

# Ontología para representar el conocimiento de Algoritmos básicos

## Ontology to represent the knowledge of basic algorithms

Lucy Nohemy Medina Velandia PhD (c)<sup>1</sup>, Álvaro Escobar Escobar MBA<sup>2</sup>.

1. Grupo Desarrollo tecnológico y ciencias del espacio – Línea de Investigación Desarrollo de Software.

2. Grupo Desarrollo tecnológico y ciencias del espacio – Línea de Investigación Convergencia.

Universidad Sergio Arboleda, Bogotá.

*lunome@gmail.com ; lucy.medina@usa.edu.co ; alvaroescobar@usa.edu.co*

Recibido para revisión 02 de octubre de 2010, aceptado 03 de enero de 2011, versión final 09 de febrero de 2011

**Resumen**— El objetivo de este documento es explicar por qué y cómo se desarrolló una ontología en el área de algoritmos básicos, que a través de tecnologías asociadas como la Web Semántica y los Sistemas Multiagentes, dan cumplimiento al proyecto macro, del cual hace parte la construcción de la ontología. Particularmente, se pretende que la construcción de la ontología sea uno de los pilares para soportar otros proyectos que se relacionan con los temas antes mencionados.

Una solución que puede darse al problema de la heterogenización de la información y su dificultad para la recuperación de la misma de manera ágil, es a través de la trilogía Web Semántica, Ontologías y Agentes Inteligentes, cuya combinación permitirá reutilizar el conocimiento de una temática estudiada. Para atender las dificultades en la recuperación de la información se construyó la ontología OntoUSA, que trata la temática de algoritmos básicos para que más adelante, con la ayuda de un Sistema Multiagente Inteligente, pueda recuperarse el conocimiento capturado en la ontología y ser aprovechado por las personas interesadas.

**Palabras Clave**— Ontología, Protege, Sistemas Multi-agente, Web Semántica

**Abstract**— The aim of this paper is to explain why and how it developed an ontology in the area of basic algorithms, which through associated technologies like the Semantic Web and Multiagent Systems, give effect to the macro project, which is part of the building ontology. In particular, it is intended that the construction of ontology is one of the pillars to support other projects that relate to the aforementioned issues.

A solution may be the problem of heterogenization of information and the difficulty of recovering it in an agile, is through the trilogy Semantic Web, Ontologies and Intelligent Agents, whose combination enables to reuse the knowledge of a topic being studied. To address the difficulties in information retrieval

**OntoUSA ontology was built, that it deals with basic algorithms that later, with the help of an Intelligent Multi-Agent System, to recover the captured knowledge in the ontology and be exploited by stakeholders.**

**Keywords**— Methontology, Ontology, Protege, multi-agent systems, Semantic Web.

### I. INTRODUCCIÓN

Recuperar información en la web es un proceso que engloba un conjunto de acciones como es la identificación, selección y acceso a recursos de información para extraer documentos de una base de datos, listas, catálogos o directorios [1]. Estas tareas se han vuelto complejas y ameritan reflexionar sobre, la cantidad, calidad y precisión de lo recuperado. Éstos son algunos de los múltiples cuestionamientos que los usuarios pueden hacerse al obtener documentos sobre un tema específico; debido a que en la web existe mucha información sobre temáticas diversas, pero éstas no están clasificadas, indizadas e inclusive los formatos y contenidos heterogéneos.

Existe una idea clara entre las personas que se dedican a transformar la red en un verdadero espacio para que el saber no sea solamente información disgregada, sino que se le dé un tratamiento inteligente que no requiera de la intervención humana y en el que los datos que representen conocimiento en un dominio específico puedan ser interrelacionados, de tal forma que si se necesita hacer una búsqueda, ésta se haga de forma rápida y sencilla, para facilitar la comunicación y el intercambio entre personas y entre sistemas.

