

# Aplicación de la Lógica de Predicados para la Realización de Inferencias en Documentos de Pasantía Descritos a Nivel Ontológico

**Germán Alejandro Pinilla García**

Estudiante Ingeniería de Sistemas  
Universidad Distrital Francisco José de Caldas  
germanalp@hotmial.com

(Tipo de Artículo: Investigación Ciencia y Tecnología. Recibido el 16/10/2015. Aprobado el 22/06/2016)

**Resumen.** El número de usuarios que en la actualidad se apropia de prácticas de consulta mediante sistemas documentales cada vez es mayor, en este documento se hizo una representación a través de la lógica de predicados, aplicada a una ontología que describe trabajos de grado en modalidad pasantía, con el fin de realizar inferencias que permitan obtener información implícita a partir de las descripciones semánticas de los documentos. Se presentaron algunos componentes de la ontología, su representación en OWL (Ontology Web Language), la notación algebraica de las reglas obtenidas para representar la información mostrada en la ontología y finalmente las inferencias obtenidas. Lo anterior permite que los repositorios puedan ser utilizados por herramientas de software en diferentes instituciones, complementando la información explícita que proporcionan con información implícita pertinente a dirección, evaluación y temáticas de consulta en documentos de pasantía.

**Palabras clave.** Inferencia, Lógica de predicados, Ontología, Pasantía, Web Semántica.

## Application Predicate Logic for the Conduct of Documents Inferences Internship Described Ontological Level

**Abstract.** *The number of users currently appropriate practices through consultation document systems is increasing. This was a proposal of a representation by predicate logic, applied to an ontology that describes the internship as a form of degree work, in order to make inferences that allow to obtain implicit information from semantic descriptions of documents. Some components of ontology representation in OWL (Ontology Web Language), the algebraic notation of rules obtained were shown to represent the information shown in the ontology and finally the inferences drawn. This allows repositories could be used by software tools in different institutions, complementing the explicit information provided with implicit information relevant to the direction, evaluation and consulted issues of internship documents.*

**Keywords.** *Inference, predicate logic, ontology, internship, semantic web.*

### 1. Introducción

En las últimas décadas, el acceso a los trabajos de grado, a nivel internacional, se ha visto inmerso en un nuevo y complejo entorno competitivo, lo que ha llevado a un continuo proceso de búsqueda de nuevas formas de obtener y mantener esta ventaja, destacándose propuestas como la base de datos de Tesis Españolas Ordenadas [1] implementada por el Ministerio de Educación en España, mediante el registro de campos de datos sistémicos, que son aprobados por un comité. Este tipo de propuestas da a conocer los documentos que contengan únicamente la(s) palabra(s) ingresada(s) de manera textual, omitiendo varios documentos que puedan ser de importancia para el usuario. Cabe resaltar que no todos los documentos que resultan de la búsqueda pueden ser accedidos de manera gratuita. En Latinoamérica, una de las iniciativas más destacadas es Cybertesis [2], la cual tiene como objetivo la difusión de tesis electrónicas de instituciones de Europa, África y América. Igualmente en Colombia se destacan iniciativas como la promovida por la Universidad Nacional, que cuenta con una base de datos conocida como Sistema Nacional de Bibliotecas [3], la cual permite consultar documentos como revistas o trabajos de grado propuestos en todas las sedes nacionales. La Universidad Industrial de Santander planteó una iniciativa desde el año 2009 para la sistematización de trabajos de grado, ingresando los datos principales de cada uno. Este nuevo imperativo estratégico emerge como una importante consideración fundamental

empresarial, distinguiéndose como iniciativas que se apoyan en bases de datos relacionales y documentales para almacenar información referente a los trabajos de grado, permitiendo consultas únicamente de tipo sintáctico. A nivel semántico, la literatura de servicios está centralizada como una propuesta donde se plantea el lenguaje SPARQL como lenguaje de consulta [4]. De este modo han sido varios los autores que han destacado la consulta de una manera más amplia para la percepción de los consumidores.

### 2. Fundamentos teóricos

#### 2.1 Ontología

Las ontologías son una herramienta flexible en la representación del conocimiento, definidas como “un conjunto de términos de conocimiento, que incluye un vocabulario, relaciones y un conjunto de reglas lógicas y de inferencia sobre un dominio en particular” [5]. Desde las ciencias de la computación, se define como “an explicit specification of a conceptualization” [6]. Una ontología se compone de elementos como las clases o conceptos del dominio, propiedades que se representan mediante relaciones o atributos, también consta de individuos o instancias de las mismas clases y axiomas los cuales son restricciones determinadas para uno o más elementos de la ontología. Para conformarla es importante el lenguaje de representación formalizado reflejado en la organización de las clases en forma jerárquica, con el objetivo de aplicar la relación clase-subclase, establecer una aproximación multimodal con

propiedades de su superclase. Es así como se identifica una ontología específica a un grupo de individuos [7]. Finalmente, se señala a la ontología, como una especificación compartida de un dominio de conocimiento que permite la comunicación entre personas y sistemas heterogéneos [8]. Por otro lado los componentes de la atmosfera de la ontología han mostrado la representación computacional en el contexto OWL, el cual es un lenguaje desarrollado por el W3C Web Ontology Working Group. Se creó como una extensión de Resource Description Framework (RDF) y se utiliza en el desarrollo de ontologías y el uso compartido por la web [9], permitiendo especificar las características como instancias y sus interrelaciones.

## 2.2 Lógicas descriptivas

Son un conjunto de representaciones semi-formales, las cuales componen un conjunto de formalismos de representación de dominios que usan la lógica para la representación del conocimiento [10]. Las lógicas descriptivas pueden ser aplicadas a nivel abstracto evaluando las relaciones conceptuales y a nivel concreto procesando las instancias disponibles, dando lugar a:

1 T-box:

Para la descripción de conceptos jerárquicos o propiedades generales de los conceptos

2 A-box:

Para establecer la relación de cada una de las instancias de los conceptos con su ubicación en la jerarquía [11].

