OntoTerm^{TM}: un sistema abierto de representación conceptual

Antonio Moreno Ortiz
amo@uma.es
Universidad de Málaga
España

Resumen En la presente contribución se describe la aplicación OntoTerm, un gestor de ontologías creado con la finalidad de permitir el modelado de conocimiento general o específico en una red jerárquica conceptual denominada *ontología*. OntoTerm permite la creación de nuevas ontologías a partir de un número reducido de nodos de nivel superior (OBJETO, EVENTO, PROPIEDAD (ATRIBUTO, RELACIÓN)), facilitando el modelado conceptual mediante una interfaz de usuario amigable. Este software se está empleando actualmente dentro del proyecto OncoTer,m[1] en el que se está empleando para la creación, gestión y publicación HTML del recurso terminológico objeto del proyecto.

1 Introducción

La aplicación que presentamos aquí, tal y como está disponible en estos momentos,[2] consta de dos módulos fundamentales. El primero, denominado *Ontology Editor*, es donde tiene lugar la representación conceptual propiamente. La creación de esta aplicación se inspira en el software y ontología desarrollados en el Computing Research Laboratory (CRL) de la New Mexico State University bajo la dirección de S. Nirenburg (Viegas et al., 1999) bajo el proyecto de traducción automática basada en el conocimiento Mikrokosmos (μK). El segundo módulo, llamado *TermBase Editor*, está enfocado a la gestión de información terminológica. Aquí el dominio conceptual modelado en el anterior adquiere su dimensión lingüística. Así mismo, OntoTerm incluye otras herramientas como los árboles parciales, el navegador de ontologías o el generador de informes HTML, que permite publicar la ontología en formato HTML, convirtiendo las relaciones entre conceptos en hyperenlaces.

En esta demostración nos centramos en el módulo de representación conceptual.

2 Fundamentos

El objetivo primero fue desarrollar un sistema paralelo al gestor de ontologías de μK, capaz de crear y gestionar estructuras conceptuales de un modo parecido, pero que facilitase este tipo de tareas al usuario no experto. Al mismo tiempo se ha tratado de preservar las directrices que siguieron los creadores de este μK, para lo que es necesario, entre otras cosas, mantener los nodos superiores de esta ontología, así como el empleo de ranuras especiales tales como "dominio" y "rango" (Mahesh, 1996). En la actualidad, dentro del proyecto Oncoterm, utilizamos la ontología de μK como base para modelar conceptualmente el dominio específico de la oncología, tal y como se describe en (Ortiz & Hernández, 2000). OntoTerm es la aplicación que se encarga de gestionar esta ontología, así como las bases de datos terminológicas asignadas a la misma. Para poder integrar en una misma ontología distintos dominios, hemos implementado un sistema de gestión de subconjuntos.

3 Estructuras de datos

La caracterización de OntoTerm como sistema abierto se debe, fundamentalmente al tipo de soporte que emplea para almacenar la información. Durante su desarrollo, uno de los objetivos fundamentales ha sido garantizar la conectividad con otros sistemas y aplicaciones, de modo que el usuario experto sea capaz de emplear sus datos de un modo abierto. Para ello decidimos utilizar una base de

[1] Este trabajo se ha elaborado en parte dentro del marco del proyecto OncoTerm: sistema bilingüe de información y recursos oncológicos (PB 98-1342: proyecto subvencionado por el MEC).

[2] OntoTerm se encuentra disponible en el sitio web http://www.ontoterm.com/

datos relacional, concretamente el formato de Microsoft Jet 3.51 (MS Access 97). El diagrama E-R de la figura 1 muestra el nivel conceptual de una base de datos de ontología de OntoTerm.

De este modo, cualquier desarrollador puede acceder libremente a los datos generados por OntoTerm y gestionarlos para su utilización en cualquier aplicación. En el futuro nos planteamos la creación de una biblioteca de funciones o componente ActiveX que facilite el acceso a las funciones de consulta más comunes, especialmente a las más complejas, tales como la herencia.