Luego de haber identificado el problema sobre el aprendizaje en el tema de algoritmos en varias universidades bogotanas a través de encuestas, se evidenció la dificultad que los estudiantes han tenido para captar las temáticas tratadas en las asignaturas de algoritmos, bien sea porque no entienden a su profesor, o porque no les es posible pasar los problemas reales a código y explicitar los temas concretos como el estudio de los vectores, matrices y la práctica de los lenguajes de programación. Por ello se decidió desarrollar la ontología, bautizada como OntoUSA, que trata precisamente esa temática y sirve como punto de partida para el siguiente proyecto, en el cual se está desarrollando un sistema multiagente inteligente que permite extraer la información de la web de forma relacionada, ordenada y coherente para presentarla al estudiante.

## II. METODOLOGÍA UTILIZADA

Son dos los tipos de investigación que se siguieron para desarrollar el presente trabajo. De una parte, la investigación básica, que tomó teorías y métodos existentes en lo referente a ontologías, Web semántica y algoritmos; a partir de ellos se estableció una forma ordenada y coherente para desarrollar la ontología sobre algoritmos básicos.

De otra parte, se desarrolló la investigación aplicada. Luego de elaborar la ontología se publica en la web. A través de un Sistema Multiagente pueden extraerse los conocimientos de forma inteligente y ser utilizados en todo el mundo, especialmente por los estudiantes que toman asignaturas o gustan del tema de algoritmos.

Teniendo la metodología de investigación base para el proyecto, se desarrollaron seis fases que se presentan a continuación.

## III. DESCRIPCIÓN DE LA METODOLOGÍA

Fase 1. Fase exploratoria. Se realizó una revisión de la literatura; a partir de ella se consultó, se hizo la selección y se obtuvieron de forma selectiva la bibliografía y los materiales útiles sobre las temáticas de algoritmos, ontologías, Sistema Multiagente, web semántica, lenguajes para desarrollar ontologías y las plataformas apropiadas; se recopilaron las diversas ontologías que se han desarrollado en el mundo, en América Latina y Colombia, relacionadas con el tema.

Fase 2. Fase de descripción y explicación. Se revisaron a fondo las ontologías encontradas en los diferentes contextos analizados y se clasificaron por área de conocimiento. Esta clasificación permitió tener claridad sobre las posibles preguntas que se realizarían a los estudiantes de los diferentes cursos con el fin de conocer la problemática surgida al estudiar de la forma tradicional el tema de algoritmos.

Fase 3. Fase de encuestas. Se identificó el sector objetivo de las diferentes universidades bogotanas que hicieron parte en la aplicación del instrumento. Se prepararon los formatos para aplicar las encuestas, teniendo en cuenta los elementos propuestos en el problema de investigación y lo arrojado por el análisis de la Fase 2. Se aplicaron las encuestas a la población objetivo y se realizó el análisis de los datos con las conclusiones pertinentes.

Fase 4. Fase de construcción de la propuesta. Con base en la comparación y análisis de resultados de las Fases 2 y 3, se construyeron los lineamientos, las pautas o criterios para desarrollar la ontología. Una vez concluida ésta, se montó en un servidor para ser probada y utilizada.

Fase 5. Fase de documentación. Se preparó un documento general con la descripción del trabajo realizado, la especificación de la Ontología propuesta y el aporte que desde la línea de Desarrollo de Software de la Universidad Sergio Arboleda se está haciendo al difundirla entre las distintas universidades que puedan utilizarla.

Fase 6. Fase de difusión. La ontología, resultado de este trabajo, se ha presentado por ahora en eventos académicos internos en la Universidad Sergio Arboleda.

## IV. RESULTADOS Y ANÁLISIS

Para sustentar el problema que se solucionaría con la ontología construida, se utilizó la técnica de encuestas entre la población estudiantil de diferentes universidades, en los semestres 1, 2 y 3 de las Facultades de Ingeniería de Sistemas, pues son ellos los que cursan o han cursado la asignatura de algoritmos o las relacionadas y tienen mayor recuerdo de lo que fue o es esa materia, para algunos compleja, difícil o problemática, para otros, sencilla y fácil de entender. A continuación se presentan los resultados de dicho estudio.