Cada una de las afirmaciones está representada por predicados que tienen un valor de verdadero o falso [12]. Esta lógica tiene un alto grado de expresividad, permitiendo así representar el dominio de los trabajos de grado de una universidad. Entiéndase la información como una colección de metadatos categorizada como información semántica o instructiva [13]. La lógica de predicados de orden cero está limitada a afirmaciones o proposiciones que tienen un valor de verdad definido. Sin embargo, carece de estructuras como cuantificadores o variables que representen entidades, siendo imposible representar afirmaciones como "Todos los hombres son mortales" o "Aristóteles es un hombre". De esta forma se optó para este trabajo el uso de la lógica de predicados de primer orden, la cual permite investigar la estructura interna de las variables de tipo proposicional y tiene una semántica definida acorde con lo requerido. En consecuencia y como objetivo dominante se utiliza la representación del conocimiento mediante las lógicas descriptivas como un sub campo de investigación dentro de la inteligencia artificial [14].

## 2.3 Lógica de predicados de primer orden

La lógica de predicados de primer orden es un lenguaje formal que hace uso de elementos como constantes, variables, funciones y cuantificadores para representar hechos, los cuales forman el alfabeto de este lenguaje [15]. Los cuantificadores son símbolos matemáticos que especifican condiciones cuantitativas

de un determinado conjunto indicando cuantos elementos poseen una propiedad. Esta lógica permite realizar deducciones mediante razonamientos de tipo  $\{P_1, P_2, P_3, \dots\} \rightarrow q$ , teniendo validez cuando la conclusión obtenida es una consecuencia de tipo lógico obtenida de las premisas evaluadas, las cuales están a la vez estructuradas por conectores como la negación, disyunción, implicación y bicondicional. Estos conectores permiten representar afirmaciones mediante predicados, donde cada uno puede tener uno o más parámetros para representar la información como pueden ser las premisas  $A(x)$  o también  $B(X_1, X_2, \dots, X_n)$  [16]. Como parte atómica de esta lógica se encuentran las clases, las cuales representan conceptos o entidades que se utilizan en la ontología. Éstas son representadas mediante instancias o elementos individuales de una clase en particular. Los predicados que reúnen 2 o más elementos son conocidos como relaciones y finalmente se encuentran los axiomas formales, los cuales son modelamientos de afirmaciones que se consideran siempre verdaderas y que son utilizadas para validar ciertos predicados y a la vez realizar deducciones.

## 3. Metodología

### 3.1 Materiales y Métodos

Las herramientas usadas para la realización de este trabajo incluyen el lenguaje OWL y la versión de software Protege 5.0 con la cual se construyó la ontología junto con el razonador Pellet para verificar su consistencia. Para la realización de la ontología, se eligió la metodología propuesta por Natalya y Deborah [7] debido a que es sencilla para la realización de este tipo de modelos. La metodología de este documento se conforma de los siguientes apartados: La sección 3.2 presenta la descripción de los conceptos importantes de la ontología. En la sección 3.3 se hace uso del lenguaje OWL para representar la ontología presentada en la sección 3.2. En la sección 3.4 se utiliza la lógica de predicados de primer orden para representar las reglas correspondientes a la ontología. Finalmente, la sección 4 presenta las inferencias obtenidas a partir de las reglas.

### 3.2 Planteamiento de la ontología

La ontología propuesta representa de manera organizada las características principales de modalidades de grado como lo son la monografía y pasantía. Esto con el fin de permitir una adecuada gestión semántica de la información referente a éstas. Para la obtención de este modelo se siguieron los pasos presentados a continuación.

#### 3.2.1 Enumeración de los conceptos importantes de la ontología.

Entre los conceptos elegidos para la representación de la ontología están los siguientes:

1 Trabajo de grado: Documento que elabora un estudiante para finalizar su proceso de formación de pregrado.

2 Estudiante: Persona que realiza el proyecto de grado.

3 Tema: Área de conocimiento al que pertenece un trabajo de grado

4 Jurado: Persona que califica la sustentación final de un proyecto de grado.

5 Director: Persona que guía a un estudiante para realizar su trabajo de grado.

6 Revisor: Persona que emite un concepto de tipo evaluativo sobre cada uno de los trabajos de grado.

### 3.2.2 Definición del modelo ontológico

El modelo ontológico está representado por las relaciones entre los conceptos definidos en el apartado 3.2.1. Las relaciones establecidas fueron las siguientes:

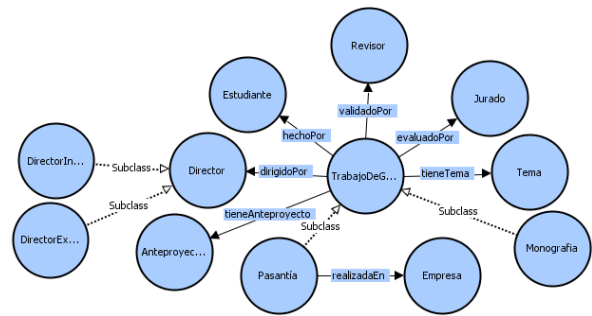
- 1 Trabajo de grado-Estudiante: Hecho por
- 2 Trabajo de grado-Tema: Tiene tema
- 3 Trabajo de grado-Revisor: Validado por
- 4 Trabajo de grado-Jurado: Evaluado por
- 5 Trabajo de grado-Director: Dirigido por
- 6 Trabajo de grado-Pasantía: Subclase
- 7 Trabajo de grado-Monografía: Subclase
- 8 Pasantía-Empresa: Realizada en
- 9 Director-Director interno: Subclase
- 10 Director-Director externo: Subclase

Donde la relación Subclase indica que el primer concepto tiene como subclase el segundo concepto. La Tabla 1 explica las relaciones entre las entidades utilizadas, así como la cardinalidad de cada una de éstas.

