

Hacia la Definición de Ontologías Orientadas a Aspectos

María del Pilar Romay¹ y Carlos E. Cuesta²

¹ Departamento de Sistemas Informáticos
Escuela Superior Politécnica, Universidad Europea de Madrid
pilar.romay@gmail.com

² Departamento de Informática (Arquitectura, C. Computación y Lenguajes)*
Escuela Técnica Superior de Ingeniería Informática, Universidad de Valladolid
cecuesta@wwisa.org

Resumen En este trabajo se propone la adopción de una estrategia orientada a aspectos para la definición de ontologías durante el desarrollo de sistemas de software. Tras exponer brevemente el ámbito e interés de la noción de ontología, se comenta el tradicional problema derivado de la combinación de ontologías diferentes. Se sugiere que el uso metodológico de un enfoque orientado a aspectos, que considere su combinación de manera sistemática, tendría además como consecuencia una resolución natural de estos conflictos. Para plantear esta combinación, se propone el uso de un marco categórico basado en la noción de *superposición*, procedente del ámbito de concurrencia, que permita expresar las relaciones entre los distintos aspectos. Se sugiere, asimismo, cómo la noción lógica de *superposición*, totalmente independiente, podría ser utilizada para aplicar estos mecanismos sobre una base de Lógicas de Descripción. Se concluye señalando algunas de las posibles aplicaciones de este tipo de sistemas.

1. Introducción

Hoy en día, las aplicaciones informáticas se ven obligadas a manejar conceptos e informaciones muy diversas, procedentes de distintos dominios, puntos de vista y áreas de conocimiento. Para permitir a estas aplicaciones dialogar entre sí, así como para conciliar las diferentes perspectivas de los distintos subsistemas que constituyen una aplicación compleja, se hace necesario poder definir e intercambiar información relativa a la base conceptual usada por cada una de estas aplicaciones y subsistemas. Las *ontologías* son uno de los mecanismos más expresivos y conocidos, tanto para la definición como para la manipulación de estas bases conceptuales. Su origen se ha ligado tradicionalmente a los sistemas de representación del conocimiento dentro de la Inteligencia Artificial; pero tanto su uso como su popularidad se han extendido de manera notable en los últimos años, especialmente desde su aplicación en la Web Semántica, que se constituye de hecho como un gran sistema basado en conocimiento.

El diseño de una ontología adecuada es una tarea muy compleja, que resulta crítica en la definición de grandes sistemas. También es uno de los primeros detalles que ha

* Este trabajo ha sido parcialmente financiado por el Ministerio de Educación y Ciencia dentro del Proyecto Nacional MCYT-TIC2003-07804-C05-01 (DYNAMICA-PRISMA), y por la Consejería de Educación de la Junta de Castilla y León dentro del Proyecto JCYL-VA117/03.

de ser contemplado en la definición y desarrollo de un sistema de este tipo, ya que se relaciona directamente con el estudio de los dominios implicados, en el sentido más amplio de la palabra. Por este motivo, se podría considerar a los aspectos ontológicos como pertenecientes al grupo de los denominados *aspectos tempranos* (*early aspects*), junto a los arquitectónicos y los relativos a requisitos.

Aunque la expresión *Desarrollo de Software Orientado a Aspectos* es genérica, la creciente popularidad de los entornos de programación orientada a aspectos, en particular AspectJ, proporcionan una connotación del término *aspecto* que lo concibe como un módulo asimétrico a nivel de implementación. Sin embargo, a lo largo de este trabajo se utiliza este término en su sentido más general, esto es, como encarnación del principio de separación que sugiere tratar cada área de interés activo –cada *aspecto*³ del sistema– de manera totalmente independiente. En particular, se asume que los aspectos son *simétricos* [6], esto es, que no se conciben como extensiones de un aspecto “principal”, sino como perspectivas independientes de un mismo sistema, que se encargan de describir diferentes puntos de vista o contenidos referidos a distintos dominios.

