

Agentes y ontologías para el tratamiento de información: clasificación y recuperación en Internet.

Dr. Jesús Tramullas
Dep. de CC. de la Documentación
Univ. de Zaragoza
tramullas@posta.unizar.es
<http://piramide.unizar.es>

Resumen:

Se plantean los principios y fundamentos de las ontologías en el campo de los agentes de software para recuperación de información en Internet. Se revisan varios proyectos, y se propone una arquitectura para recuperación de información en Internet, basada en la utilización de ontologías.

Palabras clave:

Agentes de software, ontologías, recuperación de información en Internet

Abstract:

This paper revises the principles and fundaments of ontologies for Internet information retrieval softbots. Some projects are analized, and is proposed a teorical architecture for information retrieval on Internet, using ontologies.

Keywords:

Software agents, ontology, Internet information retrieval.

Los agentes de software.

Los agentes de software, también llamados *softbots*,¹ son una de las áreas clave de desarrollo de la ciencias de los computadoras para los próximos años. En el entorno anglosajón estas tecnologías suelen recibir la denominación de “*killer apps*”. Sin ninguna duda, puede afirmarse que los *softbots* son el interés prioritario de los especialistas en organización de información, por las perspectivas que abren en lo que se refiere a localización, identificación, relación, mantenimiento y selección de recursos de información.

Sin embargo, el primer problema con el que se encuentran los investigadores es que no existe un concepto único y diáfano de lo que es o no es un *softbot*. En un reciente trabajo, Bradshaw² ha señalado en que en numerosas ocasiones la definición de lo que es un *softbot* depende del punto de vista del investigador, o de la descripción que se hace del agente de software según sus atributos, sin que esos mismos atributos coincidan en todos los autores. En la misma línea, Nwana ha señalado que el concepto de *softbot* puede ser ya rastreado en la investigación en Inteligencia Artificial desarrollada en la década de 1970, pero que continúa siendo un término difuso, un meta-término o paraguas que da cobertura a diferentes enfoques³. Precisamente esta ambigüedad en la definición y concepto de agente ha tenido como consecuencia la aparición de intentos de síntesis, entre los que debe destacarse el trabajo de Franklin y Graesser⁴. Estos autores revisan un elevado número de definiciones dadas por otros investigadores, para concluir en una definición: “An autonomus agent is a system situated within and a part of an environment that senses environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future.” Debe destacarse que en este

difuso contexto, gran parte de las definiciones toman como referencia precisamente los atributos que se consideran inherentes al *softbot* (véase el resumen realizado por Bradshaw, 1997). Como consecuencia de esta definición, proponen una taxonomía o clasificación de agentes, intento que también ha sido abordado por otros autores.

La complejidad que rodea a todo al ámbito de los *softbots*, en la que intervienen Inteligencia Artificial, Sociología, Comunicación, Ciencia de la Computación, Lógica, Telecomunicaciones y otras disciplinas, ha llevado a proponer la creación de una nueva disciplina, la *Agent-Based Software Engineering*.⁵ El fundamento de la misma es solucionar el problema que supone la heterogeneidad de las situaciones en las que puede encontrarse un *softbot*, y la solución a este problema se basa en la comunicación y en la cooperación.

Cooperación y comunicación entre *softbots*.

Las tareas que se encargan a los *softbots* requieren forzosamente interactuar con su entorno. Sin embargo, el progresivo desarrollo de la complejidad de esos mismos entornos, así como la heterogeneidad en los mismos, que apuntábamos en párrafos anteriores, han impuesto dos necesidades, correspondientes a las habilidades de los *softbots*, como son comunicar y cooperar entre ellos. Evidentemente, la comunicación y cooperación presupone la existencia de un lenguaje de comunicación que pueda ser entendido por todos ellos, con independencia de su origen o misión. Como cualquier otro lenguaje, deberá tener una sintaxis y una semántica, y ser capaz de reflejar estructuras de conocimiento. Genesereth y Ketchpel (1994) han propuesto la creación de lo que han llamado Agent Communication Language (ACL), compuesto de tres partes: un vocabulario, un lenguaje interno KIF (Knowledge Interchange Format) y un lenguaje externo KQML (Knowledge Query and Management Language). Un mensaje en ACL sería una expresión en KQML, en la cual los argumentos serían términos o frases en KIF, formadas usando los términos incluidos en el vocabulario.