**Tabla 1. Dominio, rango y cardinalidad de las relaciones en la ontología. Fuente: autores.**

Relación	Dominio	Rango	Cardinalidad
Tiene tema	Trabajo de grado	Tema	Simple
Hecho por	Trabajo de grado	Estudiante	Múltiple
Validado por	Trabajo de grado	Revisor	Múltiple
Evaluado por	Trabajo de grado	de Jurado	Múltiple
Dirigido por	Trabajo de grado	Director	Múltiple
Realizada en	Pasantía	Empresa	Simple

La ontología utilizada para el desarrollo de este trabajo se presenta mediante un modelo en forma de grafo. Los nodos como *Estudiante*, *Director* y *Jurado*, representan clases, las cuales pueden ser referenciadas por instancias, mientras que los arcos como *hechoPor*, *validadoPor* y *evaluadoPor* representan relaciones que hacen explícitas las asociaciones entre conceptos, como se presenta en la Figura 1.



**Figura 1. Ontología sobre trabajos de grado**

### 3.3 Representación de la ontología en owl.

Se utilizó la versión FULL de este lenguaje, debido a que es la versión más general que se ha creado [9]. La ontología expresada anteriormente puede ser representada con la sintaxis de OWL-FULL como se presenta a continuación:

La representación descriptiva de una ontología puede realizarse desde la perspectiva de clases y sus relaciones, así como desde la especificación de restricciones como clases disjuntas y cardinalidades entre las relaciones. Esto es posible a través del lenguaje OWL como se presenta a continuación:

#### 3.3.1 Clases

Se utilizaron clases como trabajo de grado, monografía, pasantía, estudiante, tema, revisor, jurado, empresa, director, director interno y director externo. La representación general es de la siguiente forma:

```
Class(: TrabajoDeGrado)
Class(: Tema)
Class(: DirectorInterno)
```

#### 3.3.2 Individuos

Estos son instancias de las clases y se representan así:

1. Individual( :TrabajoDeGrado :1 )
2. Individual( :Tema :IngenieriaDeSoftware )
3. Individual( :DirectorInterno :MaryRodríguez)

#### 3.3.3 Propiedades

Éstas son las relaciones entre las clases. Cada una se representa mediante la estructura *ObjectProperty*, donde se especifica su dominio (domain) y su rango (range).

Se tomaron las siguientes relaciones como las más relevantes:

- 1.ObjectProperty(: tieneTema  
domain (: TrabajoDeGrado)  
range (: Tema))
- 2.ObjectProperty(: dirigidoPor  
domain (: TrabajoDeGrado)  
range (: Director))

3.ObjectProperty(: hechopor  
domain (: TrabajoDeGrado)  
range (: Estudiante))

4.ObjectProperty(: validadopor  
domain (: TrabajoDeGrado)  
range (: Revisor))

### 3.3.4 Subclases

Se representan con la estructura *subClassOf*. Se establecieron las siguientes:

1. SubClassOf( :TrabajoDeGrado :Pasantía )
2. SubClassOf( :TrabajoDeGrado :Monografía )
3. SubClassOf( :Director :DirectorInterno )
4. SubClassOf( :Director :DirectorExterno )

### 3.3.5 Relación 'uno de'

Son enumeradores considerados como tipos de datos antes de ser considerados clases. Algunos ejemplos que se tuvieron en cuenta son los siguientes:

1. Class( : Tema  
OneOf( :Testing :InteligenciaArtificial :Control ))
2. Class( : Director  
OneOf( : MaryRodríguez :DiegoBuitrago ))

### 3.3.6 Clases disjuntas

Algunas instancias solo pueden pertenecer a una única clase entre varias por ser mutuamente excluyentes. Como ejemplo se puede decir que un director únicamente puede ser interno o externo, por lo tanto una opción excluye a las demás, como se muestra a continuación:

disjointWith (:DirectorInterno :DirectorExterno)

Donde cada director existente es de un solo tipo como puede ser interno o externo de manera excluyente.

### 3.3.7 Cardinalidad

OWL permite establecer las cardinalidades entre las relaciones, permitiendo dar el número mínimo *minCardinality*, máximo (*maxCardinality*) o la cantidad exacta (*cardinality*) de las instancias utilizadas por cada relación. La estructura base es de la forma (*cantidad, relación, entidad rango*). Algunas de éstas se representan de la siguiente manera:

1. cardinality( 1 :TieneTema :TrabajoDeGrado )
2. mincardinality(1 :DirigidoPor :Monografía )
3. maxcardinality(3 :HechoPor :TrabajoDeGrado )
4. mincardinality(1 :ValidadoPor :TrabajoDeGrado )

El uso de herramientas como éstas permite tener una idea más clara de la ontología construida, así como brindar la facilidad de uso adecuado en caso de que se desee ampliar o modificar de acuerdo a las condiciones impuestas por cada institución universitaria a nivel

administrativo, buscando siempre mejorar los procesos de manejo de la información, debido a que en el futuro existirá una cantidad incalculable de esta y se necesitará agilidad en su gestión [17].

## 3.4 Planteamiento de la lógica de predicados

Dado que existen diferentes modalidades de trabajos de grado, en la primera parte se presentan las características generales del trabajo de grado y la segunda y tercera se centran en los procesos de iniciación y finalización de la modalidad como pasantía, respectivamente.