Cabe plantearse si la definición de estos aspectos ontológicos puede ser considerada también en los mismos términos, es decir, “orientada a aspectos”. En un sistema complejo como los descritos, en el que cada subsistema maneja su propia ontología y hay múltiples aspectos involucrados, se puede utilizar un enfoque totalmente distinto. En lugar de intentar la definición de una ontología global, compleja y difícil de mantener, se concibe que cada aspecto, dominio y subsistema puede tener su propia ontología parcial, de modo que éstas sean combinadas posteriormente.

Nuestra propuesta consiste en plantear una metodología en la que la elaboración de una ontología se hace por partes, en lugar de realizar un diseño global. Cada una de estas partes viene determinada por un aspecto distinto. Para cada uno de estos aspectos, se define una ontología independiente, que permita describir el conocimiento que se tiene sobre un aspecto determinado. Los conceptos que se presentan para caracterizar este aspecto deben centrarse sólo en las propiedades intrínsecas del mismo, aislándolo de los entornos en los que se podría incorporar. Por ejemplo, una ontología de seguridad solamente haría referencia a cuestiones de seguridad, sin considerar su aplicación.

2. El Papel de las Ontologías

Como es bien conocido, el término *Ontología* procede de la Filosofía, donde designa a la parte de la Metafísica que se ocupa de la teoría del *ser*, es decir, de los entes y sus relaciones. Desde esta perspectiva, se concibe la realidad como estructurada en un sistema de categorías. El uso del término en Informática procede de la evolución de este último aspecto, asimilándose como equivalente a un *corpus* estructurado de conceptos interrelacionados, algo mucho más amplio que un simple vocabulario.

La utilización de sistemas de representación ontológica a través de una descripción terminológica ha dado lugar a diferentes tipos de lenguajes. Estos lenguajes han

³ En este documento, la expresión “orientado a aspectos” ha de entenderse como equivalente a la expresión “*concern-oriented*”, y en consecuencia el término *aspecto* se utiliza como una traducción del término *concern*, definido como área de interés activo. Aunque esto no es cierto en el caso más general, se ha decidido hacerlo así para mayor simplicidad y claridad.

evolucionado, a partir de KL-ONE, hasta llegar a lenguajes basados en Lógicas de Descripción, con una base formal sólida, y que se utilizan incluso para la representación de conocimiento dentro de la Web Semántica [11]. En este ámbito heterogéneo, los sistemas de representación deben permitir trabajar de manera distribuida, y contemplar la posibilidad de trabajar con diferentes tipos de ontologías.

Las Lógicas de Descripción [1] son un tipo de sistema formal muy potente, derivado a partir de la lógica de primer orden sin funciones; son la gran apuesta actual para la representación de ontologías, y han dado lugar a lenguajes de distinto tipo y de origen diverso, tales como FaCT [7] o RACER [5], entre otros [11].

En las Lógicas de Descripción, la base de conocimiento está formada por dos conjuntos de axiomas, que se definen como el componente intensional (TBox) y el extensional (ABox). El primero está formado, básicamente, por definiciones y especificaciones de conceptos, además de sus relaciones de inclusión e igualdad. El segundo conjunto, aún más complejo, contiene información de tipo relacional, desarrollando la red de conceptos que proporciona la verdadera expresividad de una ontología. Los axiomas ABox indican la pertenencia de un objeto a un determinado concepto, así como la participación de pares de objetos en un rol (una relación).

El problema de la combinación y conciliación de las diversas ontologías presentes en un sistema es un problema que se ha hecho patente casi desde el principio. Existen dos enfoques extremos: el primero asume que se tiene una única “gran ontología”, y que todos los elementos del sistema han de utilizar únicamente ésta; el segundo asume que existen muchas “pequeñas ontologías”, de modo que cada aplicación o fragmento tiene la suya propia, y no se define relación alguna entre ellas.

