

# Gestión Semántica: Aplicando las Ontologías a la Gestión de Red

Jorge E. López de Vergara, Víctor A. Villagrà, Julio Berrocal.  
Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid.  
E.T.S.I. de Telecomunicación. Av. Complutense, s/n. 28040 Madrid.  
Teléfono: 91 549 57 00. Fax: 91 336 73 33.  
E-mail: {jlopez, villagra, berrocal}@dit.upm.es

*Abstract. The multiplicity of Network Management models (SNMP, CMIP, DMI, WBEM...) has raised the need of defining multiple mechanisms to allow the interoperability among all involved management domains. One basic component of such interoperability is the mapping between the information models that each domain specifies. These mappings, usually carried out with syntactical translations, can reach the semantic level by using ontologies. This article shows the advantages of using formal ontology techniques to improve the integration of current network management models. Applying this representation method, network managers can work and reason with an abstract view of the management information, independent of the specific management model used to interoperate with the managed resources, which can also include basic behaviour constraints.*

## 1 Introducción

En la actualidad existen varios modelos de gestión de red integrada que usan distintas tecnologías. Entre ellos está SNMP (*Simple Network Management Protocol*, Protocolo Simple de Gestión de Red), CMIP (*Common Management Information Protocol*, Protocolo Común de Gestión de Información), DMI (*Desktop Management Interface*, Interfaz de Gestión de Ordenadores de Sobremesa) y WBEM (*Web Based Enterprise Management*, Gestión Basada en Web de Empresa). También se han aplicado tecnologías de procesamiento distribuido como CORBA (*Common Object Request Broker Architecture*, Arquitectura Común de Intermediarios de Peticiones de Objetos) [1]. Cada uno de estos modelos ha identificado la necesidad de definir información para describir los recursos a gestionar, permitiendo de esta forma que exista un conocimiento común entre gestores y agentes. Por ello, cada modelo de gestión posee un lenguaje de definición propio con el que se han definido bases de información de gestión (MIBs, *Management Information Bases*) también específicas de cada dominio de gestión.

Esta multiplicidad de modelos supone un problema cuando hay que utilizar distintas de estas tecnologías para acceder a los distintos recursos que componen un sistema. En estos casos es necesario establecer mecanismos que permitan la interoperabilidad entre los distintos modelos de gestión implicados. Sin embargo, las propuestas existentes hasta la fecha, tales como IIMC (*ISO-Internet Management Coexistence*, Coexistencia de Gestión ISO e Internet), JIDM (*Joint Inter-Domain Management*, Gestión Inter-Dominio Unificada), o incluso CIM (*Common Information Model*, Modelo

de Información Común) sólo han planteado traducciones sintácticas que únicamente reescriben los modelos de información [2]. Esto quiere decir que si un mismo recurso está descrito en dos modelos de información, se puede realizar una traducción directa entre las estructuras que componen la especificación, pero no entre el significado que contienen, lo que supone un problema para alcanzar una gestión integrada. Para solventarlo, es necesario aprovechar la semántica contenida en la información.

Al mismo tiempo, las ontologías se han utilizado con éxito para resolver problemas similares en otros dominios como el de la Web Semántica [3], en el que esta técnica de representación del conocimiento proporciona semántica a las páginas y servicios web. Este artículo estudia cómo las ontologías también pueden ser útiles para la gestión de red, permitiendo unificar desde un punto de vista semántico las actuales definiciones heterogéneas de información. Para ello, antes que nada se presentan las ontologías y se comparan con los actuales modelos de información de gestión. Tras esto se proponen distintos pasos para obtener modelos con una mayor capacidad semántica, e integrables en una ontología de gestión. Finalmente se aportan las conclusiones obtenidas en este estudio.

## 2 Ontologías

Las Ontologías son una de las principales aproximaciones utilizadas en el ámbito de la Gestión del Conocimiento y la Inteligencia Artificial para resolver cuestiones relativas a la semántica. Para comprender cómo se pueden aplicar a los modelos de información de gestión, conviene explicar en qué consisten. Una Ontología

se puede definir como “una especificación explícita y formal de una conceptualización compartida” [4]:

- Es explícita porque define los conceptos, propiedades, relaciones, funciones, axiomas y restricciones que la componen.
- Es formal porque es legible e interpretable por máquinas.
- Es una conceptualización porque es un modelo abstracto y una vista simplificada de las entidades que representa.
- Finalmente, es compartida porque ha habido un consenso previo sobre la información, que ha sido acordado por un grupo de expertos.