### 3.4.1 Reglas generales.

Se consultaron algunas universidades para obtener información generalizada sobre los trabajos de grado. Las reglas construidas en notación algebraica fueron las siguientes:

1 Todo trabajo de grado es dirigido por un director.

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists y/ (\text{Director}(y) \wedge \text{EsDirectorDe}(y,x)) \text{ [18]}$$

2 Todo trabajo de grado tiene un tema específico

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists ! t/ (\text{Tema}(t) \wedge \text{EsTemaDe}(t,x)) \text{ [18]}$$

3 Todo trabajo de grado debe tener al menos un estudiante.

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists y/ (\text{Estudiante}(y) \wedge \text{EsTrabajoDeGradoDe}(x,y)) \text{ [19]}$$

4 Un estudiante tiene un único trabajo de grado.

$$\forall x(\text{Estudiante}(x)) \rightarrow \exists ! t/(\text{TrabajoDeGrado}(t) \wedge \text{EsTrabajoDeGradoDe}(t,x))$$

5 Un director no puede ser jurado ni revisor del mismo trabajo.

$$\forall x,y (\text{Director}(x) \wedge \text{TrabajoDeGrado}(y) \wedge \text{DirectorDe}(x,y) \wedge (\text{EsJuradoDe}(x,z) \vee \text{EsRevisorDe}(x,z))) \Rightarrow y \neq z$$

6 Todo trabajo de grado en modalidad pasantía debe tener director interno y externo.

$$\forall x (\text{TrabajoDeGrado}(x) \wedge \text{Pasantía}(x)) \rightarrow \exists y,z /(\text{DirectorInterno}(y) \wedge \text{DirectorExterno}(z) \wedge \text{EsDirectorInternoDe}(y,x) \wedge \text{EsDirectorExternoDe}(z,x)) \text{ [20]}$$

7 Todo anteproyecto tiene al menos un revisor.

$$\forall a(\text{Anteproyecto}(a) \rightarrow (\exists r/ \text{Revisor}(r) \wedge \text{EsRevisorDe}(r,a)))$$

8 Un revisor puede ser recomendado para ser jurado

$$\forall r/(\text{Revisor}(r)) \rightarrow \exists r/\text{PuedeSerJurado}(r)$$

Las actividades institucionales para la realización de un proyecto de fin de carrera en modalidad pasantía implican: oficialización, desarrollo y cierre. En el caso de pasantía las acciones de estas actividades se organizaron en dos procesos: iniciación y finalización. Se utilizaron tres estados para determinar la situación de la pasantía, explicados en la Tabla 2.

**Tabla 2. Estados de una pasantía según el proceso.**  
Fuente: autores.

Nombre	Descripción	Posibles estados
Oficialización	Aprobación iniciación pasantía.	Viable, modificable o no viable.
Desarrollo	Una vez finalizada y radicado el documento de pasantía, los jurados emiten concepto.	Modificable o sustentable.
Cierre	Estado de la socialización de la pasantía.	Sustentado.

### 3.4.2 Reglas de la iniciación de la pasantía.

Estas reglas describen detalladamente el proceso de iniciación de esta modalidad de grado.

- Se debe realizar en una única empresa [21].  

$$\forall x(\text{Pasantía}(x)) \rightarrow \exists! y/ (\text{Empresa}(y) \wedge \text{EmpresaDePasantía}(y,x))$$
- Una pasantía debe tener tema y alcance [21].  

$$\forall x(\text{Pasantía}(x)) \rightarrow \exists y,z/ (\text{EsTemaDe}(y, x) \wedge \text{AlcanceDe}(z,x))$$
- El estudiante debe redactar anteproyecto de la pasantía [21].  

$$\forall x,y (\text{Estudiante}(x) \wedge \text{Pasantía}(y)) \text{EsTrabajoDeGradoDe}(y,x) \rightarrow \exists a(\text{AnteproyectoDe}(a, x))$$
- Se debe radicar el anteproyecto en el consejo curricular [21].  

$$\forall x(\text{Pasantía}(x)) \rightarrow \exists y/ (\text{AnteproyectoDe}(y,x) \wedge \text{Radicación}(\text{'ConsejoCurricular'}, y, \text{fecha}))$$
- Una vez radicado, se asignan 2 revisores [21].  

$$\forall x, y(\text{Pasantía}(x) \wedge \text{Anteproyectode}(y,x)) \rightarrow \exists w,z/ (\text{RevisorDe}(w,x) \wedge (\text{RevisorDe}(z,x) \wedge \neg \text{Iguale}(w,z)))$$
- Una vez asignados revisores, tienen 15 días para notificar el resultado al consejo curricular [21].  

$$\forall x(\text{Anteproyecto}(x) \wedge \text{Radicación}(\text{'ConsejoCurricular'}, x, \text{fecha})) \rightarrow \exists \text{fecha2}/(\text{Notificación}(\text{'ConsejoCurricular'}, x, \text{fecha2}) \wedge \text{fecha2} \leq \text{fecha1} + 15)$$
- El resultado puede ser viable, modificable o no viable [21].  

$$\forall x(\text{Anteproyecto}(x) \wedge \text{Notificación}(\text{'ConsejoCurricular'}, x, f)) \rightarrow \exists x/(\text{Oficialización}(x, \text{'viable'}) \vee \text{Oficialización}(x, \text{'modificable'}) \vee \text{Oficialización}(x, \text{'noViable'}))$$
- Si es viable, el consejo curricular notifica a la oficina de pasantías, dando la fecha de aprobación [21].  

$$\forall x(\text{Anteproyecto}(x) \wedge \text{Oficialización}(x, \text{'viable'})) \rightarrow \exists x(\text{Notificación}(\text{'OficinaDePasantías'}, x, f))$$
- Luego de aprobada, se dice que la pasantía ha sido iniciada oficialmente [21].  

$$\forall x,y(\text{AnteproyectoDe}(x,y) \wedge \text{Notificación}(\text{'OficinaDePasantías'}, x, f)) \rightarrow \text{Iniciada}(y, f)$$

