OEG Publication

Aguado de Cea G, Bañón A, Bateman J, Bernardos MS, Fernández-López M, Gómez-Pérez A, Nieto E, Olalla A, Plaza R, Sánchez A

Ontogeneration: Arquitectura basada en ontologías para la generación de textos en castellano

Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 1999)
Actas de la VIII Conferencia de la Asociación Española para la Inteligencia Artificial
November 1999.
Murcia, Spain.
Pages: 78 to 87
ISSN: 931170-0-5

Ontogeneration: Arquitectura basada en ontologías para la generación de textos en castellano[1]

G. Aguado[1] A. Bañón[2] J. Bateman[3] S. Bernardos[2] M. Fernández[2] A. Gómez-Pérez[2]
E. Nieto[2] A. Olalla[2] R. Plaza[1] A. Sánchez[2]
[1] Departamento de Lingüística Aplicada. Facultad de Informática. Univ. Politécnica de Madrid
[2] Laboratorio de Inteligencia Artificial. Facultad de Informática. Univ. Politécnica de Madrid
Campus de Montegancedo, sn.
28660 Boadilla del Monte, Madrid, España.
[1]e-mail: {lupe, rplaza}@fi.upm.es
[2]e-mail: {abanon, sgalindo, mfernand, asun, enieto, aolalla, alvaro}@delicias.dia.fi.upm.es
[3] Comunication and Language Research, Dept of English Studies. University of Stirling.
Stirling FK9 4LA, Scotland, UK.
e-mail: j.a.bateman@stir.ac.uk

Palabras clave: ontología, generación de lenguaje natural, recuperación de información, gramática sistémica funcional, GUM, KPML.

RESUMEN. *En este trabajo se presenta Ontogeneration, una arquitectura para generar textos en castellano utilizando ontologías lingüísticas y de dominio con la tecnología KPML de generación de lenguaje natural. Además se ha construido un sistema que genera textos en castellano en el dominio de las sustancias químicas. Para alcanzar tales resultados, se han seguido los siguientes pasos: a) se ha tomado como fuente de conocimiento una ontología en el dominio químico construida usando la metodología de desarrollo de ontologías llamada METHONTOLOGY y el Entorno de Diseño de Ontologías (Ontology Design Environment: ODE), b) se ha extendido y modificado la ontología lingüística GUM (Generalized Upper Model) para el castellano, c) se ha construido una gramática para el castellano siguiendo el modelo sistémico-funcional usando el entorno KPML (Komet Penman Multilingual). Se consigue, así, que los contenidos almacenados en la ontología de dominio sean accesibles por usuarios legos en la materia.*

1.- INTRODUCCIÓN.

Uno de los principales objetivos de las **ontologías** (Uschold & Gruninger 96) es aumentar el entendimiento compartido en un dominio dado, eliminando así las diferencias, solapamientos y malentendidos en conceptos, estructuras, terminologías, etc. De este modo, las ontologías pueden funcionar como un marco de trabajo que unifica distintos puntos de vista y mejora la comunicación. Gruber definió una ontología como "una especificación explícita de una conceptualización" (Gruber 93a) que incluye: conceptos, instancias, relaciones, funciones y axiomas. Sin embargo, mientras que, por un lado, las ontologías generalmente especifican una conceptualización con un alto grado de formalidad, por el otro, hay pocas metodologías para construir ontologías. Además, se pueden utilizar distintos formalismos y lenguajes para implementarlas. Este hecho hace imposible que los usuarios sin una cierta experiencia en este campo puedan reutilizar, consultar o comprender los conocimientos incluidos en ellas. Una manera práctica de resolver parcialmente este problema es presentar el contenido de la ontología en un conjunto de representaciones intermedias en el nivel de conocimientos (Newell 82). El marco de trabajo METHONTOLOGY (Fernández *et al.* 97) facilita precisamente esa solución, y el entorno de desarrollo ODE (*Ontology Design Environment*) (Blázquez *et al.* 98) ayuda a los expertos en ontologías y a los expertos de dominio construir ontologías en el nivel de conocimientos utilizando tablas y grafos en vez de lenguajes formales para codificarlas. Los traductores de ODE generan

[1] Este trabajo ha sido financiado por el programa "Ayudas de I+D para grupos potencialmente competitivos" de la Universidad Politécnica de Madrid (referencia A9706). Puede consultarse en http://delicias.dia.fi.upm.es/proyectos/terminados/ontogeneration/ontogeneration_proyecto_Esp.html

automáticamente ontologías estándares, coherentes y bien estructuradas en varios lenguajes computables (Ontolingua (Gruber 93b), SFK (Fischer & Rostek 93), SQL). ***Chemicals*** (Fernández *et al.* 99) es una de las ontologías construidas con este método y con este entorno.

Una vía más eficaz de difundir los contenidos expresados formalmente en una ontología es traducirlos a lenguaje natural. Expresar en palabras toda la información representada en una ontología es el mejor modo de facilitar su acceso para cualquier usuario. La generación de textos en lenguaje natural, a partir de los contenidos de una ontología, también permitiría a los expertos del dominio evaluar los conocimientos ya formalizados en las ontologías. Además posibilitaría la reutilización de las ontologías de dominio en aplicaciones prácticas y comerciales relacionadas con la generación de texto multilingüe, la gestión de conocimientos, la recuperación de información "on-line", las explicaciones en lenguaje natural en sistemas expertos o en sistemas inteligentes de tutoría, y el acceso a bases de datos, entre otros.