En breve se puede decir que una ontología es la definición de un conjunto de conceptos, su taxonomía, interrelación y las reglas que gobiernan dichos conceptos.

Las ontologías se pueden clasificar en dos grandes grupos: las ligeras y las pesadas. Las primeras incluyen a aquellas que modelan la información referida a un dominio pero sin usar axiomas o restricciones, por lo que es difícil razonar con ellas. Las segundas sí que incluyen todos los elementos que les permiten inferir conocimiento a partir de la información definida.

Según esto, los modelos de información de gestión existentes se podrían entender como ontologías ligeras. Las MIBs de Internet o los esquemas CIM definen la información del dominio de la gestión de manera parcialmente formal y han sido acordadas en grupos de trabajo. Sin embargo, su semántica está limitada, al no poder especificar restricciones sobre la información [5].

Otra cuestión que diferencia a las ontologías de los modelos de información de gestión es la forma en que se aborda el problema de la interoperabilidad. Las soluciones que se aplican para integrar distintas ontologías no tratan sólo la traducción sintáctica de distintos lenguajes, sino otras cuestiones para permitir la interoperabilidad a un nivel semántico. Las propuestas por parte de los distintos grupos de investigación que trabajan en este ámbito incluyen la fusión de modelos para obtener un modelo común, o bien el establecimiento de correspondencias entre los modelos.

### 3 Aplicando las ontologías a la gestión de red

La presente sección contempla el problema de integración de los modelos de gestión desde la perspectiva de las ontologías y plantea una propuesta para mejorar la interoperabilidad semántica de las distintas especificaciones de

información. Para ello se han estudiado tres pasos, contenidos en los tres apartados siguientes, ilustrados en la Figura 1:

1. Primero se estudian las posibilidades de emplear un lenguaje de definición de ontologías para especificar información de gestión, adaptando dicho lenguaje para que pueda expresar algunas construcciones típicas de los lenguajes de información de gestión.
2. Partiendo de los modelos de información, expresados con lenguajes de ontologías, se podrán aplicar de manera conjunta las técnicas de fusión y correspondencia definidas en este campo. Para ello se define un método que combina ambas técnicas y las particulariza para el caso concreto de la información de gestión. Con esto se podrá obtener un nuevo modelo que realmente integre los ya existentes teniendo en cuenta la semántica contenida en los mismos y declarando al mismo tiempo las distintas reglas de correspondencia con los modelos iniciales. De esta manera se podrá solventar el problema de interoperabilidad identificado anteriormente.
3. Finalmente, se analiza la forma de incluir comportamiento a las especificaciones de información de gestión aprovechando las características de formalización que proporcionan los lenguajes de ontologías, pudiendo integrar dichas especificaciones en el modelo común generado anteriormente para poder tener modelos de información completos como ocurre con las ontologías pesadas.

#### 3.1 Escribiendo información de gestión con lenguajes de ontologías

Para especificar la información de gestión teniendo en cuenta el significado de la misma, sería necesario un lenguaje de definición cuya capacidad expresiva le permita expresar una semántica precisa, incluyendo restricciones sobre la

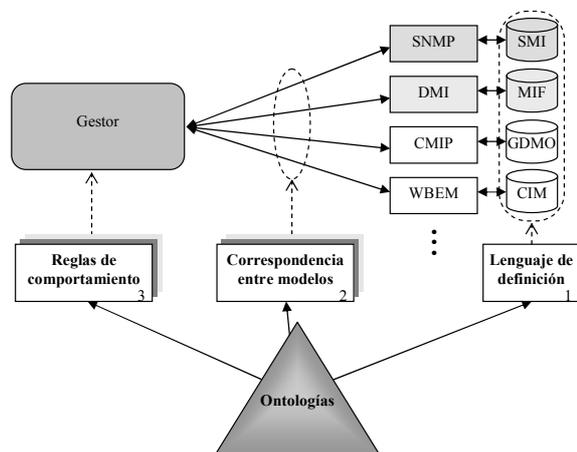


Figura 1. Aplicaciones de las ontologías.

información. Este apartado estudia las posibilidades de emplear un lenguaje de definición de ontologías para especificar información de gestión. Esto supone tener que adaptar dicho lenguaje para que también pueda expresar algunas construcciones típicas de los lenguajes de información de gestión.