Ninguna de las dos es totalmente adecuada. Esto deriva, necesariamente, en estrategias mixtas. Todas ellas parten de la definición de “pequeñas” ontologías que se relacionan. Las dos más significativas son la definición de una “ontología superior” y la introducción de un *mediador*. En el primer caso, se asume una gran ontología común, con la que se relacionan todas las demás; las pequeñas ontologías particulares son fragmentos de esta gran ontología. En el segundo caso, las ontologías permanecen separadas, pero se define algún tipo de elemento intermedio, que gestiona las dependencias y correspondencias entre ellas.

En ambos casos se trata de enfoques *a posteriori*, que tratan la cuestión como un problema a resolver, y resultan notablemente complejos a medida que el tamaño del sistema crece. Nuestra propuesta parte de un planteamiento *a priori*, en la que el proceso de combinación de ontologías se considera como una tarea más en la metodología de desarrollo del software, y en particular en el diseño de ontologías.

3. Ontologías Orientadas a Aspectos

La utilización de ontologías para la representación de información obtenida en los procesos de desarrollo de software permite establecer un lugar común para la comunicación entre las diferentes aplicaciones, herramientas y entornos que participan dentro del proceso de construcción. La gran cantidad de términos implicados en el proceso de desarrollo de software hace imposible trabajar con una única ontología. Se hace por tanto necesario la definición de un conjunto de ontologías que permitan centrar el co-

nocimiento en un determinado aspecto, y que puedan fusionarse para ampliar la visión global, reuniendo varios aspectos diferentes.

La solución que se propone es la construcción de un sistema ontológico estructurado a través de la separación de los diferentes aspectos. Concretamente, la propuesta consiste en definir una ontología para cada área de interés activo, con el objetivo de construir el sistema ontológico a través de fusión de ontologías.

El considerar a los aspectos como la estructura base de construcción del sistema ontológico en el proceso de desarrollo de software va a permitir trabajar a diferentes niveles de refinamiento y con diferentes vistas, en base a los aspectos deseados y las relaciones entre ellos. Así, la construcción de una ontología “completa” para el desarrollo software se realiza por la composición de diferentes *partes*, que se consideran asignadas a aspectos determinados. La construcción de la ontología se soporta por una estructura a nivel abstracto basada en la fusión (*merge*) de ontologías con base en un mecanismo de superposición, que se describe en el apartado siguiente. Las propiedades necesarias para llevar a cabo la fusión ontológica van a estar relacionadas con las relaciones de superposición y subsunción, así como las vinculadas con los términos y axiomas considerados en el proceso de construcción.

4. Combinación por Superposición

La operación de *superposición* fue concebida originalmente dentro del contexto de la Teoría de la Concurrencia, como una relación de *refinamiento*, que extendía la aplicación de este concepto al ámbito de los sistemas reactivos. La superposición se concibe simultáneamente como una noción de *extensión* y de *combinación*. Expresa, por una parte, cómo una especificación se refina en otra más elaborada, extendiendo su comportamiento o proporcionando más información. Por otra parte, expresa también cómo la especificación de los nuevos detalles se combina con la anterior para producir la versión refinada. Otros trabajos posteriores parten de esta misma idea para definir la superposición como una noción general de *composición*, más amplia que la convencional. Esto da lugar a la definición de una serie de variantes de la noción de superposición, que no son mutuamente equivalentes [4].

Recientemente, la noción de superposición ha sido propuesta como un posible fundamento formal para los modelos de orientación a aspectos, en particular desde la perspectiva de los aspectos *simétricos* [10]. En estos casos, la superposición se utiliza para hacer una fusión (*merge*) de modelos. Esto resulta especialmente evidente en los trabajos que exponen este concepto en términos de Teoría de Categorías [4].

