같을 경우, 개발자는 원하는 타겟 라벨에 대하여 선택된 라벨을 이용할 수 있다. 하지만, 선택된 라벨의 이용이 새로운 라벨과 같지 않은 경우, 개발자는 그 라벨을 새로운 라벨로 복사할 수 있다.

[0018] 선택된 라벨이 새로운 라벨로 복사될 경우, GUID는 새로운 라벨에 대하여 생성되고, 그 라벨로 복사된 라벨의 GUID를 표시하는 새로운 라벨의 기록에서의 엔트리가 생성된다. 이러한 엔트리는 새로운 라벨의 텍스트가 부모 라벨의 텍스트가 변경될 때 갱신되는 것을 가능하게 하기 위해 제공된다. 또한, 라벨이 새로운 라벨에 복사된 경우, 임의의 연관된 번역은 새로운 라벨에 대한 라벨 텍스트 테이블로 복사된다. 이로 인해 비지니스 솔루션 소프트웨어 시스템은 새로운 라벨을 가용 언어로 번역하는 것과 관련된 부가적인 비용을 발생시키지 않고도 새로운 라벨로 옮겨갈 수 있다. 다른 실시예에 있어서, 라벨의 번역된 버전이 갱신될 경우, 동일한 마스터 라벨을 공유하는 모든 관련 라벨이 번역된 라벨의 새로운 버전으로 갱신된다.

[0019] 선택된 라벨이 새로운 라벨로 복사될 때, 마스터 라벨의 라벨 ID를 제외한 모든 기타 정보 뿐만 아니라 라벨의 텍스트 정보가 새로운 라벨로 옮겨간다. 만약 라벨에 대한 원하는 텍스트가 찾아지지 않는다면, 개발자는 원하는 라벨에 대한 새로운 라벨을 생성해야 한다. 또한, 개발자는 새로운 라벨에 대한 라벨 텍스트를 위하여 새로운 번역을 생성해야 한다. 다시 말하여, 만약 개발자가 시스템에서 라벨을 찾는다면, 라벨이 복사된다. 하지만, 만약 개발자가 라벨에 대한 정보를 변경하면, 라벨은 시스템에 대한 새로운 라벨로서 나타나게 될 것이다.

[0020] 도 1은 본 발명이 구현될 수 있는 적절한 컴퓨팅 시스템 환경(100)의 예를 나타낸다. 이 컴퓨팅 시스템 환경(100)은 단지 적절한 컴퓨팅 환경의 일 예이며 본 발명의 사용 또는 기능의 범위를 제한하도록 의도된 것은 아니다. 컴퓨팅 환경(100)은 예시적인 운영 환경(100)에 도시된 컴퓨트트를 중의 임의의 하나 또는 조합에 관하여 임의의 종속성(dependency) 또는 요구사항(requirement)을 갖는 것으로 해석되어서는 안된다.

[0021] 본 발명은 많은 다른 범용 또는 전용 컴퓨팅 시스템 환경들 또는 구성들과 함께 동작될 수 있다. 본 발명과 함께 사용하기에 적합할 수 있는 잘 알려진 컴퓨팅 시스템, 환경, 및/또는 구성의 예로는, 퍼스널 컴퓨터, 서버 컴퓨터, 핸드헬드(hand-held) 또는 램프 장치, 멀티프로세서 시스템, 마이크로프로세서-기반 시스템, 셋 톱 박스(set top box), 프로그램 가능한 가전제품(programmable consumer electronics), 네트워크 PC, 미니컴퓨터, 메인프레임 컴퓨터, 상기의 시스템 또는 장치 중의 임의의 것을 포함하는 분산형 컴퓨팅 환경 등이 포함될 수

있지만, 이에 한정되지 않는다.

[0022]

본 발명은 컴퓨터에 의해 실행되는, 프로그램 모듈과 같은 컴퓨터 실행가능 명령과 관련하여 일반적으로 기술될 수 있다. 일반적으로, 프로그램 모듈은 특정 태스크를 수행하거나 특정 추상 데이터 유형을 구현하는 루틴, 프로그램, 오브젝트, 컴포넌트, 데이터 구조 등을 포함한다. 본 발명은 또한 통신 네트워크 또는 다른 데이터 전송 매체를 통해 링크된 원격 프로세싱 장치에 의해 태스크를 수행하는 분산형 컴퓨팅 환경에서 실행될 수 있다. 분산형 컴퓨팅 환경에서, 프로그램 모듈 및 그와 데이터는 메모리 저장 장치를 포함하는 국부 및 원격 컴퓨터 저장 매체 내에 위치할 수 있다.

[0023]

도 1을 참조하면, 본 발명을 구현하기 위한 예시적인 시스템은 컴퓨터(110)의 형태의 범용 컴퓨팅 장치를 포함한다. 컴퓨터(110)의 컴포넌트들로는, 프로세싱 유닛(120), 시스템 메모리(130), 및 시스템 메모리를 포함하는 다양한 시스템 컴포넌트를 프로세싱 유닛(120)에 연결시키는 시스템 버스(121)가 포함될 수 있지만, 이에 한정되는 것은 아니다. 시스템 버스(121)는 다양한 버스 아키텍처 중의 임의의 것을 사용하는 로컬 버스, 주변 버스, 및 메모리 버스 또는 메모리 컨트롤러를 포함하는 몇 가지 유형의 버스 구조 중의 임의의 것일 수 있다. 예로서, 이러한 아키텍처는 산업 표준 아키텍처(ISA) 버스, 마이크로 채널 아키텍처(MCA) 버스, 인핸스드 ISA(Enhanced ISA; EISA) 버스, 비디오 일렉트로닉스 표준 어소시에이션(VESA) 로컬 버스, 및 [메자닌(Mezzanine) 버스로도 알려진] 주변 컴포넌트 상호접속(PCI) 버스를 포함하지만, 이에 한정되는 것은 아니다.

[0024]

컴퓨터(110)는 통상적으로 다양한 컴퓨터 관독가능 매체를 포함한다. 컴퓨터 관독가능 매체는 컴퓨터(110)에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수 있으며, 휴발성 및 비휘발성 매체, 분리형(removable) 및 비분리형(non-removable) 매체를 모두 포함한다. 예로서, 컴퓨터 관독가능 매체는 컴퓨터 저장 매체 및 통신 매체를 포함할 수 있지만, 이에 한정되는 것은 아니다. 컴퓨터 저장 매체는 컴퓨터 관독가능 명령, 데이터 구조, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현되는 휴발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다. 컴퓨터 저장 매체는 RAM, ROM, EEPROM, 플래쉬 메모리 또는 기타 메모리 기술, CD-ROM, DVD(digital versatile disk) 또는 기타 광학 디스크 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 기타 자기 저장 장치, 또는 컴퓨터(110)에 의해 액세스될 수 있고 원하는 정보를 저장하는 데 사용될 수 있는 임의의 기타 매체를 포함할 수 있지만, 이에 한정되지 않는다. 통신 매체는 통상적으로 반송파 또는 기타 전송 메카니즘 등의 변조된 데이터 신호에 컴퓨터 관독가능 명령, 데이터 구조, 프로그램 모듈, 또는 다른 데이터를 구현하며, 임의의 정보 전달 매체를 포함한다. "변조된 데이터 신호"라는 용어는 신호 내에 정보를 인코딩하도록 설정되거나 변환된 특성을 하나 또는 그 이상을 갖는 신호를 의미한다. 예로서, 통신 매체는 유선 네트워크 또는 직접 유선 접속 등의 유선 매체와, 음향, RF, 적외선 및 기타 무선 매체 등의 무선 매체를 포함하지만, 이에 한정되지 않는다. 상술한 것들 중의 임의의 조합이 컴퓨터 관독가능 매체의 범위 내에 포함되어야 한다.

[0025]

시스템 메모리(130)는 ROM(131) 및 RAM(132)과 같은 휴발성 및/또는 비휘발성 메모리 형태의 컴퓨터 저장 매체를 포함한다. 시동 중과 같은 때에 컴퓨터(110) 내의 구성요소들간의 정보 전달을 돋는 기본 루틴을 포함하는 기본 입출력 시스템(BIOS; 133)은 전형적으로 ROM(131)에 저장된다. RAM(132)은 전형적으로 처리 장치(120)에 의하여 즉시 접속 가능하거나 곧 작동될 수 있는 데이터 및/또는 프로그램 모듈을 포함한다. 예로서, 도 1은 운영 시스템(134), 응용 프로그램(135), 다른 프로그램 모듈(136) 및 프로그램 데이터(137)를 도시하지만, 본 발명이 이들에 의하여 제한되는 것은 아니다.