Existen otras propuestas en este sentido. En [6] se propone una traducción entre SMIng (*Structure of Management Information, next generation*, Siguiente Generación de la Estructura de la Información de Gestión) y RDF (*Resource Description Framework*, Marco de Descripción de Recursos) para que lo puedan usar agentes inteligentes que se comuniquen mediante un vocabulario común basado en ontologías. Aunque en este caso se definen términos referidos a SMIng en RDF, no se hace para el caso general de cualquier lenguaje de gestión. Por otro lado, en [7] se trata de integrar el metamodelo de CIM dentro de una herramienta de ontologías. La justificación es poder utilizar la información definida en CIM para el intercambio de información entre agentes inteligentes usando OKBC (*Open Knowledge Base Connectivity*, Conectividad Abierta de Bases de Conocimiento) [8]. Sin embargo, no llega a utilizar directamente ningún lenguaje de ontologías para la definición de información de gestión.

Los lenguajes de ontologías actualmente en mayor uso y con mayor número de herramientas disponibles son aquellos relacionados con la Web Semántica. Entre estos destaca DAML+OIL (*DARPA Agent Markup Language + Ontology Inference Layer*, Lenguaje de Marcas de Agentes de DARPA + Capa de Inferencias de Ontologías) [9]. Otra ventaja de este lenguaje frente a los de definición de información de gestión es que está formalizado en KIF (*Knowledge Interchange Format*, Formato de Intercambio de Conocimiento) [10], con lo que su semántica es unívoca y puede ser usada por sistemas inteligentes. Por el contrario, DAML+OIL no es un lenguaje específico de gestión, y no posee construcciones para definir todas las facetas típicas de gestión.

Aunque el estudio que se muestra en este apartado se refiere a DAML+OIL, se puede generalizar para otros lenguajes de definición de ontologías con características similares. Todos aquellos lenguajes que permitan definir clases y propiedades pueden valer para definir información de gestión, si bien es posible que se pierda claridad si estos lenguajes no poseen las facetas adecuadas, ni un mecanismo para definir otras nuevas.

DAML+OIL es un lenguaje de ontologías bastante completo, pues permite definir clases y propiedades, que pueden estar en el dominio de una clase. A su vez, las propiedades poseen distintas facetas, como la restricción de tipo o de cardinalidad, así como la documentación. También permite definir especializaciones de clases, con

herencia múltiple, así como otro tipo de relaciones con restricción de rango. Finalmente, también se pueden expresar ejemplares de estas clases. Como muestra de ello, a continuación se define en DAML+OIL la clase del esquema nuclear de CIM `CIM_ManagedSystemElement` y sus propiedades asociadas. Los tipos de datos utilizados son los definidos en XSD (*XML Schema Data types*, Tipos de Datos de Esquemas XML) [11].

```
<daml:Class rdf:ID=
  "CIM_ManagedSystemElement">
  <rdfs:comment>
CIM_ManagedSystemElement is the base class for
the System Element hierarchy. [...]
  </rdfs:comment>
  <rdfs:subClassOf rdf:resource=
    "#CIM_ManagedElement" />
</daml:Class>

<daml:DatatypeProperty rdf:ID=
  "InstallDate">
  <rdfs:comment>
A datetime value indicating when the object was
installed. A lack of a value does not indicate that
the object is not installed.
  </rdfs:comment>
  <daml:domain rdf:resource=
    "#CIM_ManagedSystemElement" />
  <daml:range rdf:resource=
    "http://www.w3.org/2000/10/XMLSchema
#dateTime" />
</daml:DatatypeProperty>

<daml:DatatypeProperty rdf:ID="Name">
  <rdfs:comment>
The Name property defines the label by which the
object is known. When subclassed, the Name
property can be overridden to be a Key property.
  </rdfs:comment>
  <daml:domain rdf:resource=
    "#CIM_ManagedSystemElement" />
  <daml:range rdf:resource=
    "http://www.w3.org/2000/10/XMLSchema
#string" />
</daml:DatatypeProperty>

<daml:DatatypeProperty rdf:ID="Status">
  <rdfs:comment>
A string indicating the current status of the object.
Various operational and non-operational statuses
are defined. [...]
  </rdfs:comment>
  <daml:domain rdf:resource=
    "#CIM_ManagedSystemElement" />
  <daml:range rdf:resource=
    "http://www.w3.org/2000/10/XMLSchema
#string" />
</daml:DatatypeProperty>
```