### Unificación de predicados:

Algunos predicados fueron unificados, mediante la adición de más parámetros al predicado. Esto con el objetivo de simplificar y facilitar el entendimiento de las reglas. Por ejemplo en la regla: “Se debe radicar el anteproyecto en el consejo curricular”, inicialmente se planteaba su representación de la siguiente forma:

lugarRadicación('ConsejoCurricular',y)  $\wedge$   
 fechaRadicación(y,fecha)  $\wedge$   
 lugarRadicación('ConsejoCurricular', fecha)

Donde la primera parte corresponde a la existencia de la radicación del anteproyecto y en el consejo curricular. La segunda parte corresponde a la verificación que el anteproyecto y tiene una fecha de radicación denotada como fecha. La tercera parte corresponde a la existencia de una radicación en el Consejo Curricular en la fecha encontrada anteriormente. Una manera más sencilla y entendible de representación se presenta en la siguiente sentencia:

Radicación('ConsejoCurricular', y, fecha)

En este caso se verifica la existencia de una radicación en el Consejo Curricular, donde se radicó el documento y correspondiente al anteproyecto y que se realizó en una fecha denotada como *fecha*, la cual puede ser o no pre-establecida al momento de buscar el hecho. Se realizó igualmente para el predicado de tipo Notificación(x, y, f), donde x corresponde al lugar donde se realizó la notificación, 'y' corresponde al documento enviado o el trabajo correspondiente a la notificación y finalmente f corresponde a la fecha del trámite realizado.

### 3.4.3 Reglas de la finalización de la pasantía.

Estas reglas explican la finalización de la modalidad pasantía.

1 El director interno, debe hacer una visita a la empresa a los dos meses del inicio de la pasantía [21].

$$\forall x(\text{Pasantía}(x) \wedge \text{Iniciada}(x,f)) \rightarrow \exists d, f_2/(\text{Director}(d) \wedge \text{EsDirectorDe}(d,x) \wedge \text{Visita}(d, x, f_2) \wedge f_2 \geq f + 60 \wedge f_2 \leq f + 80)$$

Para esta regla y la siguiente se utilizó un rango de 20 días.

2 El director interno, debe hacer una visita a la empresa a los 4 meses del inicio de la pasantía [21].

$$\forall x(\text{Pasantía}(x) \wedge \text{Iniciada}(x,f)) \rightarrow \exists d, f_2/(\text{Director}(d) \wedge \text{EsDirectorDe}(d,x) \wedge \text{Visita}(d, x, f_2) \wedge f_2 \geq f + 120 \wedge f_2 \leq f + 140)$$

3 Existirá un acta por cada visita, llenando el formato de la pasantía, firmada por ambos directores [21].

$$\forall x(\text{Pasantía}(x) \wedge \text{Visita}(d, x, f)) \rightarrow \exists a/(\text{Acta}(a) \wedge \text{ActaDe}(a, x) \wedge \text{ActaFecha}(a,f))$$

4 Luego de firmar estas actas, se da como finalizada la pasantía [21].

$$\forall x(\text{Pasantía}(x) \wedge \text{ActaDe}(a, x) \wedge \text{ActaFecha}(a, f_1) \wedge \text{ActaFecha}(a, f_2) \wedge \neg \text{Iguale}(f_1, f_2)) \rightarrow \text{Finalizada}(x, f_2)$$

5 Se deben radicar las 2 actas de las visitas en la oficina de pasantías [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f)) \rightarrow \exists a, b / (\text{Acta}(a) \wedge \text{Acta}(b) \wedge \text{ActaDe}(a,x) \wedge \text{ActaDe}(b,x) \wedge \text{Radicación}(\text{'OficinaDePasantías'}, a, f_2) \wedge \text{Radicación}(\text{'OficinaDePasantías'}, b, f_3) \wedge f_2 \geq f \wedge f_3 \geq f \wedge \neg \text{Iguale}(a,b))$

6 La oficina de pasantías debe enviar la información al consejo curricular [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f)) \rightarrow \exists f_2 / (\text{Notificación}(\text{'ConsejoCurricular'}, x, f_2) \wedge f_2 \geq f)$

7 Luego se asignan 2 jurados calificadores [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f)) \rightarrow \exists w, z / (\text{JuradoDe}(w,x) \wedge (\text{JuradoDe}(z,x) \wedge \neg \text{Iguale}(w,z))$

8 El resultado puede ser modificable o sustentable [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f)) \rightarrow \exists x / (\text{Desarrollo}(x, \text{'modificable'}) \vee \text{Desarrollo}(x, \text{'sustentable'}))$

9 Si es sustentable, los jurados, directores internos y estudiantes deben acordar fecha de sustentación [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f) \wedge \text{Desarrollo}(x, \text{'sustentable'})) \rightarrow \exists f_2 (\text{Sustentación}(x, f_2) \wedge f_2 \geq f)$

10 A la sustentación los jurados le asignan una nota de 0 a 5 [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Finalizada}(x,f) \wedge \text{Cierre}(x, \text{'sustentado'})) \rightarrow \exists y (\text{NotaDePasantía}(y, x) \wedge y \geq 0 \wedge y \leq 5)$

11 A un acta se le asigna la nota respectiva [21].

$\forall x(\text{Pasantía}(x) \wedge \text{Cierre}(x, \text{'sustentado'})) \rightarrow \exists a, y / (\text{NotaDePasantía}(y, x) \wedge \text{Acta}(a) \wedge \text{TieneNota}(a, y))$

Como se puede apreciar, mediante la lógica de predicados es posible representar cada una de las expresiones obtenidas inicialmente, dado que esta lógica permite dividir las oraciones en sujetos, verbos e incluso adjetivos [22]. De estas reglas es posible utilizar algunas con el objetivo de realizar inferencias y obtener información que no está explícitamente representada mediante ellas. Durante la obtención de reglas siempre se debe buscar expresar claramente los detalles de los procesos que se deben realizar durante cada actividad como lo son los trabajos de grado. Esto es muy importante porque muchas veces se omiten algunas características u objetivos no tan relevantes, llegando algunas veces al punto de hacer algunas suposiciones sobre estos y afectar las expectativas y requisitos de los interesados.