KIF y KQML son el resultado de las investigaciones desarrolladas en el marco del proyecto DARPA Knowledge Sharing Effort (KSE). Ha sido definido como un lenguaje de cálculo de predicados de primer orden, con extensiones propias para resaltar la expresividad. Con este enfoque ofrece medios para incluir en sus expresiones datos simples, constricciones, disyunciones, reglas, negativas, metainformación, etc. Sin embargo, en su diseño es independiente del contexto, por lo que cada mensaje debería incluir información implícita sobre el emisor, el receptor, el camino o vía, la historia de lo comunicado, etc. Para aumentar la eficiencia del sistema de comunicación, la misión de KQML es actuar como una capa lingüística que se encarga de tomar en consideración el contexto en el que se produce la comunicación, y liberar a KIF de trabajar con la información contextual.

Sin embargo, la utilización de KIF/KQML no es la única aproximación al problema que plantea el lenguaje de comunicación entre *softbots*. Existen otros lenguajes, como TeleScript⁶, AgentTCL, AOP, etc., que también son utilizados en la construcción de *softbots*.

Las ontologías.

El primer elemento de los apuntados por Genesereth y Ketchpel (1994) es el vocabulario. Estos autores entienden el vocabulario como un diccionario de palabras que poseen un significado claro para el ser humano. En un diccionario de este tipo puede haber una o varias ontologías. Por lo tanto, el contenido semántico, el conocimiento que poseen e intercambian los agentes de software se encuentra formalizado en estructuras cognitivas que adoptan la forma de ontologías. Ante la importancia que adquieren entonces las ontologías que utilizan los *softbots*, cabe plantearse varias cuestiones cruciales: ¿Qué es una ontología en el contexto de los *softbots*? ¿Utilizan las ontologías los mismos principios lógicos de diseño y organización? ¿Son comunes las ontologías al conjunto de los agentes, o son particulares a cada caso? ¿Existen repositorios o servidores de ontologías?.

La definición de partida más aceptada es "An ontology is an explicit specification of a conceptualization."⁷ Profundizando en ello, se trata, por lo tanto, de una especificación explícita

y formal, que sigue unas reglas lógicas y semánticas, y que se supone contiene el significado informativo y las relaciones presentes en una conceptualización. Conceptualización que, por otra parte, se supone corresponde a una parte del mundo o universo que es objeto de tratamiento⁸. Consecuentemente, una ontología es el resultado de seleccionar un dominio, y aplicar sobre el mismo un método para obtener una representación formal de los conceptos que contiene y las relaciones que existen entre los mismos. Sin embargo, Guarino⁹ ha argumentado razonadamente que esa conceptualización debe verse como un conjunto de reglas informales que, sobre un aspecto de la realidad, un *softbot* usa para aislar y organizar objetos y relaciones relevantes, independientemente de la estructura que ofrezcan los mismos en una situación dada. Se deduce que esta aproximación, a pesar de añadir un elemento de complejidad, resulta de mayor interés y profundidad para la construcción de ontologías. El mismo Guarino (1996) desarrolla un análisis de las diferentes definiciones de ontología, prestando especial interés a aquellas que incorporan la noción de meta-nivel, y actúan como “meta-modelos representacionales”.