No obstante, DAML+OIL no incluye facetas típicas de los lenguajes de gestión como el valor por defecto o el tipo de acceso, entre otras. Para que este lenguaje contenga dicha información habría que definir en RDF cada una de estas construcciones. A continuación se presenta un par

de ejemplos que especifican algunas facetas de gestión:

```
<!-- VALOR POR DEFECTO -->
<rdf:Property rdf:ID="defaultValue">
  <rdfs:label>defaultValue</rdfs:label>
  <rdfs:comment>
    It defines the default value of a property.
  </rdfs:comment>
  <rdfs:domain rdf:resource=
    "http://www.w3.org/1999/02/22-rdf-
    syntax-ns#Property" />
</rdf:Property>
```

```
<!-- ACCESO -->
<rdf:Property rdf:ID="access">
  <rdfs:label>access</rdfs:label>
  <rdfs:comment>
    It defines the access of a property, which can be
    readable and/or writeable.
  </rdfs:comment>
  <rdfs:domain rdf:resource=
    "http://www.w3.org/1999/02/22-rdf-
    syntax-ns#Property" />
  <rdfs:range rdf:resource=
    "#accessString" />
</rdf:Property>

<xsd:simpleType name="accessString">
  <xsd:restriction base="string">
    <xsd:enumeration value=
      "read-only" />
    <xsd:enumeration value=
      "read-write" />
    <xsd:enumeration value=
      "read-create" />
  </xsd:restriction>
</xsd:simpleType>
```

Con todo, no es posible expresar métodos u operaciones en este lenguaje. Sin embargo, en general las especificaciones de información de gestión no suelen incluir muchas de estas construcciones, por lo que éste no es un gran problema.

Entonces, a partir de las estructuras básicas de DAML+OIL junto con las específicas de gestión, la información de gestión actual es fácilmente traducible a un lenguaje de ontologías. Únicamente habría que comparar las construcciones que posee cada lenguaje como se hace en [12] para llevar a cabo esta traducción.

### 3.2 Fusión y Correspondencia (Método M&M)

Aunque todos los modelos de información de gestión se expresen en un mismo lenguaje, esto no supone integrarlos semánticamente, a pesar de que el lenguaje sea de ontologías. Es necesario llevar a cabo otros procedimientos que identifiquen el significado contenido en estos modelos. Esto es posible haciendo uso de las técnicas de fusión y correspondencia empleadas en las ontologías. Para

ello este apartado expone un método que no se basa en realizar reescrituras de la información, sino que combina distintas definiciones en un modelo común, declarando las distintas reglas de correspondencia con los modelos iniciales, como se ilustra en la Figura 2.

Para asistir en este proceso se ha definido el método denominado M&M (*Merge and Map*, Fusión y Correspondencia). En él se propone un conjunto de pasos para ayudar a obtener simultáneamente el modelo común y las reglas de correspondencia. Está basado en el método de fusión de [13] adaptándolo al caso particular de la gestión de red, y añadiendo todo lo que se refiere a la definición de reglas de correspondencia, con lo que por cada elemento que se fusione se añade una regla que relaciona los elementos fusionados en la ontología de correspondencia.

Este método no genera un resultado de manera automática. Se trata de un método que ayuda a la persona que deba realizar esta labor en el proceso de fusión y correspondencia. Para esto se proponen los siguientes heurísticos, que permitirán identificar candidatos a fusionar con gran probabilidad:

- De correspondencia por similitud en cadenas de caracteres, analizando y comparando aquellas que estén incluidas en identificadores de clases o propiedades, o en su descripción, y seleccionando aquellas que sean similares.
- De correspondencia por similares jerarquías de herencia, que permitirán fusionar clases si sus clases madres son semejantes, dado que las clases hijas de una dada suelen ser similares a las clases hijas de la clase fusionada con ésta.
- De correspondencia por el dominio al que pertenecen las propiedades, pues si una clase se corresponde con otra, será normal que las propiedades de dicha clase también se correspondan con las de la otra. Al mismo tiempo, si es posible identificar dos propiedades semejantes, también se puede suponer que las clases a las que pertenecen también se corresponden.

Para facilitar la fusión no se ha pensado en definir un nuevo modelo común, sino partir de uno de los

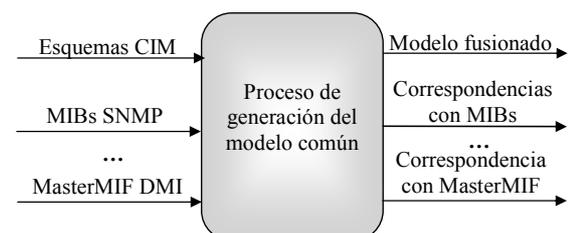


Figura 2. Proceso de fusión y correspondencia de información de gestión.

ya existentes [12]. En este caso se utiliza el esquema CIM como base para fusionar otros modelos ya existentes, como la MIB-II, la HOST-RESOURCES-MIB o la definida en la recomendación M.3100.