El establecer una conexión entre los campos de la generación de textos y la reutilización de ontologías también es beneficioso para la generación de textos[2]. Un problema clave y muy extendido en los sistemas de generación es la ausencia de modelos de dominio organizados adecuadamente. Las fuentes de conocimientos usadas para los sistemas de generación suelen estar hechas a mano (orientadas a la lengua final) o están construidas con fines no lingüísticos, lo cual dificulta la producción de lenguaje y no aconseja el uso de la tecnología de generación de textos. De nuevo, una posible solución es reutilizar ontologías de dominio estandarizadas o bien definidas como un recurso de representación para los sistemas de generación de texto.

Dado que las ontologías de dominio desarrolladas con METHONTOLOGY y ODE no incluyen características lingüísticas, se hace necesario el uso de una interfaz entre la ontología y los sistemas de generación de lenguaje natural. Un medio que ya se ha demostrado muy eficaz en sistemas de generación es el uso de la ontología lingüística llamada **GUM** (*Generalized Upper Model*) (Bateman *et al.* 95). GUM ofrece un nivel de abstracción semántica que está lo suficientemente lejos de las realizaciones superficiales como para facilitar el enlace con modelos de dominio bien estructurados, pero que, a la vez, está lo suficientemente cercano a la forma lingüística como para permitir correspondencias bien definidas de sus conceptos con la expresión lingüística.

En íntima relación con GUM, se encuentra el entorno de desarrollo **KPML** (*Komet Penman Multilingual*) (Matthiessen & Bateman 91). Esta herramienta sirve para construir y mantener recursos lingüísticos multilingües y para su uso en la generación de textos.

Con GUM y con KPML, y usando las ontologías de dominio construidas con METHONTOLOGY, se ha construido ***Ontogeneration***, que es una arquitectura para la recuperación de información que permite a los usuarios consultar y acceder en castellano a los conocimientos explícitos contenidos en una ontología de elementos químicos. Su desarrollo sigue en progreso aunque su estado actual sirve de ejemplo claro de cómo los dos campos de investigación, ingeniería ontológica y generación de lenguaje natural se pueden enlazar, resolviendo así algunos problemas clave en ambas disciplinas. Nuestra meta, en este artículo, no ha sido el proceso de generación en sí mismo y, de hecho, reutilizamos deliberadamente todo lo que podemos de la tecnología y técnicas de generación establecidas. Nuestra contribución principal se debe ver desde el punto de vista de:

a) la reutilización de dos tipos distintos de ontologías (de dominio y lingüísticas) construidas por separado con diferentes tecnologías y propósitos,
b) la reutilización de la tecnología KPML para construir recursos para la generación de textos en castellano y
c) la integración de estos recursos en una arquitectura nueva y la construcción de una aplicación nueva que genera textos en castellano en el dominio de la química.

En este artículo primero describiremos el diseño general de esta arquitectura, y cómo se pueden combinar ontologías y generación de textos para probar los beneficios de nuestro enfoque. En segundo lugar, dado que nuestro sistema integra recursos desarrollados independientemente: una ontología de dominio (*Chemicals*, almacenada en una base de datos relacional), una ontología lingüística (GUM, implementada en Loom) y un entorno de generación (KPML, en CommonLisp) ya utilizados en otros

[2] Son varios los sistemas de generación desarrollados con distintos fines, como PlanDoc (McKeown *et al.* 1994), TECHDOC (Rösner 1994) y ModelExplainer (Lavoie *et al.* 1996), entre otros, aunque la reutilización de ontologías con esta finalidad no es frecuente.

proyectos, explicaremos estos recursos, por qué y cómo los hemos reutilizado, su adaptación o extensión al castellano y su integración en la arquitectura *Ontogeneration*.

2.- ARQUITECTURA.

El diseño general de *Ontogeneration* se muestra en la figura 1. Esta figura se divide en tres niveles: entrada/salida, procesos y recursos. Cualquier sistema de generación está guiado por un objetivo comunicativo. En nuestro caso, el principal objetivo comunicativo es ofrecer toda la información pedida por el usuario sobre un dominio dado.

El sistema comienza cuando el usuario hace una consulta específica usando una interfaz basada en menús. Todas las posibles consultas han sido definidas previamente y clasificadas en varios patrones. Así, el usuario puede componer fácilmente una consulta concreta seleccionando las distintas opciones de menú.

El primer proceso es la **búsqueda y selección de conocimientos** en la ontología de dominio. Nuestro sistema usa *Chemicals*, una ontología que describe y clasifica todos los elementos químicos (véase la sección 3). Otro recurso previsto, aunque no desarrollado en la versión actual, es el modelo de usuario. Este modelo sirve para dirigir el proceso de selección y generar textos acordes con su perfil. En este proceso, es preciso decidir qué información debe omitirse, medir el tono o grado de formalidad y otros aspectos pragmáticos (aspectos interpersonales y de situación).

El siguiente proceso, la **planificación de textos**, organiza los conocimientos seleccionados en un esquema retórico adecuado. Nuestros esquemas retóricos representan patrones estándares del discurso científico. Los esquemas retóricos pueden ser descritos como modelos de un conjunto de párrafos estereotípicos que se han identificado en textos científicos y manuales. Estos esquemas retóricos guían la planificación de textos en el diseño de su estructura, lo cual incluye la organización de los contenidos con un discurso coherente, la descomposición del párrafo en oraciones, el uso de expresiones de referencia y elipsis, y la elección de construcciones sintácticas marcadas con efectos retóricos.