[0026]

컴퓨터(110)는 다른 분리형/비분리형 휴발성/비휘발성 컴퓨터 저장 매체를 포함할 수 있다. 단지 예로서, 도 1은 분리형이고 비휘발성인 자기 매체에 기록 및 관독하는 하드 디스크 드라이브(141), 분리형이고 비휘발성인 자기 디스크(152)에 기록 및 관독하는 자기 디스크 드라이브(151), 및 CD ROM이나 다른 광 매체와 같이 분리형이고 비휘발성인 광 디스크(156)에 기록 및 관독하는 광 디스크 드라이브(155)를 도시한다. 본 예시적인 운영 환경에서 사용될 수 있는 다른 분리형/비분리형 휴발성/비휘발성 컴퓨터 저장 매체는 자기 테이프 카세트, 플래시 메모리 카드, 디지털 다목적 디스크(digital versatile disk), 디지털 비디오 테이프, 고체 상태 RAM(solid state RAM), 고체 상태 ROM(solid state ROM) 등을 포함하지만, 이들에 의하여 제한되지 않는다. 하드 디스크 드라이브(141)는 전형적으로 인터페이스(140)와 같은 비분리형 메모리 인터페이스를 통하여 시스템 버스(121)에 접속되고, 자기 디스크 드라이브(151) 및 광 디스크 드라이브(155)는 전형적으로 인터페이스(150)와 같은 분리형 메모리 인터페이스에 의하여 시스템 버스(121)에 접속된다.

[0027]

상술되고 도 1에서 도시된 드라이브 및 그와 연관된 컴퓨터 저장 매체는 컴퓨터 관독 가능 명령, 데이터 구조, 프로그램 모듈 및 컴퓨터(110)를 위한 다른 데이터의 저장을 제공한다. 도 1에서, 예를 들어, 하드 디스크 드

라이브(141)는 운영 시스템(144), 응용 프로그램(145), 다른 프로그램 모듈(146) 및 프로그램 데이터(147)를 저장하는 것으로 도시된다. 이러한 구성 요소들이 운영 시스템(134), 응용 프로그램(135), 다른 프로그램 모듈(136) 및 프로그램 데이터(137)와 같거나 다를 수 있다는 점을 주목해야 한다. 운영 시스템(144), 응용 프로그램(145), 다른 프로그램 모듈(146) 및 프로그램 데이터(147)는, 최소한 이들이 서로 다른 카피라는 것을 도시하기 위하여 본 명세서에서 서로 다른 참조 부호를 부여 받는다.

[0028] 사용자는 키보드(162), 마이크로폰(163) 및 마우스, 트랙볼 또는 터치 패드와 같은 포인팅 장치(161)와 같은 입력 장치를 통하여 컴퓨터(110)에 명령 및 정보를 입력한다. 도시되지 않는 다른 입력 장치는 조이스틱, 게임 패드, 인공 위성 수신기(satellite dish), 스캐너 등을 포함할 수 있다. 이러한 입력 장치 및 다른 입력 장치는 종종 시스템 버스와 결합된 사용자 입력 인터페이스(160)를 통하여 처리 장치(120)에 접속되지만, 병렬 포트, 게임 포트 또는 USB(universal serial bus)와 같은 다른 인터페이스 및 버스 구조에 의하여 접속될 수도 있다. 또한, 모니터(191) 또는 다른 유형의 디스플레이 장치가 비디오 인터페이스(190)와 같은 인터페이스를 통하여 시스템 버스(121)에 접속될 수 있다. 모니터뿐만 아니라, 컴퓨터는 주변 출력 인터페이스(195)를 통하여 연결될 수 있는 스피커(197) 및 프린터(196)와 같은 다른 주변 출력 장치를 포함할 수 있다.

[0029] 컴퓨터(110)는 원격 컴퓨터(180)와 같은 하나 이상의 원격 컴퓨터와의 논리적 접속을 이용하는 네트워크 환경에서 동작할 수 있다. 원격 컴퓨터(180)는 개인용 컴퓨터, 핸드-헬드 장치, 서버, 라우터, 네트워크 PC, 피어 장치(peer device) 또는 다른 일반적인 네트워크 노드일 수 있고, 전형적으로 컴퓨터(110)와 관련하여 상술된 구성 요소들의 다수 또는 모두를 포함한다. 도 1에서 도시된 논리적 접속은 LAN(local area network; 171)과 WAN(wide area network; 173)을 포함하지만, 다른 네트워크를 포함할 수도 있다. 이러한 네트워크 환경은 사무실, 기업 컴퓨터 네트워크, 인트라넷 및 인터넷에서 일반적이다.

[0030] LAN 네트워크 환경에서 사용될 때, 컴퓨터(110)는 네트워크 인터페이스 또는 어댑터(170)를 통하여 LAN(171)에 접속된다. WAN 네트워크 환경에서 사용될 때, 컴퓨터(110)는 전형적으로 모뎀(172) 또는 인터넷과 같이 WAN(173)에 걸쳐 통신을 설정하기 위한 다른 수단을 포함한다. 내장형 또는 외장형일 수 있는 모뎀(172)은 사용자 입력 인터페이스(160) 또는 다른 적절한 메커니즘을 통하여 시스템 버스(121)에 접속될 수 있다. 네트워크 환경에서, 컴퓨터(110)와 관련하여 도시된 프로그램 모듈 또는 그 일부는, 원격 메모리 저장 장치에 저장될 수 있다. 예로서, 도 1은 원격 응용 프로그램(185)이 원격 컴퓨터(180) 상에 상주하는 것으로 도시하지만, 본 발명이 이에 의하여 제한되는 것은 아니다. 도시된 네트워크 연결은 예시적이며 컴퓨터간의 통신 링크를 설정하기 위한 다른 수단이 사용될 수 있다는 점을 이해할 수 있을 것이다.

[0031] 일 실시예에서, 시스템(100)은 특정 애플리케이션 모듈(specific application modules)을 갖는 일반적인 회계 및 비즈니스 기능(general accounting and business functions)을 통합할 수 있는 비즈니스 관리 솔루션(business management solution)을 생성하기 위한 소프트웨어를 포함한다. 이 모듈은, 재정(finance), 무역(trade), 병참(logistics), 생산(production), 고객 서비스(customer service), 프로젝트(projects) 및 인간 리소스(human resources)를 위한 모듈들을 포함할 수 있다. 그러나, 다른 모듈 유형이 비즈니스 관리 솔루션에서 사용될 수 있다. 본 발명의 솔루션은 다수의 통화(currencies){유로, 달러, 엔, 원(Wan) 등}, 다수의 언어(영어, 독일어, 프랑스어, 덴마크어, 러시아어, 일본어, 중국어 등) 및 다수의 세금 포맷(tax formats){다수의 조세 당국과 거래하는 엔드 사용자(end users)에 의하여 사용하기 위함임}을 지원하도록 구성될 수 있다.

[0032] 비즈니스 관리 솔루션에서 사용되는 모듈은 다양한 서로 다른 소스로부터 개발될 수 있다. 일 실시예에서, 이들 모듈 모두 간의 공통 고리(common link)는 라벨(labels)의 사용이다. 라벨은 식별자(identifier)에 의하여 표시되는 텍스트이다. 라벨은, 대화 상자(dialog boxes), 텍스트 문자열(text strings), 또는 사용자에게 정보를 전달하기 위하여 사용되는 임의의 텍스트를 포함하는 라벨 속성(label property)을 갖는 임의의 객체(object)에 대하여 사용될 수 있다. 라벨은 그래픽 사용자 인터페이스(graphical user interface; GUI)와 같은 리소스를 통하여 보통 사용자에게 제시된다. 그러나, 라벨은 사용자에게 텍스트를 제시하는 임의의 다른 수단을 통하여 사용자에게 제시될 수 있다. 더욱이, 라벨은 "label", "help", "caption" 및 "tool tip"과 같은 라벨 속성을 갖는 제어에 대하여 사용될 수 있다. 이전의 비즈니스 솔루션 소프트웨어에서, 각각의 모듈에 대한 라벨은 별도의 리소스 파일에 보존된다. 그러나, 이들 리소스 파일은 특정 모듈에 독점인 평면 파일 아키텍처(flat file architecture)에 보통 보존되었다. 흔히 모듈의 개발 프로세스 동안, 개발자는 현재 생성되고 있는 라벨의 요구 조건을 만족시키는 다른 모듈의 라벨을 알고 있다. 그러나, 종래 기술의 시스템의 전용 파일 소스 구조(proprietary file source structure)에 의하면, 사용자가 다른 모듈의 하나의 모듈로부터 라벨을 사용하는 것이 방해되고, 라벨을 검색하는 것이 불가능하다.