También se ha identificado que las correspondencias típicas entre los elementos de los modelos de información de gestión son las siguientes o una combinación de las mismas:

- Directa, si es una relación 1:1 en la que no hay que hacer ninguna transformación. En este caso, un elemento de un modelo es exactamente igual al de otro modelo.
- De valores, si es una relación 1:1 en la que cada elemento toma valores según una enumeración, que es distinta para cada caso.
- De tipos de datos, si es una relación 1:1 en la que cada elemento posee un tipo de datos distintos, teniéndose que adaptar para que cada dominio de gestión reciba el que tiene definido.
- De operación aritmética sobre un elemento, si es una relación 1:1 en la que un elemento se obtiene realizando una operación aritmética con una constante sobre el otro elemento. Este caso se da, por ejemplo, cuando hay que cambiar las unidades con que se mide ese parámetro.
- De operación aritmética sobre varios elementos, si es una relación 1:n en la que un elemento se obtiene por combinación aritmética del resto.
- De cadenas de caracteres, si es una relación 1:n en la que un elemento se compone de la concatenación de distintas cadenas de caracteres.

Para describir las reglas de correspondencia que traduzcan los ejemplares de un modelo concreto al modelo común se ha definido una ontología muy simple, representada en la Figura 3.

Su estructura es la siguiente: cada posible elemento (Element) que compone una ontología (clases, propiedades, etc.) posee una fórmula (Formula) de traducción. Además, cada elemento contiene

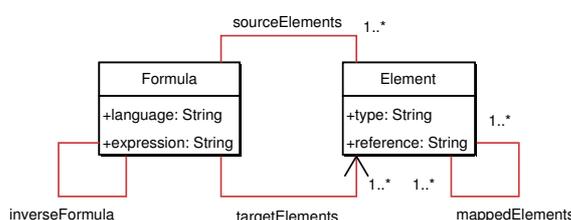


Figura 3. Ontología de correspondencia.

propiedades tales como el tipo (type) o la referencia a su definición (reference). Por su parte, las fórmulas se relacionan con el conjunto de elementos origen (sourceElements) y destino (targetElements), así como una expresión (expression) usada para traducir entre ambos en un lenguaje concreto (language). También se incluyen relaciones con los elementos de la otra ontología con que se corresponde (mappedElements) y la fórmula inversa (inverseFormula). Con esta ontología es posible representar cualquier correspondencia de las definidas anteriormente. Además, no existen las limitaciones del calificador MappingStrings de CIM, que sólo permite representar correspondencias directas.

La Figura 4 muestra el diagrama de actividad que describe el método M&M. Las actividades con fondo gris son aquellas que realizaría el usuario, mientras que las de fondo blanco serían efectuadas por el sistema. En resumen, el método M&M consiste en identificar a través de los heurísticos comentados clases similares, y tras esto fusionar los distintos atributos que hay en ellas. Al mismo tiempo, se define de manera automática en la ontología de correspondencia cada elemento a fusionar, y luego se van definiendo las fórmulas asociadas y elementos que se corresponden con éste. Durante este proceso, la persona que lo realiza debe ir validando cada una de las operaciones propuestas por el método, pudiendo también definir otras diferentes. El resultado final es un modelo común y un conjunto de ejemplares de la ontología de correspondencia que representen las reglas de traducción de los modelos fusionados.

Un gestor basado en el modelo común y esta ontología de correspondencia funcionaría de la siguiente manera. Si necesita obtener todos los ejemplares de un elemento concreto del modelo común, lo buscará en la ontología de correspondencia, encontrando a la vez la fórmula y elementos correspondientes de los modelos fusionados. Se accederá a cada dominio y se obtendrán dichos elementos. Al aplicar la expresión contenida en la fórmula se traduciría el valor de los ejemplares obtenidos en cada dominio de gestión al modelo que maneja el gestor.

### 3.3 Definiciones de comportamiento

Un paso más allá de la integración de información es el que se consigue añadiendo un conjunto de restricciones al modelo común de gestión que se obtenga. Esto permite describir el comportamiento relativo a la información contenida en dicho modelo, que podrá ser comprobado en el gestor. Las actuales definiciones de información de gestión incluyen algunas reglas acerca del comportamiento de la información de gestión, pero están escritas en los campos BEHAVIOUR o DESCRIPTION en lenguaje natural que no es legible por una máquina.