En el trabajo de Fiadeiro y Maibaum [4], el sistema se describe como un grafo, en el que el trasfondo relacional de las transformaciones se expresa como un homomorfismo entre unidades de diseño; de este modo, la operación de composición se obtiene como una construcción estándar, concretamente el colímite de este grafo. La superposición es el tipo más habitual de morfismo en este grafo.

En general, toda ontología es susceptible de ser expresada como un grafo; y, por tanto, de ser manipulada mediante técnicas categóricas. En este punto se plantea el uso de la superposición como una operación, esto es, un morfismo, que relaciona entre sí dos ontologías, esto es, dos grafos separados. La relación permite superponer dos nodos

de manera total o parcial, dando lugar a la combinación de los dos grafos originales. En un enfoque categórico, la propia operación constituye un grafo: la relación que expresa cómo se evoluciona desde los operandos hasta el resultado es la propia superposición, y en este sentido es una relación de *refinamiento*; en conjunto, sin embargo, el grafo define una relación de *composición*.

Este planteamiento permitiría incluso, a medio plazo, la definición de estructuras de superposición, definidas como un grafo genérico en los que los nodos pueden ser cualquier concepto de una ontología. Esta estructura, inspirada en los trabajos de Katz [9], implicaría que una determinada fusión de ontologías, incluso parcial, sólo se daría cuando se identificasen todos los conceptos necesarios, y todos estos conceptos pudieran ser superpuestos de manera simultánea. Este concepto es análogo al de los *patrones de composición*, definidos por Siobhán Clarke [3] con un objetivo similar: definir las combinaciones de aspectos (simétricos), en su caso a nivel de diseño.

El problema se plantea a la hora de trasladar estas nociones a los sistemas de razonamiento automático basados en Lógicas de Descripción, en las que se fundamentan los sistemas ontológicos más evolucionados. Tratar con varias ontologías, cuyas bases de conocimiento están separadas, exige que todo proceso de deducción realice una búsqueda a través de las múltiples relaciones definidas en varios axiomas ABox. Como se ha indicado, esto tiene una gran complejidad; por otra parte, la superposición es un nuevo tipo de relación que también ha de ser definido.

La implementación de este concepto en términos de deducción tiene aún que ser determinada; sin embargo, parece haber una excelente posibilidad, derivada de algunos desarrollos recientes. En este caso, la integración de modelos implica el uso de otra noción de *superposición* [2], en principio no relacionada con la anterior, y con un origen conceptual totalmente distinto.

La idea básica es que los sistemas de razonamiento automático basados en lógica utilizan diversos mecanismos de resolución. En el caso de que se trate de una lógica con igualdad, como las propias Lógicas de Descripción, se obtienen los mejores resultados utilizando mecanismos de *paramodulación* ordenada; la superposición es uno de estos mecanismos, que se caracteriza porque el que el orden en que se realizan las reescrituras de términos se elige considerando solamente uno de los miembros de la igualdad.

Recientemente, varios autores [8] han planteado la posibilidad de utilizar este tipo de mecanismos para realizar una fusión (*merge*) de modelos ontológicos en las Lógicas de Descripción de tipo *SHIQ*, como la usada en OWL y RACER. Se parte, por tanto, de varios ABox separados, pero la deducción se hace sobre el sistema conjunto, definido en este caso mediante mecanismos de superposición; se usan también relaciones de subsunción para eliminar las redundancias del modelo. En nuestro caso, la superposición estructural serviría de base para guiar a estos mecanismos de deducción.

5. Conclusiones y Trabajo Futuro

Las ideas expuestas en este trabajo se encuentran aún en una fase preliminar. El enfoque que propone separar los distintos aspectos de un sistema, para luego combinarlos en una ontología, parece muy prometedor. Del mismo modo, la relación de superposición parece la manera más adecuada de plantear la fusión de los distintos aspectos; no

obstante, hay aún muchos aspectos a determinar, tanto a la hora de describir cuáles han de ser las propiedades de un “aspecto ontológico”, como a la hora de implementar estas nociones sobre un mecanismo de razonamiento existente. En particular, se ha de determinar aún la bondad del enfoque basado en superposición lógica a la hora de manipular los sistemas con superposición estructural.

