



## Índice

Certificación.....	i
Cesión de derechos.....	ii
Autoría.....	iii
Agradecimiento.....	iv
Dedicatoria.....	v
Índice.....	1
Resumen.....	4
Introducción.....	5
1. BIBLIOTECAS DIGITALES .....	7
1.1    Generalidades .....	8
1.1.1    Concepto.....	8
1.1.2    Características.....	8
1.1.3    Tipos.....	9
1.1.4    Ventajas y Desventajas.....	10
1.2    Proyectos internacionales .....	11
1.3    Estructura.....	13
1.4    Funcionalidades de la Biblioteca Digital.....	17
1.5    Biblioteca digital de la UTPL .....	17
1.5.1    Servicios de Bases de Datos.....	17
1.5.2    Bases de datos.....	21
1.5.3    Repositorios .....	22
1.5.4    Bibliotecas Virtuales.....	23



1.5.5	Recursos Digitales .....	29
1.5.6	Requerimientos de la Biblioteca Virtual de la UTPL .....	30
2	ONTOLOGÍAS .....	32
2.1	Introducción .....	32
2.2	Concepto .....	32
2.3	Ontología o Taxonomía .....	34
2.4	Elementos .....	35
2.5	Tipos .....	36
2.6	Ventajas / Desventajas .....	38
2.7	Metodologías .....	39
2.8	Análisis de las metodologías .....	52
2.9	Habituales errores en taxonomías y posibles anomalías del diseño .....	57
2.9.1	Errores de Inconsistencia .....	57
2.9.2	Errores de completitud .....	58
2.9.3	Errores de redundancia .....	59
2.10	Diseño y Construcción .....	59
2.10.1	Criterios .....	60
2.10.2	Lenguajes para construir ontologías .....	61
2.10.3	Herramientas de diseño y construcción .....	64
2.10.4	Análisis del lenguaje y las herramientas de diseño y construcción ..	67
2.11	Evaluación de la calidad de la ontología .....	68
2.12	Método de evaluación OntoQA .....	69
3	DISEÑO Y CONSTRUCCIÓN DE LA ONTOLOGÍA PARA LA BIBLIOTECA VIRTUAL (BV) DE LA UTPL .....	77



3.1	Introducción .....	77
3.2	Metodología para el desarrollo.....	77
3.3	Aplicación de Methontology en el desarrollo.....	77
3.3.1	Especificación .....	77
3.3.2	Conceptualización .....	83
3.3.3	Formalización e Implementación.....	91
3.3.4	Plan de Pruebas .....	94
3.3.4.1	Pruebas de integración .....	94
3.3.4.2	Pruebas de validación .....	106
3.3.5	Aplicación de OntoQA .....	109
4	RECOMENDACIONES DE ENLACES DE LA BIBLIOTECA VIRTUAL .....	118
4.1	Introducción .....	118
4.2	Lenguaje de consultas empleado en la aplicación .....	118
4.3	Funcionamiento de la aplicación.....	119
4.4	Modelo de comunicación Ontología – EVA .....	121
4.4.1	Casos de uso .....	121
4.4.2	Integración del módulo de recomendaciones al EVA .....	123
4.4.3	Limitaciones de la solución .....	123
	DISCUSION .....	125
	CONCLUSIONES.....	129
	RECOMENDACIONES .....	131
	BIBLIOGRAFÍA .....	133
	ANEXOS .....	138



## **Resumen**

Cuando realizamos una búsqueda sobre algún tema en particular en la web, de la gran cantidad de información que se obtiene como resultado, algunos son de valiosa ayuda, y otros no son los esperados o es “información basura”.

Lo que se pretende en el presente proyecto, es desarrollar y construir una ontología para recomendar enlaces de la Biblioteca Virtual de la UTPL. Por medio de esta ontología se desea utilizar en gran medida, los diferentes recursos digitales (bases de datos, bibliotecas virtuales, revistas,...) que ofrece la biblioteca virtual.

El primer paso será realizar un análisis exhaustivo de cómo está estructurada la biblioteca virtual, además del perfil académico del estudiante, para que basados en éste, podamos recomendarle información relevante para su formación profesional. Para esto, se ha aplicado la metodología Methontology.

Una vez realizado dicho análisis, el siguiente paso será desarrollar y construir la ontología en el editor de ontologías llamado Protégé.