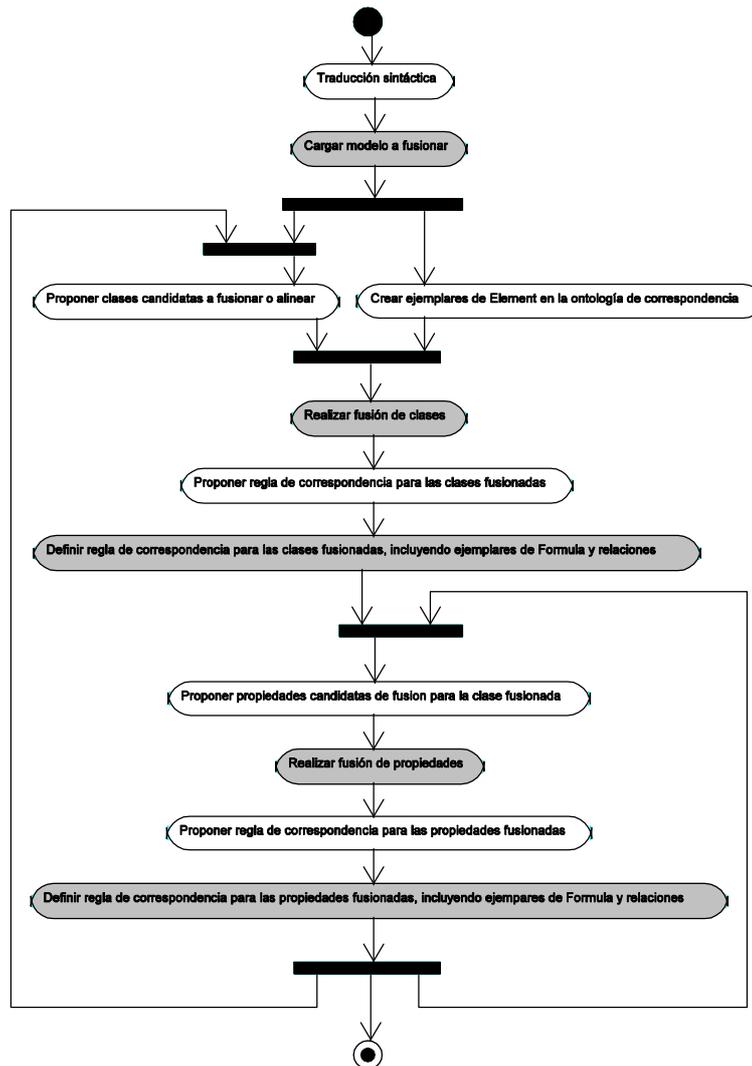


Figura 4. Diagrama de actividad del método M&M.

También aquí se pueden aplicar las ontologías, pues suelen incluir axiomas y restricciones para especificar su comportamiento, que al definirse formalmente sí pueden interpretarse de manera automática.

Las restricciones a incluir en la información pueden ser de dos tipos. El primero se ha denominado restricción implícita y se refiere a la información que en un estado de operación normal siempre se deben cumplir, porque así es como se ha definido la información. El segundo tipo se ha denominado restricción explícita y se refiere a que un gestor concreto pueda definir nuevas restricciones sobre información ya definida para particularizar de esta manera el comportamiento de los sistemas gestionados.

Las restricciones implícitas se encuentran habitualmente definidas en lenguaje natural en las descripciones de las clases y atributos de información de gestión. Un ejemplo de ello podrían ser las restricciones contenidas dentro de la clase `CIM_Printer` de los esquemas CIM, como la que dice “*a language that is used as a default by the*

*Printer should also be listed in LanguagesSupported*”, que restringe el lenguaje que debe manejar una impresora a los contenidos en la lista de lenguajes soportados.

Las restricciones explícitas siguen una política concreta. Por ejemplo, se puede definir una política que haga cumplir que el espacio disponible en los sistemas de archivos de un equipo nunca deberá ser inferior del 10% de la capacidad total.

Respecto a la definición de estas restricciones, DAML+OIL permite definir las con lógica de primer orden respecto a propiedades algebraicas de relaciones (simetría, transitividad, unicidad). Además, se pueden definir restricciones de universalidad y existencia para las clases y propiedades. Sin embargo, DAML+OIL no es un lenguaje totalmente completo a la hora de especificar restricciones, pues éstas se refieren a propiedades que referencian objetos, no siendo igual de expresivo con propiedades que contienen tipos de datos. Por este motivo, otros lenguajes de ontologías como KIF son mejores para definir este tipo de información.