El proceso de **realización gramatical** (en inglés, *linguistic realization*) es responsable de generar el texto final transformando el plan de texto dado por el planificador en una representación lingüística concreta. Para ello se sirve de la ontología lingüística GUM extendida al castellano (Bernardos 97) y de los recursos lexicogramaticales que también están siendo adaptados al castellano (Bañón 99 & Nieto 99).

Finalmente, el texto generado es editado por la interfaz de usuario para mejorar la salida final. La interfaz de nuestro sistema está preparada para controlar distintos modos de formatear el texto (incluyendo tablas, y gráficos) y, en un futuro podría incorporar ciertos aspectos de presentación multimodal (hipertexto, dibujos en 3D, salida de voz, etc.).

El prototipo actual de *Ontogeneration* funciona como un sistema interactivo de recuperación de información: el usuario puede consultar los contenidos de la ontología de dominio construyendo una consulta adecuada con las opciones del menú de la interfaz y el sistema responde generando un texto en castellano con la información requerida. Se pueden recuperar distintos tipos de información y datos de nuestra ontología de dominio: definiciones, clasificaciones, ejemplos, comparaciones o descripciones completas de elementos químicos: definiciones (de grupos, elementos, axiomas, fórmulas y propiedades); comparaciones (entre grupos o elementos), ejemplos (de grupos o elementos), clasificaciones (de grupos o elementos) y otros (descripciones o especificaciones completas).

3.- LA ONTOLOGÍA DE DOMINIO: *CHEMICALS*.

Como ya se ha mencionado, *Ontogeneration* utiliza una ontología de dominio bien definida construida con fines no lingüísticos como fuente de conocimientos para la generación de texto. *Chemicals* es una ontología de dominio desarrollada en el marco de trabajo METHONTOLOGY (Fernández *et al.* 99) y usando ODE (Blázquez *et al.* 98). METHONTOLOGY incluye: la identificación del proceso de desarrollo de la ontología que se refiere a qué tareas se deben hacer cuando se está construyendo una ontología; una propuesta de un ciclo de vida basado en prototipos evolutivos y la metodología misma que especifica los pasos que se han de seguir para realizar cada actividad, las técnicas utilizadas, los productos de salida y el modo en que han de ser evaluados. La fase principal en el proceso de construcción de la ontología es la de conceptualización. En la figura 2 se muestran las representaciones intermedias usadas durante el desarrollo de *Chemicals*.

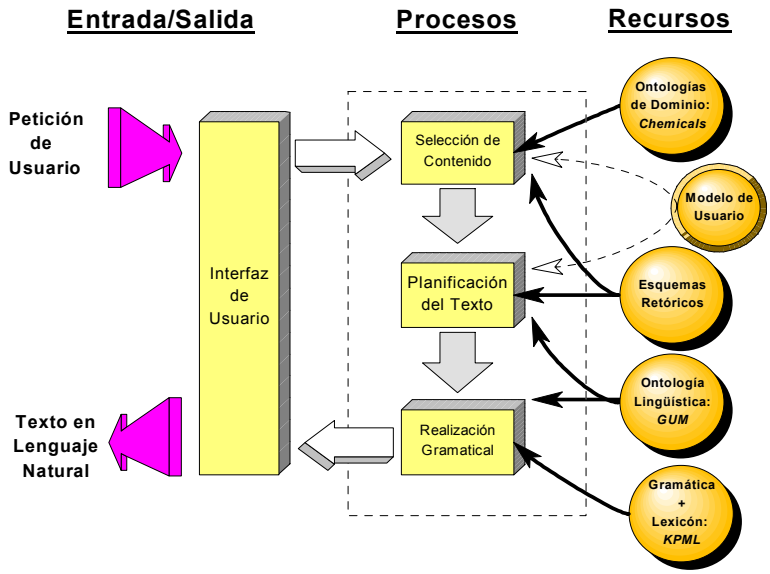


Figura 1: Arquitectura del sistema

La conceptualización de *Chemicals* se ha realizado usando un conjunto de representaciones intermedias tabulares (véase figura 3) independientes del lenguaje meta computable en el que se implementará la ontología. Dado que en esta conceptualización se han utilizado términos en inglés, para su uso en nuestra arquitectura se ha creado un lexicón con sus correspondientes términos en castellano. Por otra parte, el módulo generador multilingüe de ODE traduce automáticamente el modelo de conocimientos en lenguajes comprensibles por la máquina. También incluye traductores inversos que traducen código en Ontolingua a nuestras estructuras de representaciones intermedias mediante un proceso de ingeniería inversa (Gómez-Pérez & Rojas 99). De la misma manera, la arquitectura *Ontogeneration* está preparada para tomar como fuente de conocimientos cualquier ontología codificada en Ontolingua que haya sido previamente transformada en nuestra notación.