Como el editor Protégé nos permite desarrollar ontologías en varios formatos, optaremos por el tipo de archivo .rdf, con el fin de poder implementarla en el razonador lógico RDF API de Jena, conjuntamente con el lenguaje de programación php.

Por último, se adaptará la ontología en el EVA. Para esto se creará un bloque en dicho entorno. Además se harán las configuraciones respectivas para que la ontología pueda ser implementada sin ningún inconveniente.



## **Introducción**

El acelerado desarrollo tecnológico de los últimos años, ha incrementado las necesidades de conocimiento acerca del funcionamiento de las tecnologías de la información y cómo ponerla al servicio de la humanidad, y para ello la información debe pasar por un proceso de modelación la que consiste en seleccionar, organizar y brindar el acceso a dicha información a los usuarios.

Las potencialidades que brinda el Internet hoy en día son grandiosas, desde la comunicación entre usuarios hasta el acceso y descarga de información, es decir está evolucionando al pasar de un sistema estático y monótono a grandes y dinámicos sistemas con la interacción de la comunidad de beneficiarios.

En los últimos años el desarrollo de Bibliotecas Digitales, a consecuencia de la implementación de grandes redes de comunicación, información y conocimiento, ha contribuido al almacenamiento y recuperación de gran cantidad de información, lo que trae como consecuencia un mayor nivel de conocimiento que facilita la toma de decisiones tanto para profesionales en formación como organizaciones educativas e investigativas,...

Actualmente se hace necesario el desarrollar técnicas de recuperación de información de manera inteligente, que permitan al usuario el acceso a grandes cantidades de información almacenadas en las Bibliotecas Virtuales. En los últimos tiempos, con el desarrollo de la Web Semántica, se permite una recopilación y digitalización de metadatos (información), el desarrollo de ontologías para los metadatos (información) y creación de motores de búsqueda que puedan acceder a los datos digitales.

Teniendo en cuenta todo esto las ontologías son determinantes porque permiten la organización de la información, disminuyendo el tiempo e incrementando los resultados de los motores de búsqueda.



# CAPÍTULO I

## **BIBLIOTECAS DIGITALES**



## 1. BIBLIOTECAS DIGITALES

Desde hace mucho tiempo, la gran mayoría de la humanidad ha deseado poder almacenar y compartir toda clase de información, sin importar el cómo y el dónde se lo hará.

Con la llegada de la era digital, donde los computadores tienen una gran capacidad de almacenamiento y procesamiento, se podía pensar en ese momento en realizar cosas inimaginables. Y mucho más con la llegada de internet, se evidenció que esto era posible.

Con el pasar de los años se ha incrementado el uso del Internet, aumentando así la publicación de contenidos digitales a bajo costo y acceso rápido. Esto da lugar al surgimiento de Bibliotecas Digitales de todo tipo, que ofrecen a sus usuarios contenidos específicos y gran variedad de servicios.

Hoy en día las bibliotecas digitales son lugares virtuales, que poseen documentación de manera digital. Con este simple detalle, nos podemos dar cuenta que una biblioteca digital ofrece mayor cobertura que una biblioteca tradicional.

Se pueden mencionar algunos de los servicios que prestan las bibliotecas digitales, como son: préstamos en línea, solicitud en línea, reservas en línea, acceso a bases de datos que las hacen muy diferentes, y hasta cierto punto, es la ventaja principal, con respecto a las bibliotecas tradicionales. En este capítulo hablaremos como está constituida la biblioteca digital de la UTPL, los servicios que presta al personal de la universidad, así como también su entorno, ya que está conectada a diferentes bases de datos de todo el mundo, que abarcan toda clase de temas, por ejemplo: medicina, ingeniería civil, robótica, recursos naturales, agropecuaria,...



## **1.1 Generalidades**

En esta sección se presenta una introducción a las Bibliotecas Virtuales, en la que se especifica su concepto, características principales, tipos, ventajas y desventajas, a fin de tener una visión general del contexto en el que se desarrolla el presente proyecto.

### **1.1.1 Concepto**

Según López Guzmán citado en Faba C. & Nuño V. (n.f), una biblioteca virtual *“Es un repositorio de acervos y contenidos digitalizados, almacenados en diferentes formatos electrónicos por lo que el original en papel, en caso de existir, pierde supremacía. Generalmente, son bibliotecas pequeñas y especializadas, con colecciones limitadas a sólo algunos temas”*.