El párrafo anterior anticipa que no existe una única aproximación a la lógica que rige la construcción de ontologías, sino que éstas dependen, en gran manera, del contexto en el que se construyen. Independientemente del contexto, Valente y Breuker (1996) han establecido un conjunto de principios que deben respetar las ontologías, especialmente pensados para ontologías nucleares que van a ser compartidas, y que han sido tratadas en profundidad por van Heijst *et alii*¹⁰. La aplicación de principios comunes en la construcción de ontologías será lo que permitirá la utilización compartida de ontologías por agentes distintos y ajenos entre sí. El conocimiento almacenado en una ontología, sobre un dominio dado, podrá combinarse con el contenido en otras, formando un biblioteca de ontologías. La teoría no corresponde con la realidad: los problemas derivados de la posible utilización de bibliotecas de ontologías, a nivel general, comienzan por la aprehensión del significado de los términos incluidos en las mismas, así como el valor que adquieren dependiendo del nivel en el que se encuentran, o las relaciones en las que se incluyen. Estas cuestiones limitan seriamente, al menos por el momento, la utilización de ontologías en los *softbot*. A pesar de ello, las ontologías pueden desempeñar varios papeles o roles¹¹ bajo el concepto unificador de “knowledge sharing”¹²:

- 1.- Como repositorios para la organización de conocimientos e información, tanto de tipo corporativo como científico.
- 2.- Como herramienta para la adquisición de información, en situaciones en la que un equipo de trabajo la utiliza como soporte común para la organización del dominio.
- 3.- Como herramienta de referencia en la construcción de sistemas basados en el conocimiento, ya que la utilización consistente de los términos que supone es básica en la ingeniería del conocimiento.
- 4.- Para permitir la reutilización del conocimiento ya existente, en la creación de nuevas aplicaciones.
- 5.- Como base para la construcción de lenguajes de representación del conocimiento, acompañada de la formalización del cálculo que tenga lugar entre los términos.

Gran parte de estas ideas están presentes en el servidor Ontolingua¹³, un repositorio de ontologías, creado y desarrollado en el marco del KSE ya citado, y disponible en <http://ontolingua.stanford.edu>. El servidor Ontolingua ofrece los mecanismos para crear una ontología, o utilizar alguna de las ya construidas, así como las herramientas necesarias para usarla en conjunción con lenguajes como KIF, etc., También se ha incluido un API para poder integrar las ontologías del servidor con agentes preparados para Internet. La importancia que están alcanzando las ontologías se demuestra por el auge que están alcanzando los aspectos relacionados con ellas, desde metodologías a herramientas, llegando incluso a formar una subdisciplina en el campo de la Inteligencia Artificial y la Ingeniería del Conocimiento¹⁴.

Experiencias con ontologías en Internet.

El potencial de las ontologías en todos los campos de aplicación de los agentes de software es evidente, por lo que el descubrimiento de información en Internet, con los problemas que plantea, no podía ser una excepción¹⁵. Han sido numerosos los proyectos y experiencias que se han desarrollado en Internet, utilizando ontologías, o lo están siendo en

este momento. Es obligado comenzar una breve revisión recordando el ya citado servidor Ontolingua, fruto del KSE, que ofrece las herramientas necesarias para crear ontologías, integrarlas con otras ya existentes, e incorporarlas en nuevos productos de software. Otro enfoque diferente es el aportado por Luke, Spector y Rager (1996)¹⁶, que han desarrollado SHOE (*Simple HTML Ontology Extensions*), un complemento semántico al HTML, el cual refleja el contenido de la página web, y que puede ser utilizado por agentes para el descubrimiento de información. Posteriormente, SHOE ha evolucionado hacia RDF y CG, siendo la más reciente la especificación OML (*Ontology Markup Language*)¹⁷, mantenida por Robert E. Kent, y que se apoya en la misma filosofía que SHOE.

Uno de los proyectos más conocidos es el llevado a cabo por Gerry M. Kiermann, con el nombre de CyberStacks¹⁸. Se pretende ofrecer al usuario un listado de recursos de información en Internet, organizados según una clasificación, subclases y descripciones particulares. Sin embargo, por el momento no ha integrado completamente ontologías en su servidor, y tampoco dispone de agentes capaces de aprovechar esas ontologías.