- [0033] 도 2는 본 발명의 일 실시예에 따라, 비즈니스 솔루션 시스템(200)에 라벨을 포함하는 다양한 모음들(collections) 간의 관계(relationship)를 설명하는 엔티티 관련 다이어그램(entity relation diagram)이다. 도 2의 엔티티 관련 다이어그램은 언어 테이블(210)(또는 시스템에서 이용 가능한 언어의 다른 지시), 마스터(master; 220), 라벨 ID 테이블(230), 및 라벨 텍스트 테이블(240)을 포함한다. 일 실시예에서, 이들 모음은 테이블에 정렬된 SQL 메타데이터 저장소(metadata stores)로서 구성된다. 그러나, 모음을 위한 다른 데이터베이스 및 다른 정렬이 사용될 수 있다. 라벨 ID 테이블(230) 및 라벨 텍스트 테이블(240)은 도 3a 및 도 3b를 참조하여 보다 상세하게 설명된다.
- [0034] 언어 테이블(210)은 적어도 두 개의 서브필드(sub-fields)를 포함하는 테이블이다. 언어 테이블(210)의 두 개의 서브필드는 언어 ID 필드(211) 및 언어 이름 필드(212)이다. 언어 ID 필드(211)는 특정 언어를 지시하는 코드를 보존하고, 비즈니스 솔루션 소프트웨어 프로그램에 의하여 이해 가능하다. 언어 이름 필드(212)는 언어의 이름을 보존하는 텍스트 필드이다. 예를 들면, 이용 가능한 언어의 하나가 미국 영어(English-United States)라면, 실례적으로 언어 ID 필드(211)는 "01"일 수 있고, 또는 ASCII 표준을 사용한다면 "en-us"일 수 있다. 그러나, 다른 ID 유형이 언어 ID 필드(211)에서 사용될 수 있다. 예를 들면, 이 엔트리에 대한 메타테이블의 언어 이름 필드(212)는 "영어-미국(English-US)"일 것이다. 선택적으로, 이 솔루션이 특정 솔루션에 의하여 제공된다면, 특정 언어 텍스트를 갖는 라벨일 수 있다.
- [0035] 언어 테이블(210)에 저장된 정보는, 사용자가 시스템의 동작 언어를 보거나 변화시키기를 바라는 경우에, 라벨 대화 디스플레이로 사용자에게 디스플레이될 수 있다. 언어 테이블(210)은 라벨 텍스트 테이블(240)을 갖는 1:n 관계이다. 이 관계(1:n)는, 시스템(200)에 이용 가능한 각각의 언어로 상이한 라벨을 나타내는 복수의 라벨 텍스트가 있을 수 있기 때문에, 발생할 수 있다.
- [0036] 일 실시예에서, 마스터(220)는 시스템(200)의 각각의 라벨의 하나의 버전을 포함하는 테이블이다. 일 실시예에서, 마스터(220)는 원래의 언어 및 포맷으로 라벨의 원래 버전을 보존한다. 그러나, 라벨의 다른 버전은 마스터(220)에 저장될 수 있다. 예를 들면, 최적 실시 지침(best practices guidelines)이 사용되면, 시스템(200)은 미국 영어로 라벨을 마스터(220)에 저장할 수 있다. 최적 실시 지침은, 특정 카테고리 유형, 설명(descriptions) 및 언어로 라벨 개발을 표준화하는 절차들(procedures)의 세트이다. 시스템(200)의 각각의 라벨은 마스터(220)의 연관 엔트리(associated entry)이다. 라벨의 번역(label's translation)이 개신되고 있을 때, 라벨이 현재 선택된(활성) 언어로 번역되지 않았거나, 임의의 이유로 시스템(200)에 액세스 가능하지 않을 때, 마스터 라벨이 사용된다. 그러나, 선택적인 실시예에서, 마스터(220)에 저장된 정보는, 마스터 라벨의 라벨 ID를 지시하는 언어(240) 또는 라벨 테이블(230)에 필드(들)로서 저장될 수 있다. 더욱이, 마스터(220)는, 언어 라벨이 생성되었다는 정보를 보존하는 단순한 필드일 수 있다. 이 필드는 시스템(200)의 임의의 테이블에 위치될 수 있다.
- [0037] 각 마스터 테이블 엔트리(master table entry, 221)는 라벨 ID 테이블(230)의 하나의 엔트리에 대응한다. 라벨 ID 테이블(230)은 라벨을 정확하게 번역하기 위하여 번역가(텍스트를 번역하는 자)를 보조하는 다양한 특성을 갖는다. 이러한 특성은 라벨을 적절히 사용하기 위하여 개발자를 보조하기도 한다. 마스터 테이블 엔트리(221)와 라벨 ID 테이블(230)의 관계는 1:n이며, 복수의 라벨 ID들이 동일한 마스터 라벨을 공유할 수 있다.
- [0038] 라벨 텍스트 테이블(240)은 라벨 ID에 의하여 식별되는 각 라벨에 대한 텍스트를 포함하는 엔트리를 포함한다. 라벨 텍스트 테이블(240)은 각 라벨 ID에 대한 다양한 언어의 번역을 포함하는 엔트리도 포함한다. 라벨 ID 테이블(230)과 라벨 텍스트 테이블(240)의 관계나 언어 테이블(210)과 라벨 텍스트 테이블(230)의 관계는 1:n이다. 이는 하나의 라벨에 대하여 라벨 ID 테이블(230) 내에 하나의 엔트리가 있기 때문이다. 그러나 라벨의 텍스트는 복수의 언어로 존재할 수 있다. 그러나, 본 발명의 다른 실시예에 있어서, 라벨 ID와 라벨 텍스트의 관계는 1:1(또는 텍스트가 존재하지 않는 경우 1:0)일 수 있다. 이때 마스터 라벨의 각각의 번역은 라벨 ID 테이블(230)에 고유의 라벨 ID 및 엔트리를 갖는다.
- [0039] 도 3a는 본 발명의 일 실시예에 따른 라벨 ID 테이블(230)을 포함하는 필드를 도시한다. 라벨 ID 테이블(230)은 ID 필드(231), 네임 스페이스 필드(232), 카테고리 필드(234) 및 설명 필드(235)를 포함한다. 다른 실시예에 있어서, 라벨 ID 테이블(230)은, 라벨 ID 테이블 내의 엔트리가 다른 라벨과 중복되는지 또는 그 테이블 내의 다른 엔트리와 중복되는지 여부를 나타내는 필드(236)를 포함할 수 있다.
- [0040] 전형적으로, 라벨은 리소스 파일에 보유되어 있다. 현재의 비즈니스 솔루션 시스템은 SQL 테이블과 같은 데이터베이스 메타데이터 저장소를 통하여 이용 가능한 일반적인 리소스 파일을 사용하지 않는다. 현재의 비즈니스 솔루션 시스템에 있어서, 이러한 라벨들은 전용 리소스 파일(proprietary resource file)에 저장되어 있다. 전

용 리소스 파일을 사용하는 것과 관련된 한 가지 문제점은, 개발자가 시스템의 라벨 또는 라벨의 일부를 파일에 대한 모든 리소스를 구비한 파일로 대체하고자 하는 경우, 시스템은 현재의 라벨과 동일한 라벨 속성 또는 터미널러지(terminology)를 갖는 시스템 내의 다른 파일을 찾지 않는다는 점이다. 그러나, 비즈니스 솔루션 환경에 있어서, 비즈니스 솔루션 시스템은 복수의 벤더들(vendors)에 의하여 개발된 동일한 또는 유사한 문제에 대하여 다수의 서로 다른 솔루션을 취급할 것이 요구된다. 따라서, 본 발명의 라벨 시스템(200)은 다양한 벤더들에 의하여 제작된 모든 공헌의 총체를 사용할 수 있다.

[0041] ID 필드(231)는 시스템(200) 내의 특정 라벨을 식별하기 위하여 사용된다. 일 실시예에 있어서, 이 ID는 전역 고유 식별자(GUID)이다. GUID는 두 명의 벤더가 두 개의 서로 관련없는 라벨에 대하여 동일한 식별 번호를 선택함으로써 발생하는 문제를 회피하기 위하여 사용된다. 일 실시예에 있어서 GUID는 고유 식별자가 요구되는 모든 컴퓨터 및 네트워크에 걸쳐 사용될 수 있는 128 비트 정수(16바이트)이다. 이러한 식별자 시스템을 사용하면 두 개의 라벨이 동일한 ID를 가질 가능성을 감소시킨다. 일 실시예에 있어서 GUID는 스트링(string)으로서 표현되고, 다음 패턴에 따라 포맷된다.

[0042] XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX

[0043] 여기서, GUID의 값은 8, 4, 4, 4, 12개의 디지트(digits) 그룹으로 이루어지고 하이픈으로 구분되는 일련의 소문자의 16진 디지트로서 표현된다. 예를 들어 라인 301의 엔트리에 대한 GUID 반환값은 382c74c3-721d-4f34-80e5-57657b6cdc27일 수 있다. 그러나, ID 필드(231)에 대하여 다른 포맷 유형이 사용될 수도 있다. 새로운 라벨이 생성될 때마다 새로운 GUID가 할당된다. 일 실시예에 있어서, 새로운 라벨은 단순히 현존하는 라벨의 번역이 아닌 시스템(200)에서 새로운 라벨로서 정의된다. 그러나, 다른 실시예에 있어서는, ID 필드(231) 내의 새로운 엔트리는 현존하는 라벨의 번역을 위하여 생성된다.