La encuesta se conformó con 23 preguntas. Algunas de ellas incluyeron tópicos como la dificultad de los temas que estudiaron con respecto al desarrollo de algoritmos, la importancia que desde el punto de vista del estudiante tiene la asignatura en sus vidas, el tipo de lenguaje que utilizaron para aprender a programar y la utilización de las tutorías que ofrecen las universidades para solucionar las dudas o problemas de entendimiento; también se indagó sobre la actitud que tendrían los estudiantes una vez se termine el tutor inteligente que utilizaría la ontología construida para fortalecer el aprendizaje del tema propuesto.

Las universidades consultadas están situada en la ciudad de Bogotá, Colombia y fueron: Sergio Arboleda, Autónoma de Colombia y Antonio Nariño. La población total de estudiantes, o tamaño de la muestra examinada fue de 300 alumnos. Luego de obtener los datos se tabularon, codificaron y procesaron para obtener los resultados mostrados en la siguiente tabla:

**Tabla 1.** Tamaño de la muestra

Universidad	Total encuestados	Hombres	Mujeres
Sergio Arboleda	140	97	43
Autónoma de Colombia	100	75	25
Antonio Nariño	60	34	26
Total muestra	300	206	94

En la Universidad Sergio Arboleda las ingenierías tienen un tronco común, para que los estudiantes que ingresan en los primeros semestres puedan cambiar de carrera cuando consideren que cierta ingeniería no les es favorable, o no les gusta. Por ello, la asignatura de algoritmos la toman en común, estudiantes de Matemáticas, Ingeniería de Sistemas y Telecomunicaciones, Industrial y Electrónica. En primer semestre, el 54.20% se encuentra matriculado en Ingeniería Industrial, 25.23% en Ingeniería de Sistemas y Telecomunicaciones, 11.21% en Ingeniería Electrónica y 9.34% en Matemáticas. En segundo semestre, el 47.61% se matriculó en Industrial y el 52.38% en Sistemas y Telecomunicaciones. En tercer semestre, los estudiantes de Ingeniería de Sistemas son los únicos que continúan con las asignaturas de programación, y todos ellos pertenecen al programa.

Cabe señalar que el 85.046% de los encuestados se encuentra matriculado en la asignatura de algoritmos, que se toma en primer semestre, y sólo el 42.85% de los estudiantes que se encuentran en segundo semestre toman la de primer semestre, y el 16.66% de los alumnos de tercer semestre ve la misma materia; esto quiere decir que gran parte de ellos no repiten la asignatura, pero no a todos los que aprueban la asignatura les son claros los temas. Además, los grupos de estudiantes son numerosos en el primer semestre, debido a que la asignatura se dicta a estudiantes que están matriculados en las tres ingenierías; en segundo semestre sólo se unen Ingeniería de Sistemas y Telecomunicaciones con Electrónica; a partir del tercer semestre sólo siguen las asignaturas que tienen que ver con los conocimientos aplicados a ingeniería de sistemas.

Revisando las tres universidades participantes, de los estudiantes encuestados, el 68.57% cree que la asignatura de algoritmos les proporciona crecimiento profesional, 34.28% les da crecimiento intelectual y el 5.71% crecimiento personal. En cuanto a lo que más dificultad les causó de la asignatura de algoritmos, se tiene que el 20% no entiende muchos temas, el 15% ve difíciles los temas, el 13.58% cree que el profesor no explica bien y el 13% no sabe codificar lo que pide el profesor. Estos elementos se consideran suficientes para identificar completamente el problema que resolverá el propósito de esta investigación.

Al indagar por la dificultad de los temas específicos de la materia algoritmos, el 32.71% sostiene que desarrollar ejercicios por medio de un lenguaje de programación es lo más difícil; el 22.42% considera que las características de los algoritmos son lo más complicado de entender; para el 19.62% lo más difícil son las instrucciones repetitivas, 12.14% encuentran complejo

el tema de matrices o tablas, el 18.69%, cree que lo complejo son las instrucciones condicionales, 12.14% que son los métodos o funciones, 14.01%, que es el manejo de los operadores, y 7.47% que es el tema de los vectores.