La utilización de ontologías también está presente en el proyecto *FERMI (Formalisation and Experimentation on the Retrieval of Multimedia Information)*, proyecto ESPRIT 8134¹⁹, bajo la dirección K. van Rijsbergen, en el que se incluyen herramientas de planificación, descubrimiento y selección de recursos de información multimedia. *Information Manifold Project*²⁰, desarrollado en el ámbito de Bell Labs, hace uso de las ontologías para identificar las fuentes de información pertinentes a una búsqueda, acceder a las mismas, obtener documentos relevantes, compararlos, seleccionar los más adecuados y ofrece un resumen previo al usuario. En el ámbito de la medicina, cuya necesidad de normalización y rigor semántico es crucial, debe destacarse el proyecto UMLS (*Unified Medical Language System*) desarrollado por la National Library of Medicine²¹, que utiliza las ontologías como una herramienta más para la recuperación y acceso a la información biomédica.

Otros proyectos no hacen mención específica a la utilización de ontologías, pero de la lectura de las publicaciones que los detallan²² se deduce en todo momento la presencia una estructura de organización y representación del dominio objeto de trabajo, que utilizan los softbots. Esta estructura puede adoptar formas muy diversas, desde una estructura de base de datos según el modelo E/R, a un sistema basado en el conocimiento (KBS). Domingue²³ ha señalado cómo las interfaces en HTML no reúnen las prestaciones necesarias para una correcta interacción con las ontologías, y ha diseñado un servidor y un cliente en Java para las mismas. Con este enfoque ofrece un entorno de creación de ontologías, que puede utilizarse para la clasificación y recuperación de información, pero que demanda la construcción de una aplicación específica.

Una propuesta de arquitectura para descubrimiento de información basada en ontologías.

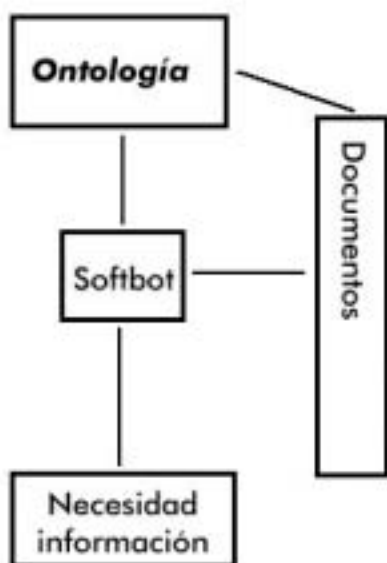
En los siguientes párrafos proponemos una arquitectura ideal para un sistema de clasificación y recuperación de información en Internet, tomando en consideración las ideas expuestas anteriormente. Para ello se toman como punto de partida dos principios básicos:

- 1.- El contenido de los documentos debe representarse usando como herramienta una ontología.
- 2.- Las tareas de descubrimiento y selección de recursos de información deben realizarse mediante softbots.

El primero de ellos está adaptado tomando la idea del actual OML. En principio, supone que la representación de un documento responde a una norma fijada por un principio de autoridad. Sin embargo, en nuestra propuesta se deja la posibilidad de que la ontología resida en un servidor, o sea una ontología propia, siempre y cuando se especifique su localización en el documento. El segundo de ellos aprovecha las prestaciones de los agentes de IR en Internet para liberar al usuario de las tareas de descubrimiento, acceso y selección de recursos de información, que realiza en un segundo plano.