[0044] GUID를 사용하면 현재의 시스템에 비하여 부가적인 장점을 제공한다. 첫째로, 각 라벨을 복수의 언어로 분할할 필요가 없다. 또한, GUID는 열 번호의 사용 등의 다른 방법으로 고유성(uniqueness)을 통제할 필요 없이 고유하게 각 라벨을 식별할 수 있게 한다. 둘째로, GUID는 라벨의 물리적 스토리지(storage)가 웹 서비스에 있어서 전용 리소스 파일에서 메타데이터 데이터베이스와 같이 공통 리소스 파일(common resource files)로 변경될 수 있게 한다.

[0045] 라벨 ID 테이블(230; label ID table)에서의 네임 스페이스 필드(232)는, 라벨 텍스트를 마스터 언어에서 타겟 언어(target language)로 번역할 때, 번역자가 라벨에 대한 바른 용어를 획득하는 것을 돋기 위하여 라벨 ID로 들어간 특수 필드이다. 현재 라벨 시스템에서, 특정 라벨이 프로그램에서 어디에 사용되고 있는지 쉽게 파악하는 것은 불가능하다. 또한, 어느 영역에서 라벨이 사용되는지 파악하는 것도 쉽지 않다. 따라서, 프로그램이 컴퓨터에 설치되고, 라벨이 어디서 사용되는지 파악할 수 없다면, 번역자가 라벨에 대한 바른 용어를 획득하는 것은, 불가능하지는 않지만 극히 어렵다. 네임 스페이스 필드(232)는 프로그램이 번역자의 컴퓨터에 설치되도록 하지 않고도 라벨이 사용되는 영역을 파악할 수 있도록 해 준다. 네임 스페이스 필드에 포함된 정보는 새로운 라벨이 생성될 때(자동적 혹은 수동적으로) 개발자에 의하여 제공되고, 특정 라벨의 사용과 관련된 정보를 제공한다. 예를 들어, 도 3a의 라인(302)의 엔트리에서, 네임 스페이스 필드(232)는 번역자 또는 개발자에게 이 라벨이 빌링 모듈에서 사용되고 있다고 알려준다.

[0046] 라벨 ID 테이블(230)에서의 카테고리 필드(234)는 라벨이 프로그램의 어느 컴포넌트에 사용되었는지를 나타내는 특수 필드이다. 또한, 바른 방식으로 라벨에 기록하기 위하여 라벨 텍스트를 생성하는 동안, 카테고리 필드(234)를 사용하여 개발자는 도움을 받게 된다. 카테고리는 라벨에 대한 노드 타입 및 속성(property)의 조합이다. 노드 타입은 라벨의 특정한 속성이다. 이러한 라벨 속성 중 일부는 "label", "help", "caption" 등을 포함할 수 있다. 따라서, 카테고리 필드(234)는 원하는 프로그램에 대하여 라벨이 적절한 방식으로 사용되는 것을 보증할 수 있도록 한다. 또한, 카테고리는 시스템에서 존재하는 모든 제어의 맵핑이다. 따라서, 원하는 라벨의 카테고리 또는 타입에 기초한 라벨의 현존 데이터베이스를 검색하는 것이 가능하다. 본 발명에서는, 카테고리는 시스템(200)에 의하여 수행되는 각각의 규칙 또는 제어에 대하여 생성된다. 모듈 추가 카테고리(module additional category)를 개발하는 동안, 만약 특수 규칙이 특정 라벨 텍스트에 대하여 필요한 경우, 카테고리가 생성될 수 있다. 예를 들어, 노드 필드(234)는, 사용자를 모듈에서의 다른 점으로 지향시키기 위하여, 모듈의 메뉴 바 상에서 엔트리(303)에서의 라벨이 사용된다고 가리키는 엔트리일 수 있다. 각각의 카테고리는 시스템(200)에서의 특정 함수로 맵핑된다.

[0047] 설명 필드(235)는 사용자 또는 개발자에게 라벨이 어떻게 사용되는지 설명하는 테이블(230)에서의 엔트리이다. 예를 들어, 설명 필드(235)는 엔트리(303)에서의 라벨이 대장(ledger)에 사용된다는 것을 가리키는 엔트리일 수

있다. 이 설명은 단어들(평문) 속에 존재하거나 미리 결정된 코드의 리스트에 대하여 코딩될 수 있다.

[0048] 필드(236)로부터의 복사는 엔트리의 관련 라벨이 라벨 시스템에서의 다른 라벨로부터 복사되었는지 여부를 지시한다. 만약 라벨이 다른 라벨로부터 복사되었다면, 필드(236)로부터의 복사는 부모(parent) 또는 마스터 라벨(220)으로부터의 ID를 포함한다. 일 실시예에서 상기 ID는 부모의 GUID이다. 그러나, 다른 ID들도 필드(236)에서 사용될 수 있다. 만약 라벨이 다른 라벨의 복사된 버전이 아니라면, 필드(236)로부터의 복사는 비어 있거나 0으로 설정된다. 또한, 만약 복사후에 라벨의 텍스트가 마스터 라벨의 텍스트로부터 변화한다면, 필드(236)로부터의 복사는 0으로 설정되고, 그에 의하여 마스터 레벨(220) 및 상기 특정 자식 라벨 사이의 임의의 링크는 삭제된다. 그러나, 엔트리에서의 다른 변화는 필드(236)로부터의 복사가 재설정되도록 할 것이다. 선택적 실시예에서, 도 3a의 라벨 ID 테이블(230)은 라벨 ID들이 복사된 라벨들을 포함한다는 것을 지시하는 필드들을 포함한다. ID 테이블(230)에서의 라벨은 마스터 라벨 또는 자식이다.

[0049] 도 3b는 특정 라벨의 라벨 텍스트 테이블(240)을 차지하는 필드를 나타낸다. 라벨 텍스트 테이블(240) 엔트리는 적어도 세가지 다른 필드{라벨 ID 필드(244), 라벨 텍스트 필드(241) 및 편집된 날짜(242)}를 포함한다. 본 발명의 선택적 실시예에서, 추가적 필드는 라벨 텍스트 테이블(240)로 추가될 수 있다. 이 필드는 텍스트가 작성된 언어 또는 라벨 텍스트 테이블에서 텍스트의 각각의 버전의 엔트리 인식자를 인식하는 필드를 지시하는 필드(243)를 포함한다.

[0050] 라벨 텍스트 테이블(240)의 텍스트 필드(241)는 시스템에 대한 모든 가능한 언어에 있어서 라벨용 텍스트의 가장 최근 버전을 포함한다. 라벨에 대한 라벨 텍스트 테이블(240)의 제1 엔트리(351)는 라벨이 작성된 원시 또는 마스터 언어에 존재한다. 이 텍스트는 마스터 텍스트로 호칭된다. 만약 라벨이 최선의 실무 지침에 따라 개발된다면, 마스터 텍스트는 영-미어로 작성될 것이다. 그러나, 다른 언어가 마스터 언어로 사용될 수 있어서, 최선의 실무 지침을 따를 필요는 없다. 일반적으로 마스터 언어는 시스템이 현재 동작하는 현재 언어에 대응한다.

[0051] 라벨 텍스트가 마스터 언어에서 다른 언어로 번역됨에 따라, 새로운 라벨 텍스트 엔트리가 라벨 텍스트 테이블(예컨대 322 및 323)에 생성된다. 이들 엔트리는 자신들의 각각의 언어로 원래의 라벨 텍스트의 번역된 버전을 포함한다. 예를 들어, 만약 라벨의 원래의 텍스트가 "Do you want to save the changes you made to" 이라면, 이 텍스트는 라벨 텍스트 테이블(240)의 첫 번째 라인(321)에 저장된다. 그 후에, 이 텍스트가 독일어, 텐마크어 및 프랑스어로 번역되는 때에, 이들 번역은 라벨 텍스트 테이블에서 마스터 텍스트 엔트리 아래의 엔트리들에 기입된다. 이들 추가된 번역은 엔트리(352, 353 및 354)에 의해 지시된다.

[0052] 개발자가, 언제 라벨의 텍스트가 라벨 텍스트 테이블에서의 특정 엔트리와 연관된 언어로 최후에 변경되었는지를 알 수 있도록, 편집 데이터 필드(edited data field; 242)가 라벨 텍스트 테이블(240)에 제공된다. 또한, "변경자(modified by)"와 같은 유효성 정보를 포함하는 다른 필드가 추가될 수 있다. 이러한 데이터는, 텍스트에 대한 번역이 제공되는 경우에, 그 번역이 이미 존재하는 보다 최신 버전의 번역을 대체하지 않는다는 것을 보증하는데 도움이 된다. 번역이 자동적으로 시스템에 로딩되는 때에, 번역 날짜를 비교하여, 만약 엔트리에 있는 버전이 제시된 엔트리보다 최근의 것이라면, 그 제시된 엔트리는 테이블에 기입되지 않는다. 또한 편집 날짜 필드(edited data field; 242)는 개발자로 하여금 마스터 텍스트 엔트리의 가장 최신의 버전으로 번역이 현재의 것인지를 검사할 수 있도록 한다. 일 실시예에 있어서, 마스터 텍스트 엔트리의 편집 날짜(321)는 바람직하게는 특정 라벨에 대한 라벨 텍스트 테이블(240)에서의 가장 오래된 날짜이다.