Como en el segundo semestre, los estudiantes inician la programación con un lenguaje específico, a la pregunta ¿cuál de los temas vistos se les ha dificultado? las respuestas fueron las siguientes: para el 33.33% el tema más difícil fue el de las matrices; para el 23.80% el manejo de las instrucciones repetitivas y el uso de los métodos; en el manejo de cadenas, el 19.04% considera que les fue difícil este tratamiento; en cuanto a las instrucciones repetitivas, al 14.28% les fue complicado entenderlas y usarlas. Un porcentaje igual de estudiantes considera que el manejo de métodos fue muy confuso. Para el 4.77% es muy difícil el manejo y codificación de objetos.

En el tercer semestre, 15% de estudiantes ve por segunda vez la materia de algoritmos, e indica que el tema de matrices y vectores fue el que más dificultad les causó.

Los lenguajes de programación han sido para gran parte de los encuestados el obstáculo principal para entender la asignatura. No les es fácil pasar un problema real a la máquina; esto lo indica el 100% de los encuestados; también que no entienden muchos temas dictados por sus profesores.

Teniendo como premisas los resultados anteriores, se ratifica que OntoUSA es una herramienta que permitirá al alumno estudiar a su ritmo y en su tiempo, extrayendo el conocimiento allí almacenado lo cual se logrará por medio del Sistema Multiagente que se está construyendo para que opere la ontología.

## V. CREACIÓN DE ONTOUSA

Teniendo claro el problema que se va a solucionar y la temática en la que se centrará la ontología, se procede a su planeación. Para ello, se analizaron varios entornos para editar ontologías [5], así como lenguajes y metodologías de desarrollo [4].

Se escogió como editor Protege 4.1, el lenguaje OWL 2.0 para su desarrollo y como metodología para implementar la ontología se trabajó con Methonlogy.

Las razones por las cuales se eligieron los elementos de trabajo mencionados fueron, entre otras; Protege es un editor gratis y abierto, basado en java, que soporta frames, XML Schema, RDF y OWL; además cuenta con un ambiente plug and play. OWL se utiliza porque provee vocabulario para describir propiedades y clases, como por ejemplo las relaciones entre ellas, la cardinalidad, la equivalencia y las características de las propiedades. La metodología Methonlogy [7] se usó porque se hizo un estudio comparativo entre algunas de este tipo y dio como resultado la sencillez para el desarrollo de OntoUSA. Se

siguieron los pasos definidos en el estándar IEEE 1074 de desarrollo de software [2], que son:

- Especificación. Se definió el alcance y granularidad de la ontología.
- Conceptualización. Permitió organizar y estructurar el conocimiento adquirido mediante tablas, lenguaje UML, jerarquías, etc.
- Implementación. Se formalizó la ontología; es decir, pasar la conceptualización de la ontología al lenguaje OWL.
- Evaluación. Se comprobó el funcionamiento de la ontología (sin el Sistema Multiagente)

OntoUSA es considerada una ontología educacional, debido a que será utilizada en la enseñanza de algoritmos basada en tecnologías web. [3] Clasifica este tipo de tecnologías en ontologías de dominio, las cuales describen los conceptos esenciales, las relaciones y teorías de dominios de interés. Ontologías de tareas, que incluyen estructuras, partes, actividades y pasos para la solución de problemas. Ontologías para la estrategia de la enseñanza, que incluyen instructores y actores para modelar experiencias de la enseñanza, especifican el conocimiento las acciones pedagógicas y los comportamientos. Ontologías de modelo de aprendizaje, que permiten construir modelos para representar escenarios de aprendizaje adaptativos. Ontologías de interfaz, que permiten especificar el comportamiento adaptativo y las técnicas en el nivel de interfaz del usuario.

OntoUSA es una mezcla de ontologías de dominio y de tareas, pues incluye conceptos relacionados y teorías, pero también tienen estructuras y pasos que se deben seguir para solucionar problemas.