El ejemplo propuesto para la restricción implícita podría definirse de la siguiente manera en DAML+OIL:

```
<daml:Class rdf:about="#CIM_Printer">
  <daml:Restriction>
    <daml:onProperty rdf:resource=
      "#LangaugesSupported"/>
    <daml:hasClass rdf:resource=
      "#DefaultLanguage"/>
  </daml:Restriction>
</daml:Class>
```

Asimismo, utilizando KIF, se expresaría de la siguiente manera:

```
(defrange ?printer :FRAME CIM_Printer)
(forall ?printer
  (element-of
    (DefaultLanguage ?printer)
    (LanguagesSupported ?printer)))
```

Si se emplea nuevamente KIF para el segundo ejemplo, esta sería:

```
(defrange ?fs :FRAME CIM_FileSystem)
(forall ?fs
  (> (AvailableSpace ?fs)
    * 0.10 (FileSystemSize ?fs)))
```

Para el caso de DAML+OIL se aprecia lo que se comentaba anteriormente. Es posible definir una restricción de valores aprovechando las construcciones de XSD, pero no se puede definir un valor máximo con una operación. Por ejemplo, si el espacio disponible se midiera en porcentaje y no en valores absolutos sí se podría definir la siguiente restricción:

```
<xsd:simpleType name="over10">
  <xsd:restriction base=
    "xsd:positiveInteger">
    <xsd:minInclusive value="10"/>
  <!-- Como es un porcentaje se podría
  incluir una restricción de máximo -->
    <xsd:maxInclusive value="100"/>
  </xsd:restriction>
</xsd:simpleType>

<daml:Class rdf:about="#CIM_FileSystem">
  <daml:Restriction>
    <daml:onProperty rdf:resource=
      "#AvailableSpace"/>
    <daml:toClass rdf:resource=
      "#over10"/>
  </daml:Restriction>
</daml:Class>
```

El comportamiento que se defina mediante estas restricciones se incluirá en la ontología que modele la información de gestión, con lo que puede ser más tarde automáticamente procesado por sistemas de gestión inteligentes que razonen con la información basada en las ontologías que se les proporcione.

## 4 Conclusiones

En este artículo se ha presentado cómo se puede aplicar la técnica de representación formal conocida

como ontología para mejorar la definición e integración de la información de gestión de red.

Escribir la información de gestión con un lenguaje de ontologías como DAML+OIL mejora la expresividad de dicha información, existiendo asimismo múltiples herramientas desarrolladas para su uso y validación. Sin embargo, también es necesario ampliar este tipo de lenguajes para que puedan expresar toda la información contenida habitualmente en los modelos de gestión.

Los trabajos existentes hasta la fecha respecto de la integración de información de gestión trataban ésta de forma muy limitada, basada sobre todo en traducciones sintácticas. La aplicación del método M&M permite realizar esta tarea a partir del significado contenido en esta información, lo que permite que un gestor pueda manejar un único modelo, con total transparencia de los dominios de gestión subyacentes. Esto redundará en una mejora de las aplicaciones de gestión, que pueden relacionar datos que hasta la fecha no tenían una asociación directa al pertenecer a distintos dominios de gestión. Durante la experimentación se ha constatado que su aplicación a modelos de información de gran tamaño puede llevar bastante tiempo debido a la necesidad de intervención humana para comprobar la validez de las reglas propuestas. Sin embargo, este tiempo será en general menor que si se realiza esta tarea a mano.

Al aprovechar las características de los lenguajes de ontologías, la posibilidad de definir de restricciones se obtiene por añadidura. Estas reglas, a diferencia de los modelos habituales, pueden ser interpretadas. Con esto, se puede obtener una especificación de información de un modelo común e independiente del dominio de gestión, cuyos conceptos además posean restricciones acerca de comportamiento.

Basándose en estas ideas, se puede desarrollar un sistema de gestión como el que se muestra en la Figura 5, que aproveche esta aproximación basada en ontologías, integrando todos los modelos de gestión de manera inteligente, teniendo en cuenta la semántica de la información definida. Al mismo tiempo, se pueden construir pasarelas genéricas que utilicen las ontologías de correspondencia obtenidas al aplicar el método M&M para traducir la información a cada dominio de gestión.

Los trabajos actuales incluyen traductores automáticos de CIM y SMI a DAML+OIL. Al mismo tiempo, se está adaptando el método M&M a una herramienta de ontologías existente, para automatizar el proceso de fusión y correspondencia.

## Agradecimientos

Este trabajo ha sido parcialmente financiado por el Ministerio de Ciencia y Tecnología a través del proyecto GESEMAN (TIC2002-00934).