[0053] 일부 실시예에서, 라벨 텍스트 테이블(240)은 언어 필드 엔트리(language field entry; 243)를 포함한다. 언어 필드 엔트리(243)는, 개발자가 그 언어를 말하거나 이해하지 않아도, 개발자에게 라벨 텍스트 테이블에서의 특정 엔트리의 특정 언어를 지시한다. 언어 필드 엔트리(243)에서의 이러한 언어 지시자는 언어에 대한 수치적 표현일 수 있거나, 언어의 이름이나 임의의 다른 종류의 언어 식별자로 작성될 수 있다.

[0054] 만약 언어를 식별하기 위해 수치적 표현이 사용된다면, 그 언어에 대한 참조 번호나 엔트리는 예시적으로 ACSII 언어 코드나 ISO 639와 같은 공지된 표준을 따를 수 있다. 그러나, 텍스트 엔트리의 언어를 식별하기 위해 다른 코드가 사용될 수 있다. 만약 언어의 이름이 필드(243)에 기입된다면, 그 언어의 이름은 예시적으로 공지된 표준(예컨대, 영어식 언어 이름)에 따라 작성될 수 있다. 그러나, 다른 형식이 사용될 수도 있다.

[0055] 라벨 텍스트 테이블(240)은 또한 특정 라벨 텍스트에 대한 라벨 ID(231)를 식별하는 엔트리 필드(244)를 포함한다. 이 엔트리는 개발자로 하여금 현재의 텍스트가 어느 라벨 ID(231)에 관련되는지를 알 수 있도록 한다. 라벨 텍스트 테이블(240)의 다른 실시예들은 텍스트 ID(245)에 관한 엔트리를 포함한다. 이 텍스트 ID(245)는 각

각의 텍스트 라벨을 라벨 텍스트 테이블(240)에서의 고유의 엔트리로서 개별적으로 식별하기 위해 제공된다. 텍스트 ID(245) 엔트리는 GUID가 될 수 있으며, 또한 라벨 텍스트 테이블의 구성에 부합하는 임의의 다른 식별 방법이 될 수 있다. 대체 실시예에 있어서, 라벨 텍스트 테이블(240) 및 라벨 ID 테이블(230)은 하나의 테이블이나 데이터베이스로 병합될 수 있다. 이 실시예에서는, 라벨들을 관리하고, 모듈이 실행되는 경우에 올바른 라벨 언어 텍스트가 표시된다는 것을 보증하기 위해서, 추가적인 필드들의 집합이 필요할 것이다. 또한, 라벨의 로딩을 관리하기 위해 인덱스가 추가될 것이다.

[0056] 도 4는 라벨 시스템 인터페이스(400)를 포함하는 컴포넌트를 나타내는 블록도이다. 라벨 시스템 인터페이스(400)는 개발자(401)를 메타데이터 저장소(409)에 연결시키는 인터페이스이며, 개발자로 하여금 시스템(200)에 대한 새로운 모듈을 개발하는 경우에 현존하는 라벨을 조작할 수 있도록 한다. 라벨 시스템 인터페이스(400)는 라벨 대화(402), 라벨 대화 로직 컴포넌트(403), 확장 언어 인터페이스(404), 및 라벨 인터페이스(405)를 포함한다.

[0057] 개발자는 라벨 대화(402)를 통해 라벨 시스템 인터페이스(400)와 상호작용한다. 라벨 대화(402)는 라벨 시스템 인터페이스의 특징들에 대하여 개발자가 액세스할 수 있도록 하는 사용자 인터페이스이다. 일 실시예에서, 라벨 대화(402)는 사용자로 하여금 특정 라벨의 사용을 보고(view) 관리할 수 있게 하는 원도우이다. 라벨 대화(402)는, 라벨들을 취급하고 모든 가용 언어들로 된 라벨이 이용할 수 있는 명령들의 전체 세트(full set)에 액세스하기 위해, 메타데이터 저장소(metadata store)로의 그러한 인터페이스를 사용한다. 이러한 특징들을 제공하기 위해, 라벨 대화 인터페이스(402)는 라벨 시스템(400) 내의 각각의 라벨에 대한 모든 가용 언어들에 액세스할 필요가 있다.

[0058] 한편, 라벨 대화(402)는 라벨이 특정 상황(specific situation)에서 사용될 수 있는지 여부를 결정하는 어떤 로직(logic)도 포함하지 않는다. 모든 제어들은 라벨 인터페이스 또는 라벨 대화 로직 컴포넌트(403)를 제어하는 클래스로 전달된다. 라벨 대화 로직 컴포넌트(403)는 라벨 대화(402)와 함께 사용되도록 설계되나, 그것이 라벨 대화(402)에 대한 모든 로직을 포함하기 때문에 라벨 대화 로직의 소정의 특징들은 다른 영역들에서 재사용될 수 있다. 나아가, 라벨 대화 로직 컴포넌트(403)는 각각의 라벨이 그 라벨에 대한 카테고리와 네임스페이스의 올바른 조합(combination)과 일관된 방식으로 사용되는 것을 보장한다. 이러한 로직은 먼저 새로운 마스터 라벨을 생성하지 않고 상이한 영역 내의 라벨을 부주의하게 사용하는 것을 방지한다.

[0059] 확장된 언어 인터페이스(404)는 시스템 내에서 가용한 모든 언어들에의 라벨 대화 인터페이스(400) 액세스를 제공한다. 확장된 언어 인터페이스(404)는 복수의 언어들을 라벨 대화 로직 컴포넌트(403) 및 라벨 대화(402)에 가용하게 만들기 위해 다른 클래스를 사용한다. 확장된 언어 인터페이스(404)는 시스템 내의 하나 이상의 언어에 공통된 메소드들(methods)만을 갖는다(모든 가용 언어들의 목록을 반환(return)하며, 또한 현재의 언어에 대한 ID를 반환함). 하나의 언어에만 관련된 메소드들은 라벨 클래스(405) 상에 직접적으로 위치된다. 라벨 클래스는 메타데이터 내의 라벨들에 접속할 수 있게 한다. 각각의 가용 언어에는 라벨 클래스의 하나의 인스턴스(instance)가 존재한다. 그러나, 어떤 경우에는 특정한 클래스에 대하여 인스턴스가 존재하지 않을 수도 있다.

[0060] 언어 인터페이스 컴포넌트(404)는 또한 가용 언어들에의 액세스를 제어한다. 언어 인터페이스 컴포넌트(404)는 라벨의 인스턴스들의 맵(map)을 포함한다. 또한, 언어 인터페이스 컴포넌트(404)에는 현재 라벨에 대한 각각의 가용 추가 언어 클래스(available extra language class)로의 인터페이스가 있다.

[0061] 도 5는 본 발명의 일 실시예에 따라 모듈의 일부의 개발 중에 개발자가 모듈에 대한 새로운 라벨을 생성할 때 수행되는 단계들을 나타내는 흐름도이다. 예를 들어, 개발자가 모듈 내에 "고객(customer)"에 대한 입력을 나타내는 라벨을 만들고자 하는 경우에, 개발자는 그 라벨이 모듈 내에서 어떻게 사용될지를 결정해야 한다. 용어 "고객"은 많은 상이한 의미들을 갖는다. 예를 들어, 고객은 당신으로부터 물건들 및 서비스들을 사는 사람을 의미할 수도 있고, 당신이 거래를 하는 "누구(who)"를 의미할 수도 있다. 이러한 단어의 사용은 다른 언어들로 된 텍스트(text) 상에 영향을 끼칠 것이다. 이는 도 5의 블록(501)에서 도시된다.

[0062] 그 다음에 개발자는 라벨 대화 프로그램을 연다. 열린 라벨 대화는 도 6에서 도시된 라벨 대화(600)와 예시적으로 유사하다. 한편, 다른 인터페이스들이 사용될 수도 있다. 제시된 인터페이스는 개발자로 하여금 그가 라벨에 대하여 원하는 특정 텍스트를 입력할 수 있도록 한다. 라벨 대화의 개방(opening)은 도 5의 블록(502)에서 도시된다.