Figura 1. Jerarquía de clases

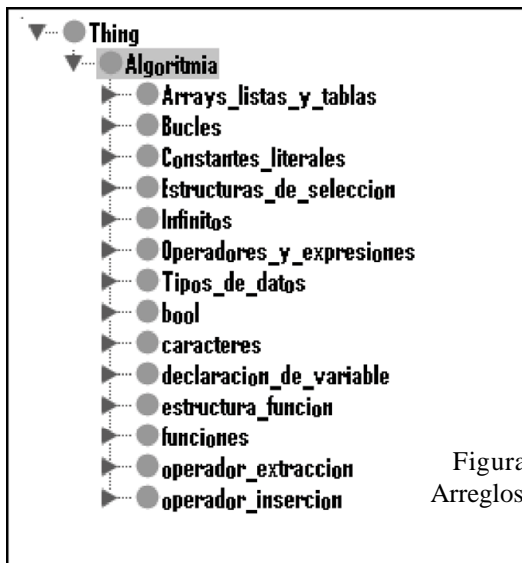


Figura 5. Vista parcial del tema Arreglos

VI. JERARQUÍA DE CLASES DE ONTOUSA

En la figura 1, se muestra parte de la jerarquía de clases que se definió en la ontología para representar los conceptos generales del dominio de la algoritmia básica.

La figura 1 representa los temas generales que componen la asignatura de algoritmos que se imparten en las diferentes universidades escogidas para realizar el estudio. Se aprecian los niveles más generales de la clase. El proceso seguido para conseguir la jerarquía presentada fue, top-down [6]; se inició con la definición de los conceptos generales en el dominio de la asignatura algoritmos, como son los temas presentados en la figura y luego se continuó con la creación de los conceptos particulares o subclases.

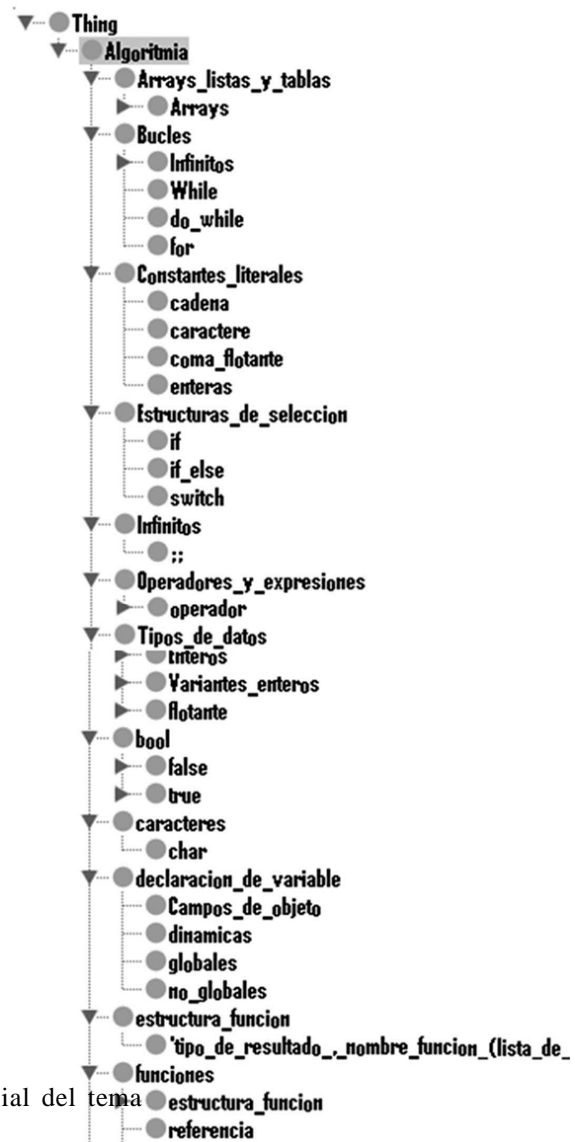


Figura 2. Instancias particulares

En la figura 2, se enseñan las instancias particulares del concepto (subclases), es decir, la descomposición entre los diferentes niveles de generalidad; muestra parte de los subtemas que componen los temas generales y las relaciones que existen entre ellos. Para ello, se seleccionaron los términos que describen los objetos y que existen independientemente de los términos que describen esos objetos. Se organizaron las clases jerárquicamente y se revisaron si las instancias de las clases serían o no instancias de otra. Luego de verificar el modelo, se generó el presentado.