#### 4. Resultados y discusión.

Los resultados obtenidos en este trabajo sugieren una serie de puntualizaciones que caracterizan un sistema de innovación mediante el uso de las reglas obtenidas anteriormente, las cuales pueden ser realizadas con inferencias, es decir, deducir nueva información o hechos partiendo de una base de hechos verdaderos. A continuación, se darán algunos ejemplos de inferencias que pueden ser obtenidas mediante la lógica de predicados. En las primeras tres inferencias se dan las premisas usadas y la conclusión obtenida. Las

siguientes presentan el problema principal a solucionar y su respectiva representación. Todas las inferencias son expresadas en lenguaje natural y notación algebraica.

#### - Inferencia sobre la relación Estudiante-Director.

Las premisas y conclusión obtenida fueron:

P1: Un estudiante tiene un único trabajo de grado.

P2: Todo trabajo de grado es dirigido por un director.

C: Todo estudiante es guiado por un director.

En notación algebraica quedarían expresadas así:

$P_1: \forall x(\text{Estudiante}(x)) \rightarrow \exists! t / (\text{TrabajoDeGrado}(t) \wedge \text{EsTrabajoDeGradoDe}(t,x))$

$P_2: \forall t / \text{TrabajoDeGrado}(t) \rightarrow \exists d / (\text{Director}(d) \wedge \text{EsDirigidoPor}(t, d))$

C:  $\forall x(\text{Estudiante}(x)) \rightarrow \text{EsDirigidoPor}(x, d)$

A través de esta inferencia, es posible generar información sobre posibles errores cometidos al momento de registrar directores o estudiantes y las relaciones que existen entre un director y cada uno de los estudiantes.

#### - Inferencia sobre la relación Estudiante-Tema.

Las premisas y conclusión obtenida fueron:

P1: Un estudiante tiene un único trabajo de grado.

P2: Todo trabajo de grado tiene un tema específico.

C: Todo estudiante debe tener un tema específico de interés.

En notación algebraica quedarían expresadas así:

$P_1: \forall x(\text{Estudiante}(x)) \rightarrow \exists! t / (\text{TrabajoDeGrado}(t) \wedge \text{EsTrabajoDeGradoDe}(t,x))$

$P_2: \forall t / \text{TrabajoDeGrado}(t) \rightarrow \text{TieneTema}(t, j)$

C:  $\forall x(\text{Estudiante}(x)) \rightarrow \text{TieneTema}(x, j)$

Con esta inferencia se verifica que por cada estudiante debe existir por lo menos un tema correspondiente a él y es posible encontrar cualquier inconsistencia al momento del registro de la información.

#### - Inferencia sobre la relación Anteproyecto-Posible jurado.

Las premisas y conclusión obtenida fueron:

P1: Todo anteproyecto tiene al menos un revisor.

P2: Un revisor puede ser recomendado para ser jurado.

C: Todo anteproyecto genera candidatos para ser jurado.

En notación algebraica quedarían expresadas así:

$P_1: \forall a(\text{Anteproyecto}(a)) \rightarrow \exists r / \text{Revisor}(r) \wedge \text{EsRevisorDe}(r,a)$

$P_2: \forall r / (\text{Revisor}(r)) \rightarrow \exists r / \text{PuedeSerJurado}(r)$

C:  $\forall a(\text{Anteproyecto}(a)) \rightarrow \exists r/\text{PuedeSerJurado}(r)$

Esta inferencia permite generar información de posibles jurados de posteriores trabajos de grado, según los temas abordados por el director en cada trabajo de grado dirigido.

**- Inferencia aplicada a la búsqueda de documentos de un tema específico.**

Muchas veces cuando un estudiante accede a algún repositorio de documentos de grado, este coloca el tema de su interés para conocer los avances y trabajos ya realizados que pueda tener en cuenta. Sin embargo, muchos de estos filtros de búsqueda tienen en cuenta el contenido introducido de manera textual al momento de elegir qué trabajos mostrar como resultado de la búsqueda, obviando hechos como lo son el que un tema puede estar directa o indirectamente relacionado con el tema de búsqueda y que también pueden ser trabajos útiles para que sean tenidos en cuenta por el estudiante, como se presenta en la Tabla 3:

**Tabla 3. Representación de temas y temas afines mediante variables. Fuente: autores**

Tema	Temas relacionados
$x_1$	$y_1, y_2 \dots y_n$
$x_2$	$y_1, y_2 \dots y_n$
$x_n$	$y_1, y_2 \dots y_n$

Donde cada termino  $x_i$  corresponde al nombre de cada uno de los temas generales de la carrera, mientras que cada termino  $y_i$  corresponde a los temas con los que están relacionados el tema  $x_i$  de la columna de la izquierda.

Una vez construida la tabla anterior, por cada tema  $x_i$ , se generan una serie de hechos de la forma  $\text{TemaRelacionado}(x_i, y_i)$ , obteniendo todas las relaciones entre cada uno de los temas de cada carrera, para luego obtener la inferencia correspondiente, como ejemplo se puede ver la Tabla 4.

**Tabla 4. Ejemplo real de temas y sus relaciones. Fuente: autores.**

Tema	Temas relacionados
1. Pruebas de software	Testing, validación, verificación.
2. Inteligencia artificial	Robótica, control.

En este caso se relaciona el tema general de pruebas de software con los temas testing, validación y verificación, así como se relaciona el tema inteligencia artificial con los temas robótica y control.