[0063] 라벨 대화가 열리면, 개발자는 라인(602)에 라벨을 위한 희망 텍스트를 입력한다. 대안으로, 개발자는 라벨을 위한 희망 텍스트의 일부를 라인(602)에 입력할 수 있다. 예컨대, 개발자는 텍스트 섹션에 "customer"를 입력

하거나, "cust"를 입력할 수 있다. 또한, 개발자는 새로운 라벨이 라인(603 및 604)에서 사용되는 방법을 나타내는 데이터를 인터페이스에 입력한다. 이러한 데이터는 라벨 시스템 내에서 기존 라벨을 검색하는데 사용되거나, 새로운 라벨을 생성하는 경우에 사용될 수 있다. 예컨대, 개발자가 거래를 관리하기 위한 모듈을 생성하고, "customer"라는 용어를 사용하는 구매자를 위한 라벨을 생성하려 한다면, 개발자는 라인(604)의 구매자를 위한 카테고리 코드를 입력할 것이다. 이러한 코드는 풀 다운 메뉴를 통해 수동으로 입력될 수도 있고, 또는 현재 시스템 설정을 사용하여 자동으로 입력될 수도 있다. 이러한 카테고리 코드는 해당 기능에 의해 사용되는 나머지 프로세스를 제어한다. 일반적으로, 라벨 대화의 카테고리 코드 및 네임스페이스 항목은 라벨 시스템의 현재 설정에 따른다. 데이터의 항목은 도 5의 블록(504)에 도시된다. 이러한 데이터는 일반 표현식 (regular expression)으로 입력된다. 또한, 대안적인 실시예에서, 개발자는 선택된 카테고리로 검색을 제한하는 체크 박스를 선택함으로써 검색을 제어할 수 있다.

[0064] 개발자는 버튼(650)을 선택함으로써 검색 기능을 활성화시킨다. 그러나, 라벨 검색 기능을 활성화시키기 위하여 다른 기술들이 사용될 수도 있다. 검색 엔진의 활성화가 도 5의 블록(505)에 도시되어 있다.

[0065] 그 후, 본 발명은 데이터베이스에서 입력된 텍스트와 일치하는 항목들을 찾기 위하여 데이터베이스를 검색한다. 검색 프로세스 동안, 진행 지시자(progress indicator)가 사용자에게 디스플레이 될 수 있다. 하나의 진행 지시자가 도 6에 참조 번호(640)로 도시되어 있다. 검색이 완료되면, 진행 지시자가 사라지고, 대화(dialog)는 초기 질의와 일치하는 모든 라벨들의 디스플레이를 선택된 언어로 개발자에게 반환한다. 이러한 과정이 블록 (506)에 도시되어 있다.

[0066] 일 실시예에서, 인터페이스(600)가 확대되어 도 6에 도시된 바와 같이 식별된 일치 결과를 디스플레이 한다. 이러한 실시예에서, 디스플레이된 일치 결과의 목록은 라벨 텍스트 테이블(240)로부터의 정보의 일부 뿐만이 아니라, 라벨 ID 테이블(230)에 포함된 정보를 포함할 수 있다. 결과 화면(result view)은 테이블(230)과의 관계를 통해 이용 가능한 추가 정보와 함께 텍스트들을 선택된 언어로 디스플레이 한다. 검색의 결과는 GUID에 의한 오름차 순(ascending order)으로 개발자에게 제시된다. 결과의 예가 도 6의 영역(610)에 디스플레이되어 있다. 그러나, 텍스트 일치에 의한 오름차 순과 같은 다른 정렬 기술을 사용할 수도 있다. 그 후, 개발자는 임의의 라벨 텍스트가 희망 텍스트와 일치하는지 확인하기 위하여 체크한다. 이러한 과정이 블록(520)에 도시되어 있다. 새로운 라벨에 대한 희망 텍스트와 일치하는 텍스트가 발견된 경우, 개발자는 희망 라벨 텍스트와 일치하는 특정 라벨을 강조하거나(highlight), 다르게 표시한다. 이러한 과정이 블록(507)에 도시되어 있다. 요구되는 경우, 라벨의 변환(transformation)이 620에 도시될 수 있다.

[0067] 다음으로, 시스템은 선택된 라벨 카테고리 및 네임스페이스가 새로운 라벨의 카테고리와 동일한지 확인하기 위하여 체크한다. 이러한 과정이 블록(508)에 도시되어 있다. 라벨 카테고리 및 네임스페이스가 선택된 라벨 및 새로운 라벨 사이에서 동일하다면, 선택된 라벨의 정보가 새로운 라벨을 위하여 사용된다. 이러한 과정이 도 5의 블록(509)에 도시되어 있다.

[0068] 라벨의 네임스페이스 및 카테고리가 새로운 라벨을 위해 요구되는 사용과 동일하지 않다면, 라벨은 복사되어야 한다. 또한, 이처럼 복사된 정보는 선택된 라벨의 모든 변화된 버전들을 포함한다. 라벨이 새로운 라벨로 복사되면, 복사된 라벨의 GUID를 나타내는, 새로운 라벨을 위한 항목이 라벨 ID 테이블(230)에 생성된다. 이러한 과정이 블록(510)에 도시되어 있다. 이러한 과정은, 마스터 라벨 텍스트가 변경된 경우에, 새로운 라벨의 텍스트를 쉽게 갱신하는 것을 가능하게 한다. 통상적인 찾기 및 바꾸기 프로토콜을 실행하거나 임의의 다른 자동화된 방법을 통해서, 변경에 따른 이러한 갱신을 수행할 수 있다.

[0069] 검색중에 어떠한 부합도 발견되지 않는다면, 개발자는 새로운 라벨을 위한 새로운 객체를 생성시켜야 한다. 새로운 객체의 생성은 블럭(521)에서 설명된다. 라벨의 생성중에, 라벨의 구체적인 특성들이 라벨 ID 표(230) 및 라벨 텍스트 표(240)에 저장된다. 라벨의 텍스트가 완전하지 않으면, 개발자는 라벨 대화(600)의 원하는 텍스트의 나머지를 관여한다. 네임스페이스 및 카테고리 코드들이 라벨 시스템의 현재 설정들을 기초로 기입된다. 또한, 개발자는 임의의 원하는 언어로 라벨 텍스트의 번역을 발생시켜야 한다. 또한, 라벨을 위한 현재 운영 언어에 대한 ID는 마스터 언어로서 저장된다. 이러한 라벨 속성들의 번역 및 입력의 추가 또는 생성은 블럭 (522)에서 설명된다.

[0070] 결론적으로, 본 발명은 비지니스 솔루션 소프트웨어 프로그램에서 라벨들을 저장하고 사용하는 새로운 방법에 관한 것이다. 각 언어의 단일의 리소스 파일을 구비하는 것 대신에, 정보는 1:n 관계로 분리된다. 관계의 '1' 측은 라벨 식별기 및 다른 실제적이고 일반적인 라벨 정보를 유지한다. 관계의 'n' 측은 특정 언어 상의 라벨 텍스트를 유지한다. 이러한 방식으로 라벨들을 구성하는 것은 실행중에 모든 언어들 상의 라벨들에 액세

스하는 것을 가능하게 한다.

[0071] 새로운 라벨 개발중에, 라벨은 시스템에 현재 있는 라벨을 사용하거나 스크래치로부터 새로운 라벨을 생성시킴으로써 복사될 수 있다. 라벨들을 검색하는 경우, 표현을 기입하는 것이 가능하다(예를 들어, '<Ledger'은 Ledger로 시작하는 모든 이용 가능한 라벨들을 나타낸다. 그러나, 시스템이 어디에서 구현되는지에 따라 다른 구문이 사용될 수 있다). 특정 네임스페이스 또는 카테고리가 검색이 되도록 선택함으로써 적중(hit)들을 감소시킬 수 있다. 또한, 용어 데이터베이스(term database)를 사용함으로써 검색이 수행될 수 있다. 또한, 용어 데이터베이스(웹-서비스)를 사용함으로써, 새로운 라벨을 생성시킬 수 있다. 용어 데이터베이스에 들어가는 경우, 현재의 검색 기준과 라벨이 사용될 실제 상황을 설명하는 여분의 기준을 결합하는 것이 가능하다. 이는 올바른 용어가 사용되는 것을 보증한다. 라벨이 용어 데이터베이스에서 발견되는 경우, 라벨은 새로운 식별기로 복사된다. 용어 데이터베이스는 지원되는 모든 언어들에 대한 번역들을 구비한다. 라벨이 사용될 경우, 검색 결과에서 실제 네임스페이스 및 카테고리를 부합하는 것이 나타나야 한다. 이러한 경우라면, 특정 라벨을 선택하는 것이 가능하다. 이러한 경우가 아니라면, 라벨 텍스트를 수기(手記)함으로써 또는 라벨을 용어 데이터베이스의 다른 네임스페이스/카테고리로부터 복사함으로써 새로운 라벨이 생성된다. 라벨을 복사하는 것은 현재의 네임스페이스 및 카테고리와 관련된 새로운 식별기를 준다. 모든 텍스트들 및 다른 일반적인 정보가 복사된다.