En la figura 3, se muestran las relaciones entre clases e instancias; es decir se visualizan las jerarquías. Se observa de manera gráfica los diferentes niveles de la taxonomía de Algoritmos, como son los Arreglos\_listas\_y\_tablas, bucles, constantes\_literales, estructuras\_de\_selección, etc., que son los conceptos más generales y se presentan el lado izquierdo de la figura, pero en el tercer nivel. Se señala también las clases específicas que en el nivel jerárquico están el cuarto y quinto nivel, y representan las clases más específicas en la jerarquía.

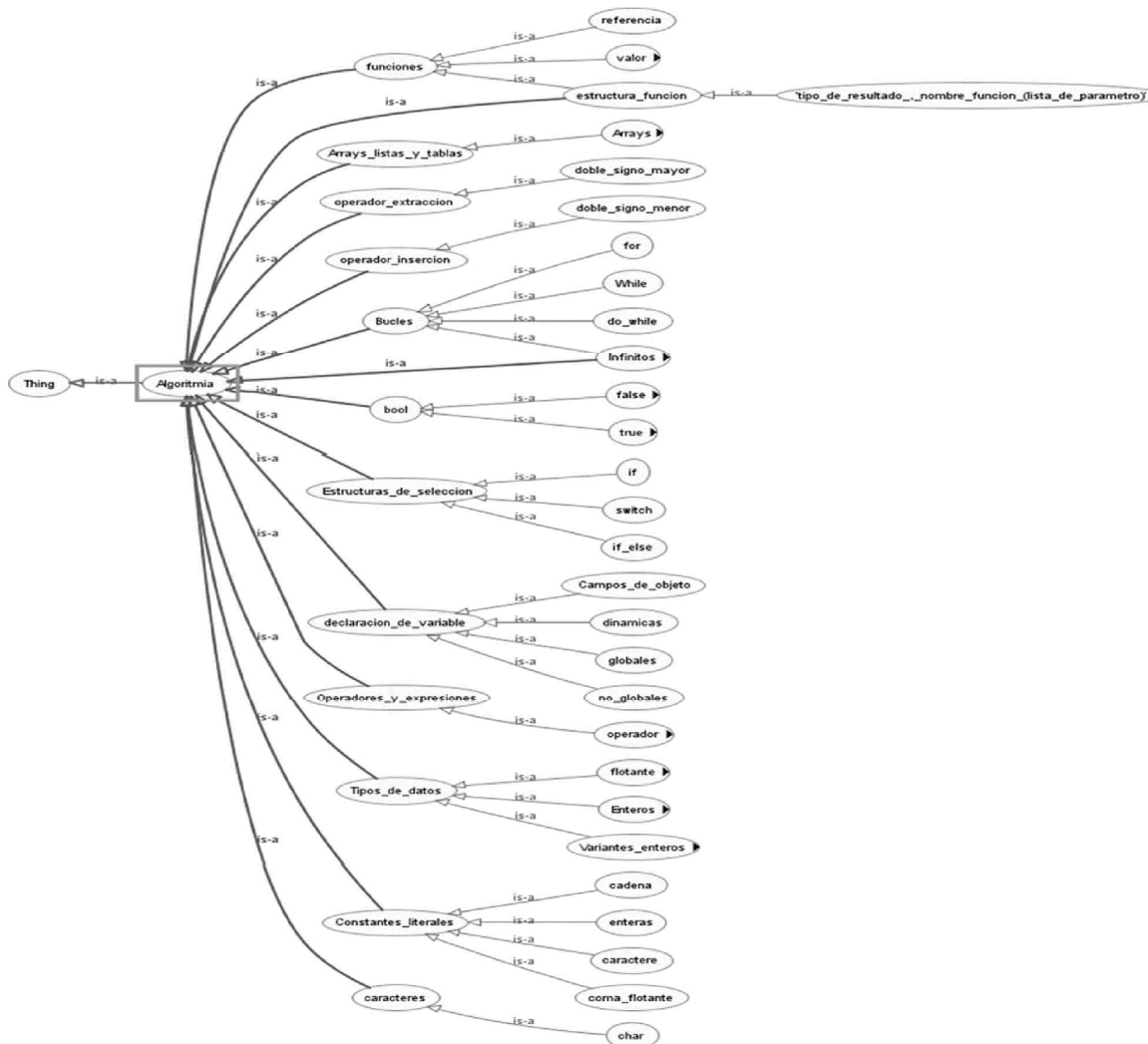


Figura 3. Vista general de OntoUSA

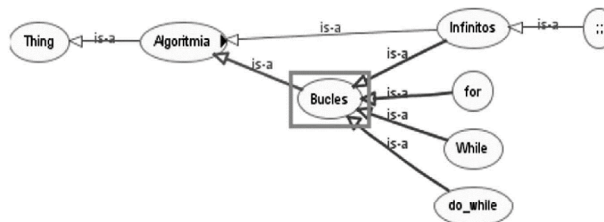


Figura 4. Vista parcial del tema Bucles



Figura 5. Vista parcial del tema Arreglos

En las figuras 4 y 5, se presentan vistas parciales de algunos temas que conforman OntoUSA. Se aprecia la jerarquía de las clases [7], junto con las asociaciones que entre ellas existe .