En principio, la existencia de la ontología de referencia es previa a la creación del documento, mientras que el proceso de descubrimiento utiliza la ontología de referencia como criterio para validar el contenido informativo del documento. En cualquier caso, las ontologías utilizadas para ello deben ser públicas y accesibles, lo que permitirá el trabajo cooperativo, y la integración de ontologías hasta poder llegar a formar meta-ontologías. Si la integración de ontologías no parece ofrecer mayores problemas (considerando que queda a la elección del creador del documento el mecanismo y la ontología que desee), mayores cuestiones plantea el proceso de descubrimiento. En este caso, debemos diferenciar si el ámbito de descubrimiento es homogéneo o heterogéneo. Por homogéneo entendemos la consistencia en la utilización de ontologías, es decir, que el usuario ha usado ontologías conocidas a otros usuarios, en un contexto informático-espacial determinado. Este podría ser el caso, por ejemplo, de una intranet. En un entorno heterogéneo, como es ahora Internet, el proceso se complica. En primer lugar, no se ha extendido todavía la utilización de ontologías, y la aproximación más cercana, los metadatos, no ha alcanzado la extensión deseable. En segundo lugar, consideramos imprescindible la presencia de un punto de partida, de un punto crítico, para iniciar el proceso de descubrimiento. Esta es la cuestión más complicada, porque supone el recurso a la consulta contra un motor de búsqueda, o la utilización de índices temáticos específicos, con los problemas y las limitaciones que ello puede suponer.

Caso A: entorno homogéneo



Caso b: entorno heterogéneo

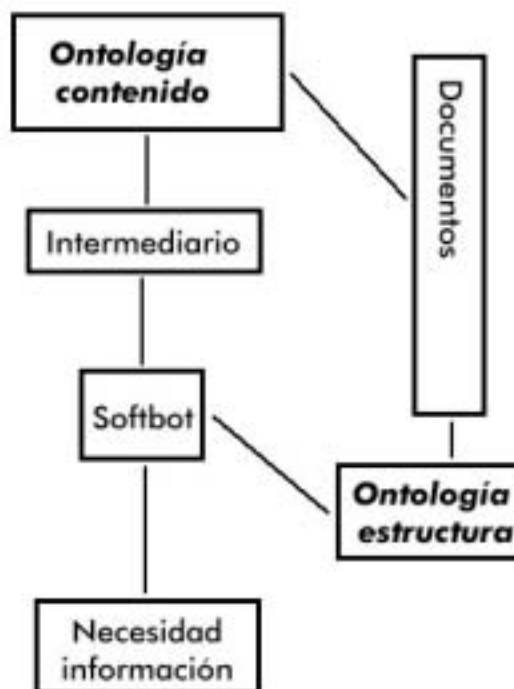


Fig.1. Esquema básico de la arquitectura propuesta

En este segundo caso, el trabajo inicial de consulta sería desarrollado por un agente, el cual validaría los enlaces obtenidos de esta forma, comprobando su localización y contenido, usando para ello las referencias a las ontologías usadas en los propios documentos. En esta aproximación no se realizaría un análisis textual de contenidos, sino sólo de la representación obtenida desde la ontología. La obtención de una copia del documento original, o sólo la comprobación de su existencia, queda al arbitrio del agente.

Esta arquitectura teórica esconde, sin embargo, un problema: los documentos existentes en Internet no son homogéneos, ni en forma, ni en estructura física y lógica. Por lo tanto, se hace necesario incorporar al sistema una ontología exclusivamente dedicada al tipo de documento. Este enfoque nos permitiría subsanar el problema que plantean los documentos

multifacetados, los documentos sonoros o gráficos, o los documentos dinámicos. La arquitectura funcionará a pleno rendimiento cuando se integren tanto los aspectos físicos del documento, como los aspectos informativo-documentales del mismo.

¹ Woolridge, M. y Jennings, N.R., "Intelligent Agents: Theory and Practice." *Knowledge Engineering Review*, 10(2), 1995, pp. 115-152; Jennings, N.R. y Woolridge, M., "Applications of Intelligent Agents." En: Jennings, N.R. y Woolridge, M., (eds.) *Agent Technology: Foundations, Applications and Markets*. Springer-Verlag, 1998, pp. 3-27.

² Bradshaw, J., "An Introduction to Software Agentes." En: Bradshaw, J., (ed.) *Software Agents*, AAAI Press / MIT Press, 1997, pp. 4-7. Este volumen recoge estudios de gran interés, que cubren casi todas las áreas abordadas en este trabajo.

³ Nwana, J., "Software Agents: An Overview." *Knowledge Engineering Review*, 11(3), 1996, pp. 205-244.