[0072] 라벨들의 관리는 특정 라벨 속성으로부터 직접 호출되는 라벨 대화를 사용하여 이루어진다. 대화는 라벨들을 유지하고, 특정 속성 상에서 사용하기 위한 특정 라벨을 선택하는 것을 가능하게 한다. 코드를 통한 라벨 시스템의 사용은 필요한 속성들을 주는 라벨 인터페이스를 통해 일어난다. 이는 라벨들이 저장되는 방식을 변경하는 것이 가능하게 한다. 그 저장은 SQL 메타데이터 저장, 리소스 파일 또는 웹 서비스일 수 있다. 실제로는 웹으로의 특정 액세스에 따른 모든 저장 형태의 조합일 수 있다.

[0073] 라벨 대화는 라벨들이 올바른 상황에서 사용된다는 것을 보증한다. 라벨 시스템에 카테고리들을 도입함으로써 라벨들이 올바르게 사용된다는 것을 보증한다. 그러나, 속성은 다른 목적을 갖는다. 라벨들을 번역하는 경우, 네임스페이스 정보 및 카테고리를 통하여 라벨들이 어디에서 사용될 것인지를 알 수 있다. 이는 사용될 올바른 용어를 찾을 수 있도록 한다. 라벨들이 어디에 사용될지를 파악함으로써 최적 실시 프레임워크(best practices framework)에 더 많은 최적 실시 체크를 추가할 수 있도록 한다. 본 발명은 알려진(또는 알려지지 않은) 번역 도구들에 적합한 특정 번역 파일들의 생성을 허용한다. 날짜 정보를 번역 파일들에 보냄으로써, 번역들을 가져올 때의 날짜를 체크하는 것이 가능하다. 이는 진행중인 번역 프로세스가 시스템의 최근 변경 날짜에 달려 있다는 것을 보장한다. 개신된 교차 참조 시스템을 이용하여, 특정 라벨이 어디에서 사용되는지를 알 수 있다. 또한, 특정 라벨에 대한 변화를 알 수 있다. 이러한 특징은 특정 라벨에 대한 변화가 단순한지를(예를 들어, 라인의 끝에 '.'을 추가) 결정하는 가능성을 주는 번역기와 관련이 있다.

### 발명의 효과

[0074] 본 발명이 특정 실시예들을 참조하여 기술되었다고 하여도, 본 기술분야의 당업자는 본 발명의 사상과 범위를 벗어나지 않으면서 형태 및 내용의 변화가 가능하다는 점을 인식할 것이다.

### 도면의 간단한 설명

[0001] 도 1은 본 발명이 사용될 수 있는 환경의 일 실시예를 도시하는 블록도.

[0002] 도 2는 본 발명의 라벨 시스템을 포함하는 테이블들을 도시하는 블록도.

[0003] 도 3a 및 3b는 라벨 ID 테이블과 라벨 텍스트 테이블 내의 필드들을 도시하는 도면.

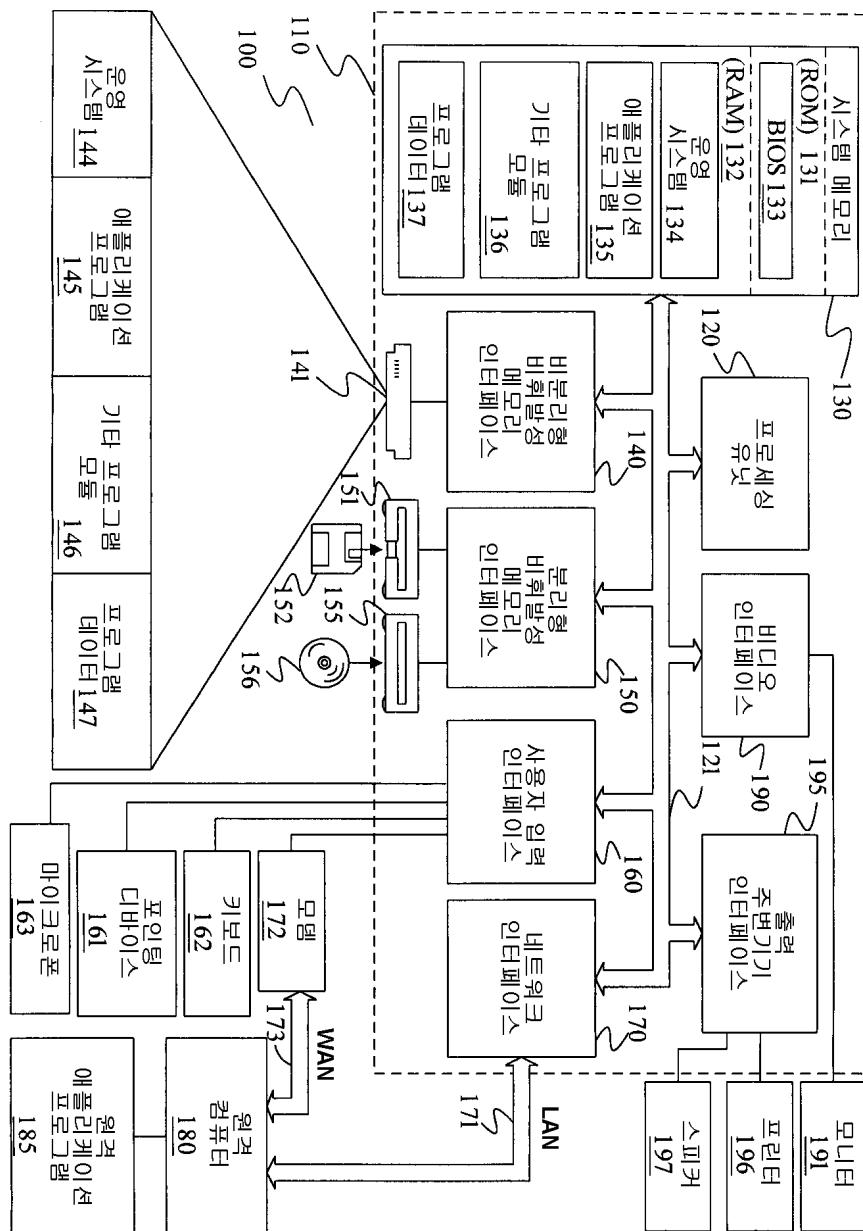
[0004] 도 4는 라벨 대화의 컴포넌트들 사이의 관계를 도시하는 블록도.

[0005] 도 5는 새 라벨이 생성될 때 실행되는 단계들을 도시하는 흐름도.

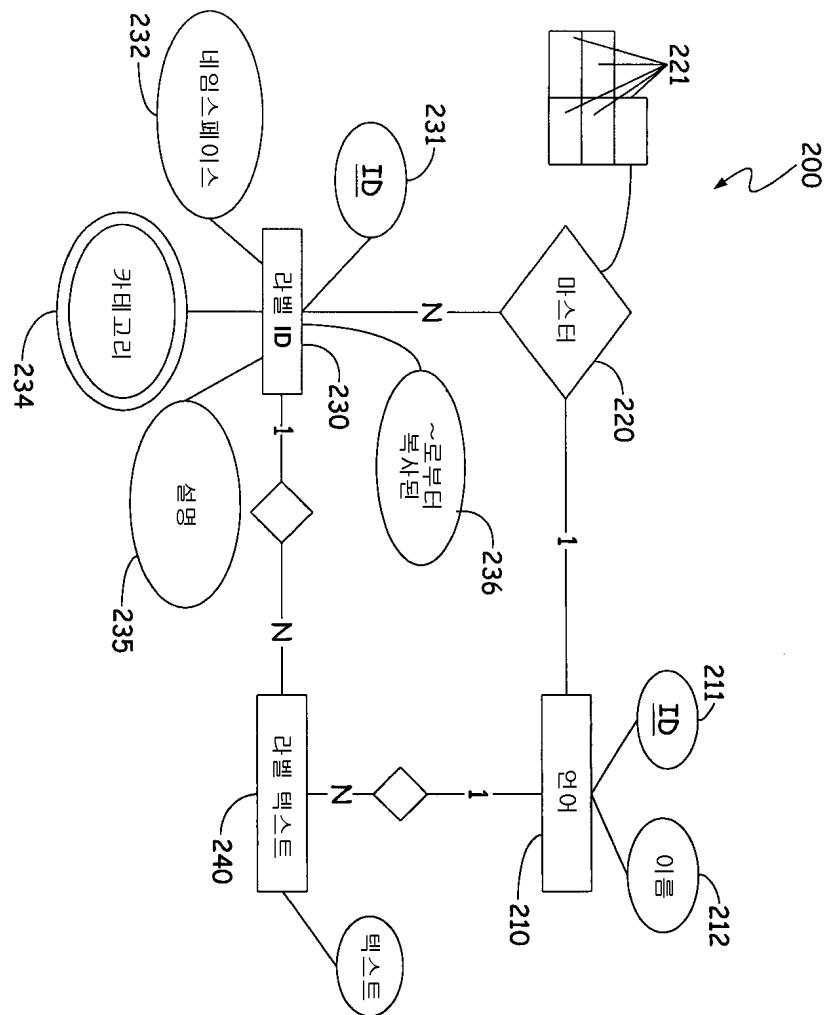
[0006] 도 6은 라벨 데이터베이스를 개발하고 검색할 때 사용자에 의해 사용되는 사용자 인터페이스의 실시예를 도시하는 도면.