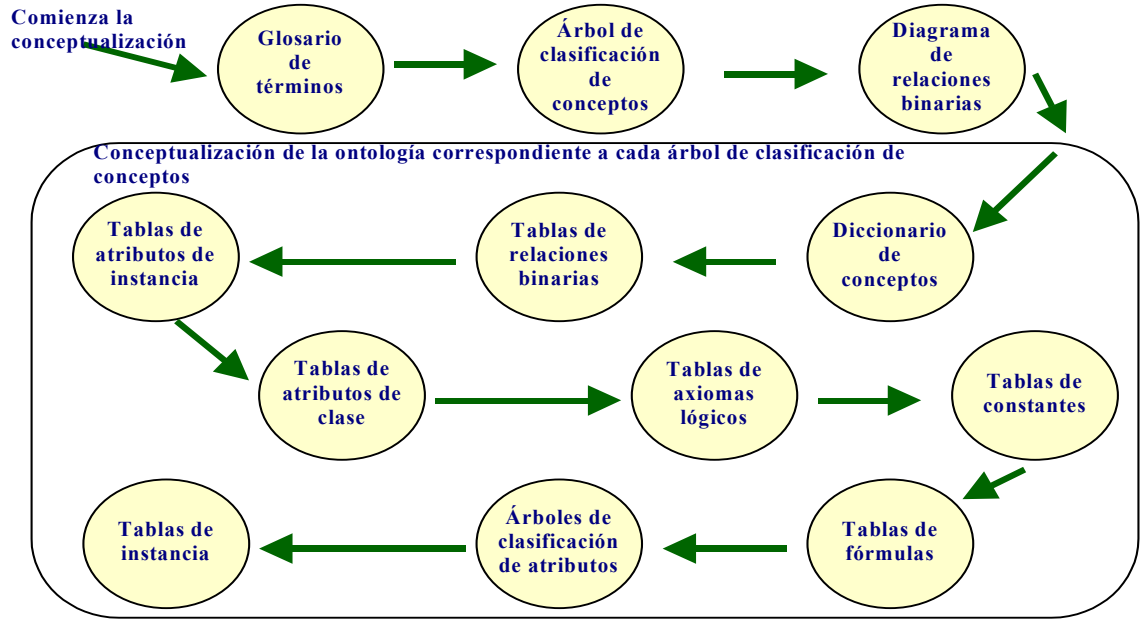
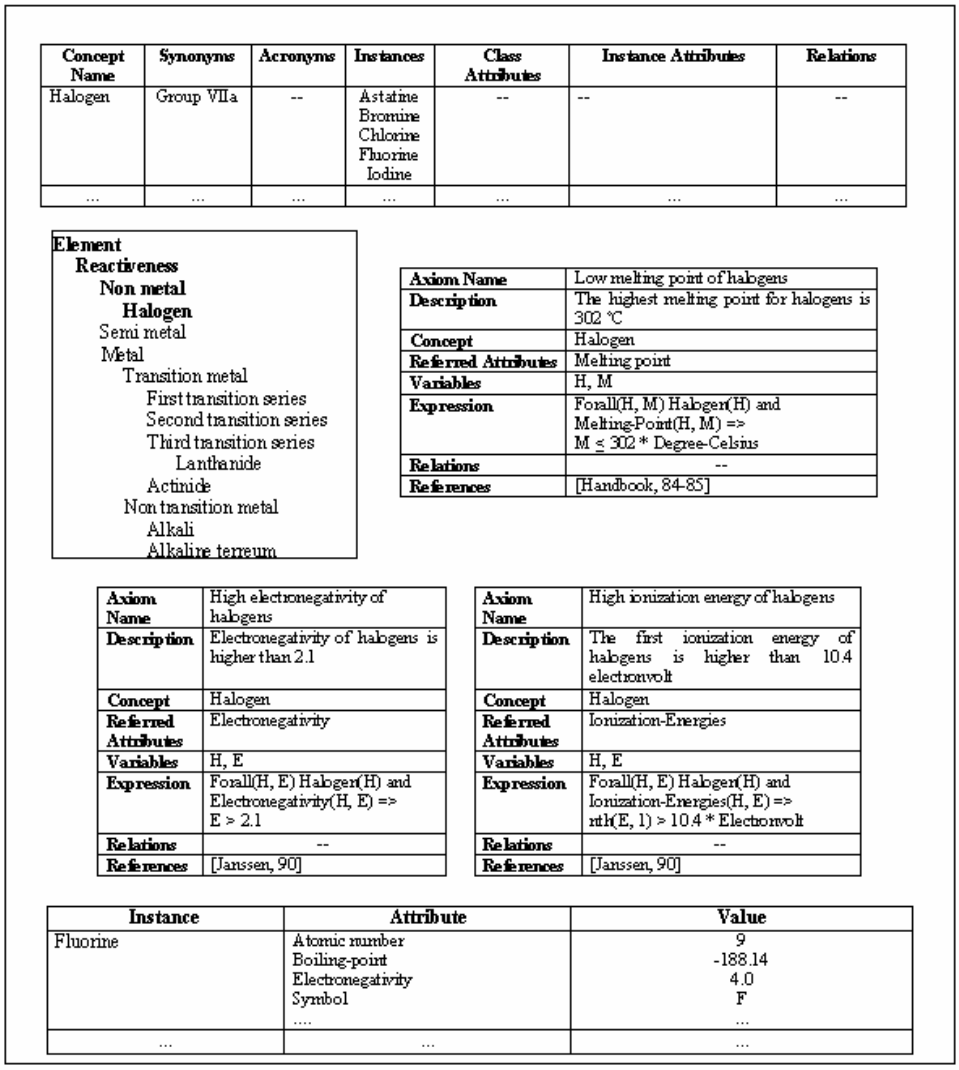


Figura 2: Fase de conceptualización



Concept Name	Synonyms	Acronyms	Instances	Class Attributes	Instance Attributes	Relations
Halogen	Group VIIa	--	Astatine Bromine Chlorine Fluorine Iodine	--	--	--
...	...	...	...	...	...	...