Los hechos generados por la tabla serían los siguientes:

- TemaRelacionado(pruebasDeSoftware, testing)
- TemaRelacionado(pruebasDeSoftware, validación)
- TemaRelacionado(pruebasDeSoftware, verificación)
- TemaRelacionado(inteligenciaArtificial, robótica)
- TemaRelacionado(inteligenciaArtificial, control)

La inferencia obtenida para ser aplicada en la búsqueda de un documento es la siguiente:

$$\forall t(\text{Tema}(t) \wedge \text{Tema}(j) \wedge \text{Documento}(y) \wedge \text{EsTemaDe}(j,y) \wedge \text{TemaRelacionado}(t,j)) \rightarrow \text{TrabajoRelacionado}(y).$$

De esta forma al recibir como parámetro inicial un término específico, es posible obtener como resultado de búsqueda todos los trabajos que sean de un tema que haya sido considerado de relación con el tema inicialmente especificado.

Cabe resaltar que esta inferencia puede ser usada por un docente al momento de conocer el estado del arte sobre algún tema de investigación en el que desee profundizar así como por una universidad al momento de determinar si es adecuado llevar a cabo algún proyecto de tipo investigativo.

**- Inferencia aplicada a la asignación de director a un trabajo de grado**

Otra situación en la que se pueden utilizar estas inferencias es al momento de asignar un director a cada trabajo de grado cuando el (los) estudiante(s) lo solicite(n). De esta manera se agilizará el proceso de asignar cada uno de los directores a estos trabajos, obteniendo resultados más rápidos y exactos en cuanto a la pertinencia del docente asignado en el tema del trabajo de grado. Esta inferencia se obtuvo de la siguiente manera: se estableció que todo trabajo de grado tiene un tema base, expresado mediante el hecho  $\text{TieneTema}(x, t)$  siendo  $x$  un trabajo de grado y  $t$  un tema específico. De manera similar a la anterior inferencia, se buscó establecer una relación entre los temas de cada carrera, donde se obtuvo la relación que si un docente domina un tema  $x$ , es muy probable que posea conocimientos adecuados acerca de varios temas denotados como  $y_1, y_2 \dots y_n$  y éste sea un docente adecuado para ser asignado como director del trabajo de grado en cuestión. Esta relación se presenta en la Tabla 5.

**Tabla 5. Esquema de relaciones entre temas y temas afines. Fuente: autores**

Tema dominado	Temas dominados relacionados
$x_1$	$y_1, y_2 \dots y_n$
$x_2$	$y_1, y_2 \dots y_n$
$x_n$	$y_1, y_2 \dots y_n$

En este caso se utilizaron los mismos hechos de la inferencia anterior para obtener el resultado, donde un docente puede conocer otros temas si estos están relacionados directa o indirectamente con su área de interés principal  $x$ , mediante la fórmula  $\text{TemaRelacionado}(x,y)$  siendo  $y$  otro tema. Como ejemplo se puede tomar un documento en búsqueda de director sobre el tema de la validación de software, representado por  $\text{TieneTema}(x, \text{'validación'})$ . De esta manera, un docente que conozca el área de 'Pruebas de Software', puede ser candidato a dirigir el trabajo nombrado anteriormente. La inferencia obtenida para este caso fue la siguiente:

$$\forall t,j,x (\text{Tema}(t) \wedge \text{Tema}(j) \wedge \text{Documento}(x) \wedge \text{TieneTema}(x, t) \wedge \text{TemaRelacionado}(t,j) \wedge \text{Docente}(d) \wedge \text{DominaTema}(d,j)) \rightarrow \text{PuedeSerDirectorDe}(d,x).$$

Al igual que en el caso anterior, se tiene en cuenta inicialmente un tema específico  $t$ , correspondiente a un trabajo de grado, permitiendo obtener como resultado un docente que posiblemente domine el tema del trabajo de grado. Esta inferencia es aplicable una vez se haya verificado que no existen docentes con conocimientos en el tema específico del proyecto de grado, de lo contrario se asignaría directamente el docente encontrado con conocimientos en el tema específico buscado inicialmente.

Con esta inferencia es posible mejorar la posibilidad de acierto al momento de asignar un director con más correspondencia con el tema de cada trabajo de grado, donde muchas veces se hacen suposiciones en cuanto al conocimiento en diversas áreas por parte de los docentes. Igualmente se reducirían tiempos de asignación debido a que se elimina la necesidad de conocer el historial de trabajos dirigidos por cada docente así como consultar los dominios generales por cada docente de manera manual. En este caso se hace necesario únicamente la existencia de un repositorio con las áreas dominadas por cada docente, así como la especificación de las relaciones entre temas dominados que pueden existir.

#### 4.1 Discusión y trabajo futuro

Algunas de las reglas utilizadas pueden ser representadas de diferente forma a nivel algebraico, conservando la equivalencia semántica. En el caso de la regla: un estudiante tiene un único trabajo de grado, se hizo la representación utilizando un cuantificador de tipo universal y uno de tipo existencial único de la siguiente forma:

$$\forall x(\text{Estudiante}(x)) \rightarrow \exists!t(\text{TrabajoDeGrado}(t) \wedge \text{EsTrabajoDeGradoDe}(t,x))$$

De manera alterna, la regla anterior puede ser representada mediante el uso de un cuantificador universal junto con uno de tipo existencial así:

$$\forall x(\text{Estudiante}(x)) \rightarrow \exists t,u(\text{TrabajoDeGrado}(t) \wedge \text{TrabajoDeGrado}(u) \wedge \text{EsTrabajoDeGradoDe}(t,x) \wedge \text{EsTrabajoDeGradoDe}(u,x) \wedge \neg \text{Igual}(t,u))$$

Igualmente la regla del proceso de iniciación de la pasantía correspondiente a: se debe realizar en una única empresa, se representó mediante un cuantificador universal y uno de tipo existencial único de la siguiente forma:

$$\forall x(\text{Pasantía}(x)) \rightarrow \exists!y/ (\text{Empresa}(y) \wedge \text{EmpresaDePasantía}(y,x))$$