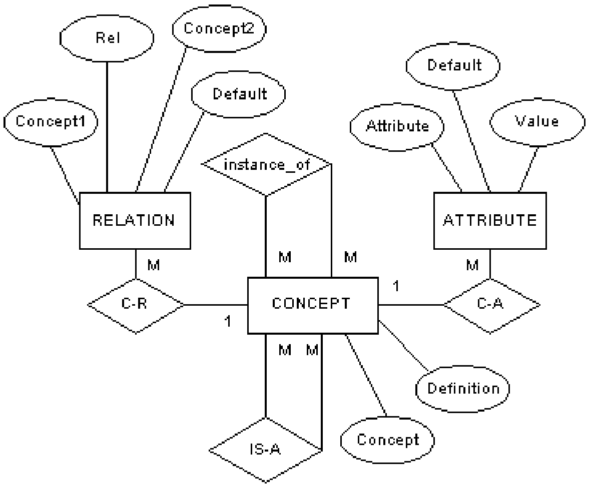

Figura 1: Diagrama E-R de la BD

Obviamente, las características de integridad relacional que aporta este tipo de sistema no son suficientes para mantener todas las restricciones necesarias en una ontología, de esto se ocupa tanto el código como la interfaz gráfica. Entre estas restricciones destacamos las siguientes:

- Todos los conceptos deben tener, al menos, un concepto padre.
- Aunque un concepto puede terner más de un concepto padre, todos deben ser de un mismo tipo (OBJETO, EVENTO, ATRIBUTO, RELACIÓN).
- Una relación o atributo sólo se puede usar para definir un conjunto concreto de objetos o eventos determinados, los definidos por su dominio. Así mismo, el valor de un atributo o la parte derecha de una relación está determinado por el rango de valores especificados para ese atributo o relación.

Además de esto, el código se encarga de gestionar la herencia de propiedades a lo largo de la jerarquía de conceptos. Los algoritmos de herencia (del tipo "primero en profundidad") se emplean no sólo para mostrar las propiedades heredadas al usuario, sino también para aplicar las restricciones señaladas anteriormente. Por ejemplo, cuando el usuario selecciona modificar las propiedades que definen a un objeto o evento, el algoritmo de herencia se encarga de recuperar sólo aquellos conceptos determinados por su dominio, es decir, el o los conceptos, marcados directamente por éste más todos sus descendientes. De este modo se impide que el usuario realice asignaciones de propiedades erróneas.

4 *La interfaz de usuario*

La interfaz de OntoTerm ha sido diseñada para facilitar al usuario no experto la interactividad con las estructuras de datos de una forma lo más gráfica posible. Para conseguir esto, utilizamos controles estándar de Windows y hacemos uso extensivo de los menús de contexto. Todo, excepto la entrada de caracteres, se puede llevar a cabo empleando únicamente el ratón. También existen teclas de métodos abreviados para las tareas más comunes. No obstante, la interfaz no sirve únicamente como medio para facilitar las tareas de edición comunes, sino que también se encarga de aplicar un buen número de restricciones sobre los datos.

Referencias

Mahesh, K. (1996). Ontology development for machine translation: Ideology and methodology. Inf. Téc. MCCS-96-292, Computing Research Laboratory. New Mexico State University.

Ortiz, A. M. & C. P. Hernández (2000). Reusing the mikrokosmos ontology for concept-based multilingual terminology databases. En *Proceedings of 2nd LREC Conference*, Athens, Greece.

Viegas, E., K. Mahesh, S. Nirenburg, & S. Beale (1999). Semantics in action. En *Predicative Forms in Natural Language and in Lexical Knowledge Bases* (editado por P. S. Dizier), Dordrecht, the Netherlands: Kluwer Academic Publishers, págs. 171–204.