⁴ Franklin, S. y Graesser, R., "It is an Agent, or just a Program?: A Taxonomy for Autonomous Agents." *Third International Workshop on Agent Theories, Architectures, and Languages*. Springer-Verlag, 1996, pp. 193-206.

⁵ Genesereth, M.R. y Ketchpel, S.P., "Software Agents." *Communications of the ACM*, 37(7), 1994, pp. 48-53.

⁶ TeleScript es un producto de General Magic, <http://www.genmagic.com> (Mountain View, CA).

⁷ Gruber, T.R., "Toward Principles for the Design of Ontologies Used for Knowledge Sharing." *International Journal of Human and Computer Studies*, 43(5-6), 1995, pp. 907-928 (también disponible como *Technical Report KSL 93-04*, Knowledge Systems Laboratory, Stanford University, 1993).

⁸ Valente, A. y Breuker, J., "Towards Principled Core Antologies." *Knowledge Acquisition Workshop 1996*. <http://ksi.cpsc.ucalgary.ca/KAW/KAW96/valente/doc.html> (consultado 2-1-1999)

⁹ Guarino, N., "Understanding, Building, and Using Ontologies." *Knowledge Acquisition Workshop 1996*. <http://ksi.cpsc.ucalgary.ca/KAW/KAW96/guarino/guarino.html> (consultado el 15-12-1998); "Formal Ontology, Conceptual Analysis and Knowledge Representation." *International Journal of Human and Computer Studies*, 43(5-6) 1995, pp. 625-640.

¹⁰ Van Heijst, G., Schreiber, A.T. y Wielinga, B.J., "Using Explicit Ontologies in KBS Development." *International Journal of Human and Computer Studies*, 1996.

¹¹ Valente y Breuker, *op.cit.*, 1996.

¹² Gruber, *op.cit.*

¹³ Farquhart, A., Fikes, R. y Rice, J., *The Ontolingua Server: A Tool for Collaborative Ontology Construction*. 1996. ftp://ftp-ksl.stanford.edu/pub/KSL_Reports/KSL-96-26.ps (consultado 24-11-1998)

¹⁴ GÓMEZ-PÉREZ, A., *What is Ontological Engineering?*. Madrid: UPM, 1998.

¹⁵ HERMANS, B., *Intelligent Software Agents on the Internet: An Inventory of Currently Offered Functionality in the Information Society and a Prediction of (Near) Future Developments*. Tilburg: Tilburg University, 1996. <http://www.hermans.org/agents/index.html> (consultado 3-6-1998)

¹⁶ LUKE, S., SPECTOR, L. y RAGER, D., "Ontology-Based Knowledge Discovery on the World-Wide Web". *Proceedings of the Workshop on Internet-based Information Systems, AAAI-96*, 1996. <http://www.cs.umd.edu/projects/plus/SOE/aaai-paper.html> (consultado 5-1-1999).

¹⁷ Disponible en <http://wave.eecs.wsu.edu/CKRMI/OML.html>

¹⁸ Disponible en <http://www.public.iastate.edu/~CYBERSTACKS/>

¹⁹ Disponible en <http://www.dcs.gla.ac.uk/fermi/>

²⁰ Disponible en <http://www.research.att.com/~levy/imhome.html>

²¹ Disponible en <http://www.nlm.nih.gov/research/umls/>

²² Por ejemplo, Birmingham, W.P. "An Agent-Based Architecture for Digital Libraries". *D-Lib Magazine*, Julio 1995. <http://www.dlib.org/July95/07brimingham.html>; Salampasis, M., Tait, J. Y Hardy, C., "An Agent-Based Hypermedia Model for Digital libraries". *Advanced Digital Libraries 1996*. <http://osiris.sund.ac.uk/~cs0mas/digital.htm> (consultado 2-4-1998).

²³ DOMINGUE, J., "Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web". *Knowledge Acquisition Workshop 1998*. Banff (Canadá). <http://ksi.cpsc.ucalgary.ca/KAW/KAW98/domingue/> (consultado 23-1-1999).