Esta regla también puede ser representada como la regla anterior mediante un cuantificador universal y uno de tipo existencial de la siguiente forma:

$$\forall x(\text{Pasantía}(x)) \rightarrow \exists p,q/(\text{Pasantía}(p) \wedge \text{Pasantía}(q) \wedge \text{EmpresaDePasantía}(p,x) \wedge \text{EmpresaDePasantía}(q,x) \wedge \neg \text{Igual}(p,q))$$

De esta manera, sería objeto de otro trabajo hacer un análisis detallado de algunas de las reglas expuestas, con el objetivo de encontrar la representación que sea óptima a nivel de procesamiento computacional en función de las operaciones a realizar como por ejemplo comparaciones y verificación de predicados, esto con el objetivo de simplificar las expresiones, siendo el proceso de creación de formas lógicas un paso entre el análisis de tipo sintáctico de las expresiones y la obtención de estructuras semánticas más expresivas y profundas.

#### 5. Conclusiones

La propuesta presentada es aplicable a otras modalidades de grado como la realización de un proyecto de investigación o un proyecto de emprendimiento, existiendo la posibilidad de seleccionar los conceptos más relevantes como lo son las temáticas y de esta manera obtener modelos adecuados para cada uno de los aspectos particulares a tener en cuenta. La lógica de predicados trabajada en repositorios federados como Teseo facilitaría a los investigadores una mejor localización de recursos, permitiéndoles realizar búsquedas con un mayor grado de pertinencia, mejorando la visibilidad de trabajos realizados anteriormente para ser utilizados en los proyectos de investigación. El uso de modelos basados en la programación declarativa en problemas de manejo de altos volúmenes de información es muy factible puesto que permite mejorar los resultados de actividades que facilitan en la institución la toma de decisiones durante el desarrollo de procesos como los trabajos de grado por parte de los estudiantes. Mediante el uso de lógicas descriptivas es posible controlar las situaciones imprevistas de las instituciones como lo es la variación de la información a través del tiempo, al momento de agregar o eliminar directores, estudiantes o temas de cada carrera que aparezcan. Esta variación implica la actualización de los repositorios y la depuración dinámica de la información existente.

#### 6. Agradecimientos

Se agradece de manera especial, los aportes generados por el Director de Investigación Msc. J. Barón Velandia, los cuales permitieron desarrollar este documento de manera organizada.

#### 7. Referencias

- [1] Teseo. Servicio de consultas, Disponible en: <https://www.educacion.gob.es/teseo/irGestionarConsulta.do>. 2015.
- [2] Cybertesis. Servicio de consultas. Disponible en: <http://cybertesis.unmsm.edu.pe/>. 2015
- [3] SINAB. Repositorio institucional. Bogotá. Disponible en: <http://www.sinab.unal.edu.co/>. 2015.
- [4] E. Abreo y C. Padilla, "Prototipo Para Recuperación Semántica De Información De Proyectos De Grado En Ingeniería De Sistemas Con Base En Un Repositorio Ontológico", *Redes de Ingeniería*, 2012.
- [5] J. Hendler, "Agents and the semantic web", *IEEE Intelligent Systems*, 2001.
- [6] T. Gruber, "Toward Principles for the Design of Ontologies Used for Knowledge Sharing", *International Journal of Human and Computer Studies*, 1995.



- [7] N. Noy, y D. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology.", en *Knowledge Systems Laboratory Stanford University*, 2000.
- [8] S. Márquez, "Ontologías", *La Web Semántica*, España, pp. 49-50, 2007.
- [9] S. Ramírez, Y. Alonso, V. Hernández, A. C. Arias y D. La Rosa, "Comparando UML y OWL en la representación del conocimiento: correspondencia sintáctica", *Revista Española de Innovación, Calidad e Ingeniería del Software*, 2010.
- [10] M. Tayal, M. Raghuvansh, y L. Malik, "Knowledge representation: predicate logic implementation using sentence-type for Natural Languages.", *Circuits, Power and Computing Technologies (ICCPCT)*, 2013.
- [11] I. Flores, "Introducción al Razonamiento Sobre Ontologías.", *Centro de Ingeniería de Software y Sistemas*, 2011.
- [12] D. Lorge, "Predicate logic for software engineering.", *IEEE Transactions On Software Engineering*, 2013.
- [13] L. Floridi, "Open problems in the philosophy of information.", *Blackwell Publishing Ltd*, 2004.
- [14] A. Ali y M. Khan, "Knowledge representation of urdu text using predicate logic.", *International Conference on Emerging Technologies*, 2010.
- [15] J. Mylopoulos y H. Levesque, "An overview of knowledge representation.", *ACM SIGAI*, 1983.
- [16] K. Onot, S. Kawanott, Y. F'ukazawat y T. Kadokurat, "A resolution method from predicate logic specification into executable code.", *IEEE Conference Publications*, 1992.
- [17] F. Usama, "From data mining to knowledge discovery in databases", *American Association for Artificial Intelligence*, 1996.
- [18] Universidad del Rosario. Reglamento sobre procesos de grado, 2005.
- [19] Universidad Industrial de Santander. Acuerdo No 240 de 2008, 2008
- [20] Universidad de Cundinamarca. Acuerdo N° 001 del 08 de abril de 2011, 2011
- [21] Universidad Distrital. "Procedimiento de iniciación de pasantía como modalidad de trabajo de grado". 2010. Disponible en: <http://www.udistrital.edu.co:8080/documents/47880/1b3595b8-8f1d-4ebc-9256-8295e26d9579>, 2010.
- [22] A. Ali y M. Khan, "Selecting Predicate Logic for Knowledge Representation by Comparative Study of Knowledge Representation Schemes", *International Conference on Emerging Technologies*, 2009.