Element
- **Reactiveness**
  - **Non metal**
    - **Halogen**
  - Semi metal
  - Metal
    - Transition metal
      - First transition series
      - Second transition series
      - Third transition series
        - Lanthanide
      - Actinide
    - Non transition metal
      - Alkali
      - Alkaline terreum

Axiom Name	Low melting point of halogens
Description	The highest melting point for halogens is 302 °C
Concept	Halogen
Referred Attributes	Melting point
Variables	H, M
Expression	Forall(H, M) Halogen(H) and Melting-Point(H, M) => M ≤ 302 * Degree-Celsius
Relations	--
References	[Handbook, 84-85]

Axiom Name	High electronegativity of halogens
Description	Electronegativity of halogens is higher than 2.1
Concept	Halogen
Referred Attributes	Electronegativity
Variables	H, E
Expression	Forall(H, E) Halogen(H) and Electronegativity(H, E) => E > 2.1
Relations	--
References	[Janssen, 90]

Axiom Name	High ionization energy of halogens
Description	The first ionization energy of halogens is higher than 10.4 electronvolt
Concept	Halogen
Referred Attributes	Ionization-Energies
Variables	H, E
Expression	Forall(H, E) Halogen(H) and Ionization-Energies(H, E) => nth(E, 1) > 10.4 * Electronvolt
Relations	--
References	[Janssen, 90]

Instance	Attribute	Value
Fluorine	Atomic number Boiling-point Electronegativity Symbol	9 -188.14 4.0 F ...
...	...	...

Figura 3: Ejemplo de conocimientos representados en Chemicals

4.- LA ONTOLOGÍA LINGÜÍSTICA: GUM.

En esta sección se presenta cómo se ha reutilizado la ontología lingüística GUM (Bateman *et al.* 95), y cómo se ha adaptado al castellano, con el nombre de GUME (Bernardos 97). GUM se utiliza para simplificar la interfaz entre los conocimientos del dominio y los componentes lingüísticos. Se trata de una ontología abstracta motivada lingüísticamente y utilizada en otros proyectos de generación para diferentes lenguas: inglés, alemán e italiano. GUM ofrece una clasificación de los tipos de significado que las construcciones gramaticales presuponen. Así, tiene un papel importante proporcionando la semántica para conceptos del dominio y conectando representaciones conceptuales con representaciones léxicas.

GUM es, pues, una ontología lingüística con un alto nivel de abstracción que está a medio camino entre las realizaciones lingüísticas y las representaciones "conceptuales" o "contextuales". Es decir, posibilita la abstracción más allá de los detalles concretos de las representaciones léxicogramaticales, a la vez que mantiene el suficiente contacto con las realizaciones lingüísticas como para permitir trabajar con los componentes de lenguaje natural.

GUM está organizada en dos jerarquías: una de conceptos y una de relaciones. La jerarquía de conceptos representa las entidades semánticas básicas de las gramáticas de lenguaje natural, incluyendo configuraciones de procesos y las distintas clases de objetos y cualidades. La jerarquía de relaciones representa los participantes y las circunstancias involucradas en los procesos y las combinaciones lógicas entre ellos.

Dos son las razones principales por las que hemos elegido GUM como base para desarrollar *Ontogeneration*.

a) Los trabajos previos realizados con GUM muestran que puede proporcionar una base sólida para la generación de lenguaje natural cuando la organización del dominio está aislada de los detalles de su realización lingüística (Bateman & Teich 95). Por tanto, usar GUM como una interfaz asegura que no tenemos que importar distinciones motivadas lingüísticamente a nuestra ontología de dominio para permitir la generación de lenguaje natural. Lo contrario comprometería el modelo de dominio considerablemente y está generalmente reconocido como una violación de la modularidad deseada en un sistema completo.
b) El trabajo previo en el desarrollo de recursos lingüísticos multilingües para la generación de lenguaje natural ha mostrado que este proceso se puede acelerar significativamente si se reutilizan las correspondencias que son necesarias entre las representaciones semánticas y las formas gramaticales. GUM permite esta reutilización proporcionando un núcleo fijo que es lo suficientemente general como para necesitar sólo variaciones pequeñas entre las lenguas. Es posible minimizar los aspectos de la descripción semántica específicos de una lengua sin necesidad de adoptar una posición interlingüe. Esta reutilización de GUM es uno de los factores principales en su origen (Bateman & Teich 95).

Ambas razones apoyan el uso de GUM en el sistema actual y permiten la reutilización de cuerpos de información significativos, en los niveles de descripción tanto gramaticales como semánticos además de posibilitar la generación de lenguaje natural. La investigación de la aplicabilidad de GUM para permitir la generación en lenguas como inglés, holandés, francés e italiano hacía suponer que la extensión al castellano tendría grandes probabilidades de éxito, como así ha sido.

4.1. Método para adaptar GUM al castellano.

Los principales criterios seguidos para adaptar GUM al castellano, de modo que se pudieran reutilizar lo más posible sus conceptos y relaciones, han sido:

- Considerar las distinciones en sus expresiones léxicogramaticales y reflejar las diferencias en el significado "experiencial"[3].
- Clasificar las categorías en dimensiones, particiones, disyunciones y especializaciones simples.
- Incluir las configuraciones con distinto número de participantes y circunstancias en distintas representaciones conceptuales. Estos participantes y circunstancias pueden aparecer explícitamente o no.
- Tener en cuenta expresiones donde las variaciones sintácticas relativas al orden en que aparecen los componentes produce un cambio de significado.
- Permitir generar una realización lingüística desde distintos puntos de vista.
- Maximizar la cohesión entre GUM y GUME para reducir el trabajo que haya que realizar en una futura integración multilingüe.