La elaboración de una ontología utilizando un mecanismo de separación de aspectos hace posible estructurar la información y razonar sobre el dominio de conocimiento de los propios aspectos, así como sobre los dominios de aplicabilidad de dichos aspectos, basando el razonamiento en los mecanismos de fusión de ontologías. La estructura resultante tiene múltiples aplicaciones.

Por ejemplo, la construcción de una ontología estructurada mediante aspectos permite la posibilidad de considerar un nivel de construcción de ontologías, dentro de un proceso de desarrollo dirigido por modelos. Así, se podría construir una ontología estructurada de este tipo, en base a la información suministrada por los distintos metamodelos y modelos; los metamodelos de los distintos *profiles* proporcionarían la base para los distintos “aspectos ontológicos”. La ontología obtenida mediante la fusión de estos aspectos permitiría establecer un vocabulario común, lo que, entre otras cosas, facilitaría la comunicación entre las herramientas utilizadas en el proceso.

Referencias

1. Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, y Peter F. Patel-Schneider, editores. *The Description Logic Handbook*. Cambridge University Press, 2002.
2. Leo Bachmair y Harald Ganzinger. Rewrite-Based Equational Theorem Proving with Selection and Simplification. *Journal of Logic and Computation*, 4(3):217–247, 1994.
3. Siobhán Clarke y Robert J. Walker. Generic Aspect-Oriented Design with Theme/UML. En Robert E. Filman, Tzilla Elrad, Siobhán Clarke, y Mehmet Akşit, editores, *Aspect-Oriented Software Development*, págs. 425–458. Addison-Wesley, Boston, 2005.
4. José Luiz Fiadeiro y Tom S.E. Maibaum. Categorical Semantics of Parallel Program Design. *Science of Computer Programming*, 28(2–3):111–138, 1997.
5. Volker Haarslev y Ralf Möller. Racer System Description. En *IJCAR '01: Proc. First Intl. Joint Conf. on Automated Reasoning*, págs. 701–706, London, UK, 2001. Springer-Verlag.
6. William H. Harrison, Harold L. Ossher, y Peri L. Tarr. Asymmetrically vs. Symmetrically Organized Paradigms for Software Composition. IBM Research Report RC22685 (W0212-147), Thomas J. Watson Research Center, IBM, Diciembre 2002.
7. Ian Horrocks. The FaCT System. En *Proc. Intl. Conf. Automated Reasoning with Analytic Tableaux and Related Methods*, volumen 1397, págs. 307–312, 1998.
8. Ullrich Hustadt, Boris Motik, y Ulrike Sattler. Reducing \mathcal{SHIQ}^- Description Logic to Disjunctive Datalog Programs. En Didier Dubois, Christopher A. Welty, y Mary-Anne Williams, editores, *Principles of Knowledge Representation and Reasoning: Proc. 9th Intl Conf (KR'2004)*, págs. 152–162, Whistler, Canada, Junio 2004. AAAI Press.
9. Shmuel Katz. A Superimposition Control Construct for Distributed Systems. *ACM Trans. on Programming Languages and Systems*, 15(2):337–356, Abril 1993.
10. Pertti Kellomäki. A Formal Basis for Aspect-Oriented Specification with Superposition. En Gary T. Leavens y Ron Cytron, editores, *FOAL 2002 Proceedings: Foundations of Aspect-Oriented Languages*, págs. 27–32, Abril 2002. ISU-TR02-06.
11. Michael K. Smith, Chris Welty, y Deborah L. McGuinness, editores. *OWL Ontology Web Language Guide*. W3C Recommendation, Febrero 2004.