## 도면

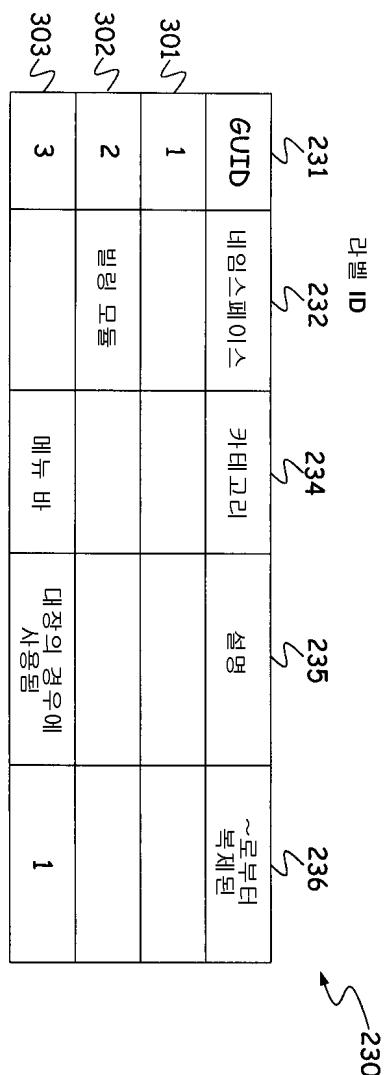
## 도면1



도면2



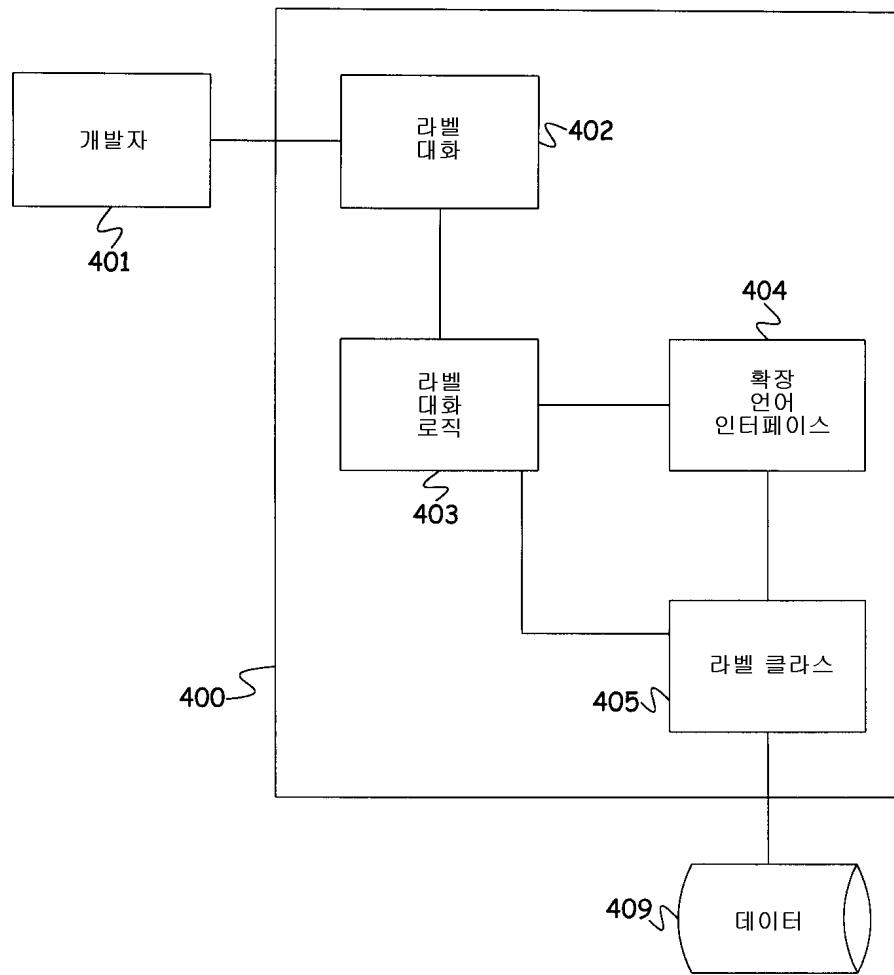
도면3a



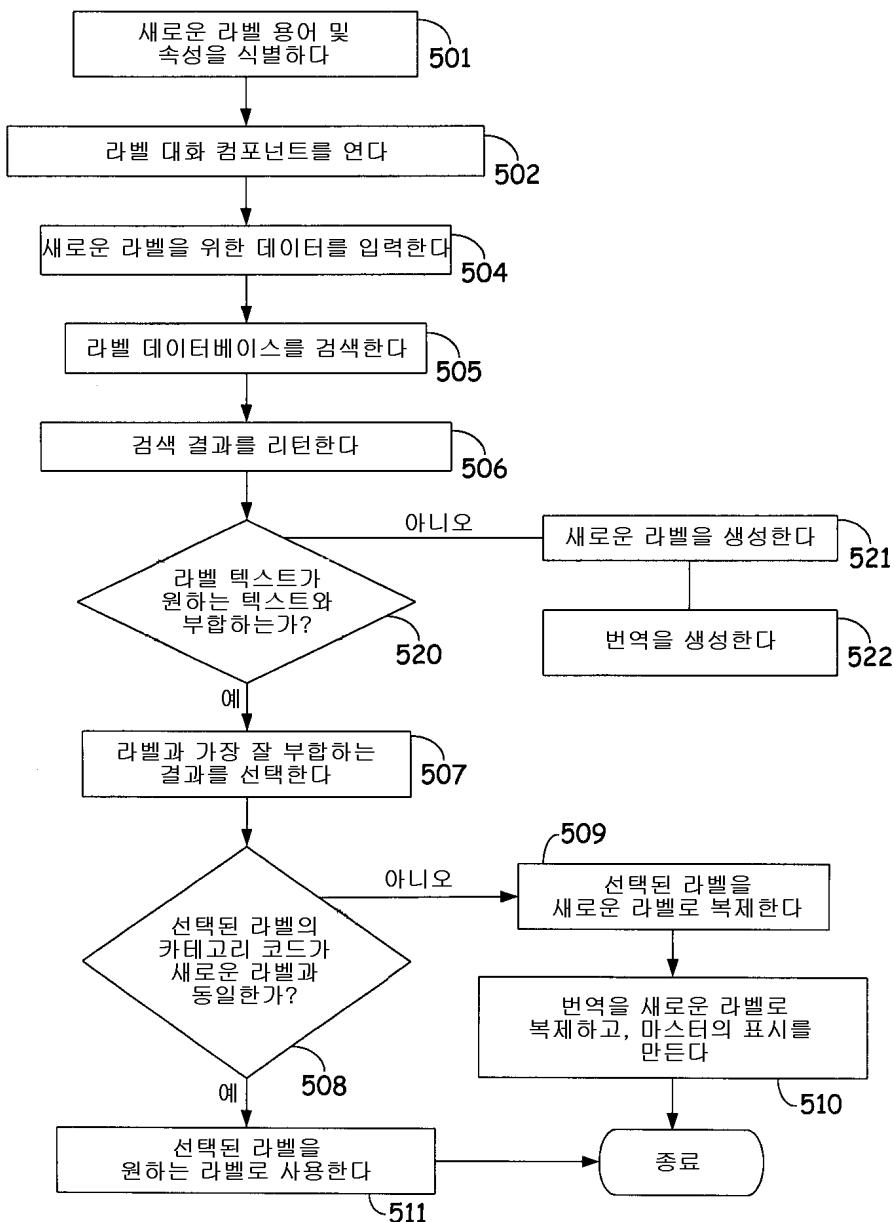
도면3b

라벨 ID	텍스트 ID	텍스트	편집 날짜	언어
1	351	DO YOU WANT TO SAVE THE CHANGES YOU MADE TO	22-AUG-2003	ENGLISH - UNITED STATES
1	352	MÖCHTEN SIE DIE ANDERWÜNGEN GEBILDET AN SPEICHEIN	27-AUG-2003	GERMAN - DEUTCH
1	353	LAVE JER VILLE GERNE OPSPAR DEN FORANDIGER JER INDRETTET TIL AT	27-AUG-2003	DANISH - DANSK
1	354	VOUS VOULEZ EN SAUVER LES CHANGEMENT VOUS FAIT Á	27-AUG-2003	FRENCH - FRANÇAIS

## 도면4



## 도면5



## 도면6