El procedimiento seguido para esta adaptación ha sido el siguiente: En primer lugar. se ha estudiado en profundidad cada categoría de las jerarquías de conceptos y relaciones mencionadas anteriormente. En segundo lugar, se han comparado los comportamientos lingüísticos del castellano con los que refleja cada categoría. Si existía alguna discrepancia, se han propuesto las extensiones correspondientes, siguiendo los criterios de diseño ya expuestos. Además se han analizado los tipos de textos que se van a generar y se han considerado las categorías de GUM dentro de las que se pueden clasificar sus componentes. Cuando algún componente no puede ser clasificado en ninguna categoría de GUM o se incluye en una categoría muy abstracta, se han creado las especializaciones necesarias para representar ese tipo de conocimientos.

El resumen en números del resultado obtenido es el siguiente: se han identificado 185 categorías de GUM totalmente válidas (119 conceptos y 66 relaciones); se han identificado 6 categorías de GUM no válidas (3 conceptos y 3 relaciones); se han identificado 20 categorías de GUM que necesitan modificaciones (12 conceptos y 8 relaciones) y se han implementado estas modificaciones; se han

[3] Según Halliday (85), cualquier oración debe tener contribuciones de tres metafunciones: textual, interpersonal e ideacional (dividida en lógica y "experiencial"), cada una de las cuales da lugar a un tipo distinto de restricción.

identificado e implementado 15 categorías nuevas (11 conceptos y 4 relaciones). Finalmente, la ontología GUME (Bernardos 97) consta en total de 134 conceptos y 77 relaciones.

5.- ENTORNO DE GENERACIÓN: KPML.

Como tercer soporte de este trabajo hemos reutilizado el entorno de desarrollo KPML (Matthiessen & Bateman 91) para construir los recursos gramaticales para el castellano. KPML es un sistema para construir y mantener recursos lingüísticos multilingües y para usar estos recursos en la generación de textos (en la actualidad inglés y alemán). Con el uso de este sistema intentamos simplificar las tareas de generación y mejorar el acceso y el manejo de los recursos. Se ha elegido KPML porque:

- Ofrece recursos lingüísticos ya probados y verificados para proyectos de generación de gran envergadura y facilita especificaciones de entrada y salida estándares adecuadas para la generación práctica.
- Ofrece a los proyectos de generación un motor básico para usar esos recursos.
- Fomenta el desarrollo de recursos estructurados de forma similar para lenguas que no tienen esos recursos.
- Minimiza los costes de proporcionar textos en múltiples lenguas.
- Permite desarrollar proyectos más complejos reutilizando otras ontologías de dominio que ya han sido creadas, además de incluir recursos de castellano en el entorno multilingüe.

Las gramáticas de KPML, cuyas unidades básicas consisten en sistemas gramaticales, electores (*choosers*), preguntas (*inquiries*), elementos léxicos, reglas de puntuación y especificaciones de entrada en forma de SPL (*Specification Planning Language*) (Kasper 89), son redes de sistemas definidas según la lingüística sistémica-funcional (Halliday 85) y construidas como árboles de opciones de comunicación.

El motor de generación de KPML usa la red de sistemas para construir cadenas de caracteres atravesando la red de sistemas de izquierda a derecha, para cada constituyente gramatical que se ha de generar. En cada sistema gramatical sólo se selecciona una característica gramatical. Cada característica seleccionada puede tener un conjunto de restricciones sintácticas para referirse a la estructura sintáctica total que se está generando.

La generación está completa cuando las estructuras construidas están lo suficientemente desarrolladas como para permitir la inserción de elementos léxicos, que pueden haber sido elegidos en cualquier momento durante el proceso de generación.

La selección de una característica gramatical en un sistema gramatical está determinada por un elector para ese sistema. El elector hace su selección atravesando un árbol de decisión de preguntas semánticas.

Las principales funciones que realiza KPML son: la inspección gráfica de los recursos de la gramática y de los ejemplos a generar, la creación y modificación gráfica de dichos recursos, la depuración de los recursos a través de trazas de ejemplos y la generación propiamente dicha.

5.1. Desarrollo de la gramática para el castellano.

KPML se puede usar con muchas lenguas puesto que permite que cualquier sistema con un nombre específico pueda especializarse de modo diferente dependiendo de la lengua que se esté usando. Así mismo, con KPML, como ya se ha mencionado, se pueden construir conjuntos de recursos lingüísticos para distintas lenguas, bien desde cero, o bien utilizando los recursos creados para otras. En este caso, se ha elegido la segunda vía. En el proceso seguido para generar textos en castellano se han tomado los elementos básicos del inglés como base para llevar a cabo las siguientes tareas:

1. Se identificó un conjunto de textos representativos en el dominio de los elementos químicos.
2. Se estudiaron los recursos gramaticales del inglés y se seleccionaron las especificaciones de entrada más representativas que tienen correspondencia con los posibles textos que se van a generar en castellano.
3. Se trabajó con un conjunto reducido y representativo de especificaciones de entrada para castellano, obtenido adaptando las disponibles para inglés o construyéndolas directamente para castellano cuando no existían ejemplos similares en inglés. En ambos casos ha sido necesario hacer cambios semánticos y léxicos en la gramática.

4. Una vez depurado el conjunto de nuevos recursos, éstos se pueden unir, si es necesario, con el conjunto general de recursos multilingües si se necesita. Este hecho permite añadir nuevas lenguas al sistema de generación.

Durante el desarrollo, hemos encontrado que el conjunto de cambios necesarios para la extensión al castellano no eran tan numerosos como inicialmente se estimó (véase (Nieto 99) para una relación exhaustiva). Como regla general, esto suponía añadir sistemas (puntos de elección en la jerarquía de clasificación gramatical), eliminar sistemas, añadir y eliminar restricciones a la estructura, etc. En resumen, todas las operaciones que, como se "sabe" son eficaces para lograr una cobertura de una nueva lengua dentro de la metodología general de KPML.