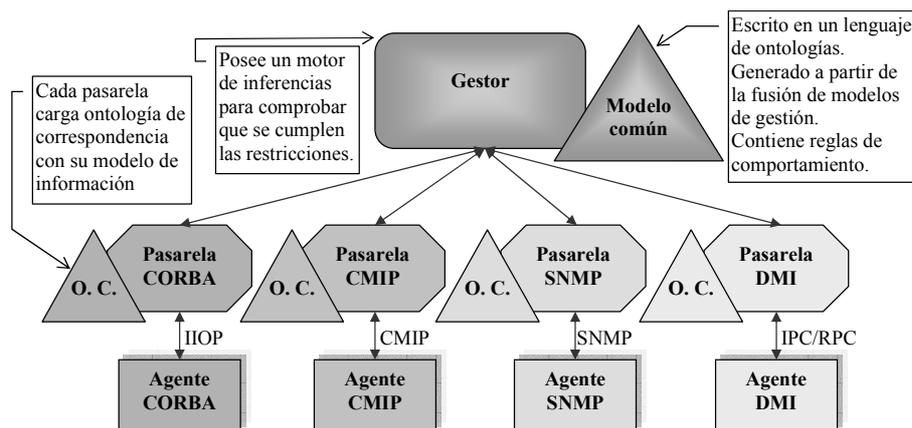


Figura 5. Arquitectura de un gestor que maneje ontologías

## Referencias

- [1] Heinz-Gerd Hegering, Sebastian Abeck, Bernhard Neumair, "Integrated Management of Networked Systems". Morgan Kaufmann, 1999.
- [2] Jorge E. López de Vergara, Víctor A. Villagrà, Julio Berrocal, "Semantic Management: advantages of using an ontology-based management information meta-model", Proceedings of the HP Openview University Association Ninth Plenary Workshop (HP-OVUA'2002), Böblingen, Alemania, junio de 2002.
- [3] Tim Berners-Lee, James Hendler, Ora Lassila, "The Semantic Web", Scientific American, mayo de 2001.
- [4] R. Studer, V.R. Benjamins, D. Fensel, "Knowledge Engineering: Principles and Methods", Data & Knowledge Engineering. 25: 161-197, 1998.
- [5] Jorge E. López de Vergara, Víctor A. Villagrà, Julio Berrocal, Juan I. Asensio, Roney Pignaton, "Semantic Management: Application of Ontologies for the Integration of Management Information Models", Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003), Colorado Springs, Colorado, EE. UU. A., marzo de 2003.
- [6] Jun Shen, Yun Yang, "RDF-Based Knowledge Models for Network Management", Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003), Colorado Springs, Colorado, EE. UU. A., marzo de 2003.
- [7] Emmanuel Lavinal, Thierry Desprats, Yves Raynaud, "A Conceptual Framework for Building CIM-Based Ontologies", Proceedings of the Eighth IFIP/IEEE International Symposium on Integrated Network Management (IM'2003), Colorado Springs, Colorado, EE. UU. A., marzo de 2003.
- [8] Vinay K. Chaudhri, Adam Farquhar, Richard Fikes, Peter D. Karp, James P. Rice, "OKBC: A Programmatic Foundation for Knowledge Base Interoperability", Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI'98), Madison, Wisconsin, EE. UU. A., julio de 1998.
- [9] Dan Connolly, Frank van Harmelen, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, Lynn Andrea Stein, "DAML+OIL (March 2001) Reference Description", W3C Notes, 18 de diciembre de 2001.
- [10] Richard Fikes, Deborah McGuinness, "An Axiomatic Semantics for RDF, RDF-S, and DAML+OIL (March 2001)", W3C Note, 18 de diciembre de 2001.
- [11] Paul V. Biron, Ashok Malhotra, "XML Schema Part 2: Datatypes", W3C Recommendation, 2 de mayo de 2001.
- [12] Jorge E. López de Vergara, Víctor A. Villagrà, Juan I. Asensio, Julio Berrocal, "Ontologies: Giving Semantics to Network Management Models", IEEE Network, special issue on Network Management, Volume 17, Number 3, mayo/junio de 2003.
- [13] Natalya Fridman Noy, Mark A. Musen, "An Algorithm for Merging and Aligning Ontologies: Automation and Tool Support", Proceedings of the Workshop on Ontology Management, Sixteenth National Conference on Artificial Intelligence (AAAI-99), Orlando, Florida, EE. UU. A., julio de 1999.