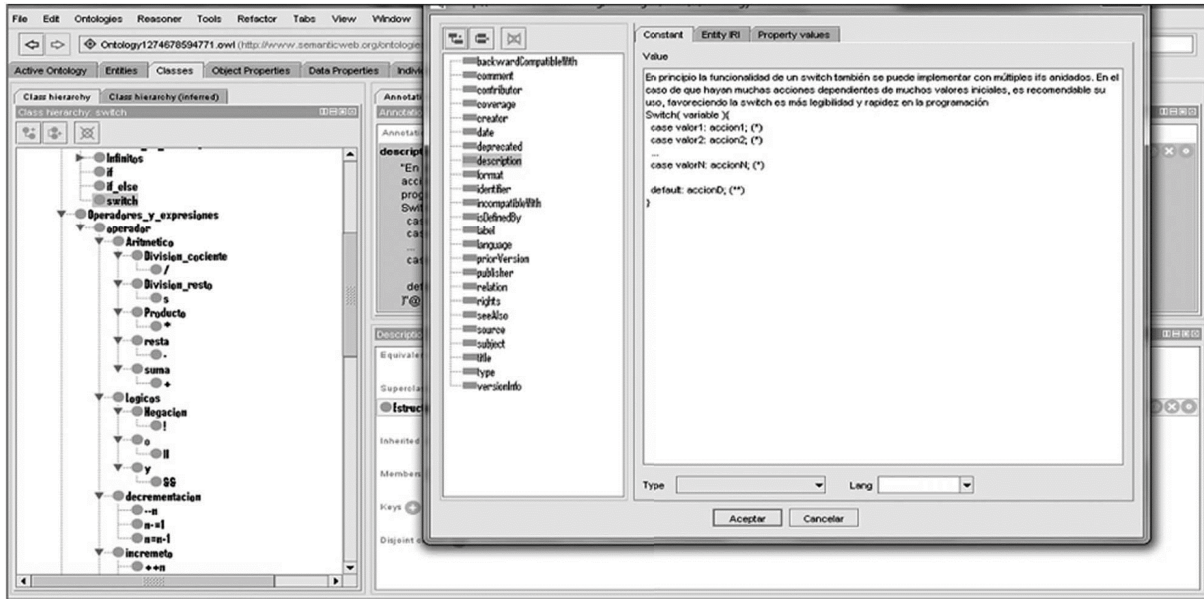


Figura 6. Vista de la descripción de la subclase

```

</SubClassOf>
- <SubClassOf>
  <Class IRI="#II"/>
  <Class IRI="#o"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#Infinitos"/>
  <Class IRI="#Algoritmia"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#Infinitos"/>
  <Class IRI="#Bucles"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#Logicos"/>
  <Class IRI="#operador"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#Multi_dimensional"/>
  <Class IRI="#Arrays"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#bool"/>
  <Class IRI="#Algoritmia"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#cadena"/>
  <Class IRI="#Constantes_literales"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#caractere"/>
  <Class IRI="#Constantes_literales"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#caracteres"/>
  <Class IRI="#Algoritmia"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#cero"/>
  <Class IRI="#false"/>
</SubClassOf>
- <SubClassOf>
  <Class IRI="#char"/>
  
```

Figura 7. Fragmento de código XML generado

En la figura 6, se aprecia un ejemplo de la descripción que se hace de una subclase, que para el ejemplo se tomó la instrucción switch. El procedimiento mostrado en la figura, fue el que se siguió para el resto de subclases.

En la figura 7 se enseña un fragmento del lenguaje generado en código XML de la ontología. En ella se observa la jerarquización que en código se da las clases y subclases.

## VII. CONCLUSIONES Y FUTURAS INVESTIGACIONES

Luego de describir a grandes rasgos el desarrollo de OntoUSA, se puede indicar que no son muchas las ontologías que tocan el tema de la algoritmia como en ésta se ha tratado. Se mostraron la jerarquía de clases que componen la ontología, así como la descripción de algunas de ellas, la gráfica que les corresponde y fragmentos de código en XML.

Se hizo una lista de las herramientas que se utilizaron en el desarrollo de la ontología, así como de la metodología que se utilizó para el desarrollo específico de la misma.