En general, durante el desarrollo de la gramática para castellano, hemos encontrado que el grado total de compartición y reutilización de recursos entre la gramática inglesa original y la gramática castellana actual es muy alto. De los 745 sistemas gramaticales, o puntos de elección, utilizados en la gramática del español, unos 724 (97%) están compartidos con los del recurso inglés inicial. Estos puntos de elección distribuyen su información gramatical en un total de 1345 características gramaticales para ambas lenguas. De estas características gramaticales, sólo 43 (3%) han necesitado, hasta ahora, modificar sus restricciones de realización estructural asociadas para producir estructuras castellanas en lugar de inglesas. Las nuevas áreas de la gramática desarrollada para el castellano, consistentes en 21 sistemas gramaticales, supusieron la adición de 35 características gramaticales, que, previsiblemente, están en su mayoría en las áreas en las que el castellano difiere del inglés, por ejemplo la concordancia de género y número, y las formas de pasiva. Sin embargo, existen también otras zonas en donde la gramática se ha extendido en cobertura y estas extensiones se podrían aplicar también a la gramática inglesa original (por ejemplo, el tratamiento de proposiciones "relacionales" y algunos tipos de nominalización). Así mismo, las adiciones para la correspondencia entre la semántica y la gramática también son muy limitadas: se han creado 11 electores nuevos para castellano (de un total de 443). Estos resultados demuestran que se pueden construir recursos para la generación en castellano con una alta reutilización de los existentes y disponibles previamente en KPML. La mayor parte del esfuerzo se dedicó a la extensión de la gramática en áreas que no se encuentran en inglés y en la construcción de los recursos léxicos necesarios para nuestro dominio.

6.- CONCLUSIONES.

El primer resultado de nuestro trabajo ha sido la construcción de una arquitectura que es capaz de generar textos, a partir de los conocimientos almacenados en una ontología, mediante la reutilización de una ontología lingüística y otros recursos (ya usados para otros proyectos y lenguas). Nuestro enfoque tiene varias ventajas que se deberían tener en cuenta en los siguientes campos:

a) Ingeniería ontológica:
   - Evaluación de ontologías: los textos en lenguaje natural se pueden usar por el experto del dominio para evaluar tanto la ontología de dominio como el modelo conceptual. Los textos finales pueden ser también útiles para medir con rapidez la calidad y la cantidad de conocimientos representados en la ontología.
   - Ontologías de dominio como fuentes de conocimientos: El marco de trabajo de METHONTOLOGY permite construir ontologías de dominio con una estructura y organización estándares y bien definidas, como es el caso de *Chemicals*. Se ofrece así, una solución genérica para reutilizar ontologías de dominio como una fuente adecuada para los sistemas de generación de textos.

b) Generación de textos:
   - Documentación: la generación de textos se puede ver como un primer paso para construir documentación semiautomática sobre la ontología de dominio y su proceso de desarrollo.
   - Difusión o distribución de conocimientos: la generación de textos a partir de ontologías es uno de los mejores modos de que los conocimientos de un dominio estén disponibles para usuarios no expertos o poco familiarizados con las ontologías.
   - Facilitación de la recuperación de información: los usuarios pueden preguntar y recuperar distintos tipos de información en su propia lengua: definiciones de conceptos e instancias, descripciones de propiedades de conceptos e instancias, relaciones entre conceptos, comparaciones entre instancias, etc.

c) Compartición y reutilización de conocimientos:
   - Las ontologías de dominio y las ontologías lingüísticas se pueden mezclar con éxito. Nuestro sistema integra recursos heterogéneos como KPML, GUM y *Chemicals*, que ya se han utilizado en otros proyectos.
   - La posibilidad de reutilizar los componentes del prototipo actual en futuras extensiones del sistema es un beneficio claro y, en buena medida, ha sido uno de los objetivos del trabajo realizado. La modularidad de nuestro diseño nos permite trabajar posteriormente con mayor profundidad o ampliarlo a otros ámbitos.

El prototipo actual basado en *Ontogeneration* genera 100 tipos de texto como respuesta a 13 tipos de consulta. Funciona en una estación de trabajo Unix (Solaris 2.5). KPML necesita Commmon Lisp y Loom; en concreto hemos usado Liquid Common Lisp 5.0 (Lucid Compilant) y Loom 2.1. La interfaz de usuario está implementada en Java.

7.-TRABAJO FUTURO.

El trabajo futuro en *Ontogeneration* implica dos tipos de extensiones: a corto plazo y a largo plazo. Entre las de corto plazo podemos mencionar:

- Preparar una versión del sistema de consulta que pueda consultarse "on-line" vía web. También una nueva versión que funcione en PC.
- Hacer más extensiones en la ontología GUM para castellano, incluyendo subcategorías y especializaciones de categorías abstractas, y en la gramática castellana, cubriendo fenómenos lingüísticos complejos relacionados con la planificación de oraciones y la estructura del discurso.

En cuanto a las de largo plazo podemos decir que el sistema actual es sólo un punto de arranque. Su construcción intenta demostrar una idea ambiciosa: la posibilidad de ampliaciones continuas en muchas direcciones mediante la reutilización y extensión de recursos sin la necesidad de empezar de cero. Así, podemos referirnos a aspectos tales como:

- Multilingualidad. Ofrecer generación multilingüe de textos es una de las principales ventajas de KPML. Esta herramienta minimiza el esfuerzo de desarrollar y reutilizar recursos gramaticales de otras lenguas. Ya hemos hecho algunas pruebas para el inglés con éxito.
- Conocimientos y Dominios. Añadir otras ontologías de dominio distintas de *Chemicals*, pero con una estructura similar (como otras taxonomías científicas cercanas) para minimizar los cambios en la gramática, que serían casi exclusivamente léxicos. El marco de trabajo de METHONTOLOGY proporciona la base para construir ontologías estructuradas.
- Multiusuario. Extender los modelos de usuario atendiendo a todas las variaciones posibles y significativas: edad (adultos, niños), experiencia o conocimiento previo del dominio, etc.
- Generación de texto. Para construir un sistema de generación que tenga en cuenta la variedad de usuarios, registros del lenguaje y dominios, es crucial abordar lo que se conoce en inglés como *deep generation*. Su objetivo es producir especificaciones con un grado de abstracción lingüística más detallado y con mayor complejidad en la tipología textual.
- Multimodalidad. Crear una interfaz de usuario multimodal que permita diferentes modalidades de entrada/salida e interacción. Los textos finales podrían combinarse con elementos multimedia: hipertexto, gráficos, dibujos en 3D, vídeo, etc. Utilizando una interfaz de lenguaje natural, el usuario podría escribir o realizar directamente las consultas en su propia lengua.
- Aplicaciones. La generación de textos a partir de ontologías se puede reutilizar en distintas aplicaciones como tutores inteligentes, sistemas basados en el conocimiento, accesos a bases de datos, recuperación de información multilingüe, traducción automática, etc.

REFERENCIAS.

Bañon, A. (1999): *Modelo de Generación Multisentencial EPRS*. Facultad de Informática. Universidad Politécnica de Madrid.

Bateman, J. A.; Magnini, B.y Fabris G. (1995): "The Generalized Upper Model Knowledge Base: Organization and Use". *Towards Very Large Knowledge Bases*, IOS Press, 60-72.

Bateman, J. A. y Teich, E. (1995). "Selective Information presentation in an Integrated Publication System: an Application of Genre-Driven Text Generation". *Information Processing and Management*, Elsevier Science Ltd., 31(5) 753-768.

Bernardos, S. (1997): *GUME: Extensión de la Ontología GUM para el Español*. Facultad de Informática. Universidad Politécnica de Madrid.

Blázquez, M.;. Fernández, M; García-Pinar, J. M. y Gómez-Pérez, A. (1998). "Building Ontologies at the Knowledge Level using the Ontology Desing Environment". KAW'98. Banf (Canada).

Fernández, M. (1996): *Chemicals: Una Ontología de Elementos Químicos*. Facultad de Informática. Universidad Politécnica de Madrid.

Fernández, M.; Gómez-Pérez, A. y Juristo, N. (1997). "METHONTOLOGY: From Ontological Art Towards Ontological Engineering", *AAAI-97 Spring Symposium Series on Ontological Engineering*, Standford University, CA (USA), 33-40.

Fernández, M.; Gómez-Pérez, A.; Pazos, A. y Pazos, J. (1999): Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment. *IEEE. Intelligent Systems and their applications.* January/February 1999, 37-46.

Fischer, D. y Rostek, L. (1993): "SFK: A Smalltalk Frame Kit", *Technical Report, GMD/IPSI*, Darmstadt (Germany).

Gómez-Pérez, A. y Rojas, M. D. (1999): "Ontological Reengineering for Reuse". *Proceedings of the Knowledge Acquisition, Modelling and Management*, 11th EKAW Conference, 1999, 139-156.

Gómez-Pérez, A. (1998): "Knowledge Sharing and Reuse". *Handbook of Applied Expert Systems*, edited by Liebowitz, CRC.

Gruber, T.R. (1993a): "Towards Principles for the Desing of Ontologies Used for Knowledge Sharing". *Workshop on Formal Ontologies*, Padua (Italia).

Gruber, T.R. (1993b): "Ontolingua: A Translation Approach to Portable Ontology Specifications". *Knowledge Acquisition*. Vol. 5, 199-220.

Halliday, M. A. K. (1985): *An Introduction to Functional Grammar.* Edward Arnold, London (UK).

Kasper, R. T. (1989): "A flexible interface for linking applications to PENMAN's sentence generator", *Proceedings of the DARPA Workshop on Speech and Natural Language*.

Lavoie, B.; Rambow, O. y Reiter, E. (1996): "The ModelExplainer". En *Demonstration Notes of the International Language Generation Workshop* (INLG-96), Herstmonceux Castle, Sussex (Reino Unido).

McKeown K.; Kukich K. y Shaw J. (1994). "Practical Issues in Automatic Document Generation". En *Proceedings of the Fourth Conference on Applied Natural-Language Processing* (APLN-1994), 7-14, 1994.

C. M. I. M. Matthiessen y J. Bateman (1991): *Text Generation and Systemic-Functional Linguistics: Experiences from English and Japanese*. Pinter Publishers, London (Reino Unido).

Nieto, E. (1999): *Metodología y Desarrollo de una Gramática Sistémica-Funcional para la Generación en Español*. Facultad de Informática. Universidad Politécnica de Madrid.

Newell, A. (1982): "The Knowledge Level". *Artificial Intelligence*. 18, 87-127.

Rösner, D. (1994): "Generating Multilingual Documents from a Knowledge Base: The TECHDOC Project". *Technical Report* FAW Ulm, Ulm (Alemania).

Uschold, M. y Gruninger, M. (1996): "ONTOLOGIES: Principles, Methods and Applications". *Knowledge Engineering Review*. 11(2).