Las potenciales aplicaciones que se le dará a la ontología serán netamente educativas; es así como OntoUSA se considera una ontología educacional, debido a que será utilizada en la enseñanza de los algoritmos basada en tecnologías web.

En las pruebas preliminares la ontología funciona correctamente, pero no se sabrá si OntoUSA es buena hasta que no experimente con el Sistema Multiagente; investigación que ya se ha iniciado y que abrirá la puerta para realizar nuevos desarrollos sobre temáticas educativas; tampoco se podrá evaluar la calidad de ésta hasta que no se utilice.

## BIBLIOGRAFIA

- [1] Boolos G., Burgess J. and Jeffrey R., 2008. Logic, Logic, and Logic, Cambridge, MA: Harvard University Press.
- [2] Corcho M., 2001. Methodologies tools and languages for building ontologies. Where is their meeting point? Universidad Politecnica de Madrid, Campus de Montegancedo s/n, Madrid 28660, España. En: [http://www.aegean.gr/culturaltec/Kavakli/MIS/papers/Corcho\\_2003.pdf](http://www.aegean.gr/culturaltec/Kavakli/MIS/papers/Corcho_2003.pdf). Recuperado el 15 de septiembre de 2010.
- [3] Devedziz V., 2006. Semantic Web and education. Springer's Integrated Series in Information Systems.
- [4] Guarino N., Giaretta P., 1995. Ontologies and knowledge bases: towards a terminological clarification. Trento, Roma: Laboratory for Applied Ontology. En: <http://www.loa-cnr.it/Papers/KBKS95.pdf>. Recuperado el 15 de agosto 2010.
- [5] Guarino N., 1998. Formal ontology and information systems. Trento, Roma: Laboratory for Applied Ontology (LOA). En: <http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/FOIS98.pdf>. Recuperado el 3 de agosto de 2010.
- [6] Stanford University. 2004. Logic and Ontology. En: <http://mally.stanford.edu/>. Recuperado el 18 de septiembre de 2010.

- [7] Swartout B., et al. 2008. Toward distributed use of large-scale ontologies. Calgary: University of Calgary. En: [http://ksi.cpsc.ucalgary.ca/KAW/KAW96/swartout/Banff\\_96\\_final\\_2.html](http://ksi.cpsc.ucalgary.ca/KAW/KAW96/swartout/Banff_96_final_2.html). Recuperado el 15 de septiembre de 2010.