En base a lo expuesto, se puede decir que una biblioteca virtual es aquella que contiene información de manera digital y brinda a los usuarios una serie de servicios (lectura, descarga, impresión, búsqueda mediante catálogo o palabras claves,...), y se puede acceder a ellos vía Web desde cualquier lugar. La información digital puede incluir varios formatos: .pdf, .doc, .ppt, .html,... y se encuentra almacenada en servidores de bases de datos.

### **1.1.2 Características**

- Se puede acceder a bancos de información electrónica a través de búsqueda de información, consultas bibliográficas,...
- Sirve para que el usuario localice el documento en formato digital.





- Permite la automatización y eliminación de determinados procesos burocráticos como: registro al ingreso de la biblioteca, préstamos de material académico, fecha de devolución de libros,...
- La biblioteca es validada de acuerdo a la calidad de los contenidos, su organización, los servicios que presta y los sistemas de gestión que facilitan el acceso a los datos almacenados.
- Es un apoyo para los procesos educativos enseñanza – aprendizaje.
- La información está disponible de manera organizada y en constante renovación y enriquecimiento, lo que contribuye a impulsar la educación y la investigación científica.
- La biblioteca no tiene existencia física, ya que toda la información esta almacenada de forma digital en bases de datos.

### 1.1.3 Tipos

Según Taly S., (2007), de la Universidad Bar-Ilan de Israel, las bibliotecas digitales se pueden clasificar de la siguiente manera:

**Biblioteca Digital Autónoma (BDA).**- es la transformación de la biblioteca tradicional a una digital, mediante procesos de automatización de la información. Dicha información se puede encontrar almacenada en una sola base de datos.



**Biblioteca Digital Federada (BDF).**- es un grupo de varias Bibliotecas Digitales Autónomas, que se relacionan entre ellas debido a que comparten información común sobre una red, y presentan una interfaz única al usuario.

**Biblioteca Digital Recolectada (BDR).**- esta biblioteca proporciona un acceso resumido a diferentes temas que se encuentran dispersos en el mundo del Internet.

De estos 3 tipos de bibliotecas, se puede concluir que la Biblioteca Virtual que posee la UTPL, es una biblioteca digital recolectada, ya que brinda acceso a diferentes recursos digitales, bibliotecas virtuales y bases de datos científicas.

Actualmente forma parte de la iniciativa Liblink, en la que a través de la plataforma Celsius, comparte la información entre diferentes bibliotecas de otras universidades, por lo que también se la puede considerar como una biblioteca digital federada.

#### **1.1.4 Ventajas y Desventajas**

En general, una biblioteca virtual posee las siguientes ventajas:

- No hay horarios para consultas o retiros. Se visita en el momento en que se desee.
- Tienen acceso a ella todas las personas, aunque no puedan trasladarse a una biblioteca tradicional por problemas de diversa índole.
- No hay que retirar, trasladar y devolver los libros en una fecha determinada, pues en las bibliotecas tradicionales, son requeridos por otros lectores.



- A un mismo libro, pueden acceder un número ilimitado de usuarios.
- Los libros digitales no se estropean ni se desgastan.
- Se puede hablar, escuchar música o trabajar en grupo, mientras se consulta en la biblioteca digital.

Entre las principales desventajas se pueden citar:

- La mayor desventaja son las brechas generadas por las barreras económicas y sociales en el mundo real.
- No siempre se encuentran los textos completos en formato digital.

## 1.2 Proyectos internacionales

ISTEC<sup>1</sup> El Consorcio Iberoamericano para la Educación en Ciencia y Tecnología ha propuesto la iniciativa de un Enlace de Bibliotecas Virtuales (LibLink).

Según lo propuesto por Gómez L., (2008), LibLink pretende fomentar el conocimiento, mediante la capacitación, integración, e intercambio de material bibliográfico, y para ello cuenta con abundante material disponible. Es un servicio al que pueden acceder los usuarios miembros de esta iniciativa, y lo pueden solicitar por medio de un formulario electrónico.

A fin de conocer la situación actual de las bibliotecas latinoamericanas se ha desarrollado un estudio a las bibliotecas de las universidades públicas y privadas que están afiliadas a LibLink, en el que según comenta Mejía L., (2007) Colombia tiene el mayor número de universidades participantes. Los resultados se muestran en el siguiente gráfico:

---

<sup>1</sup> Consorcio Iberoamericano para la Educación en Ciencia y Tecnología <http://www.istec.org>

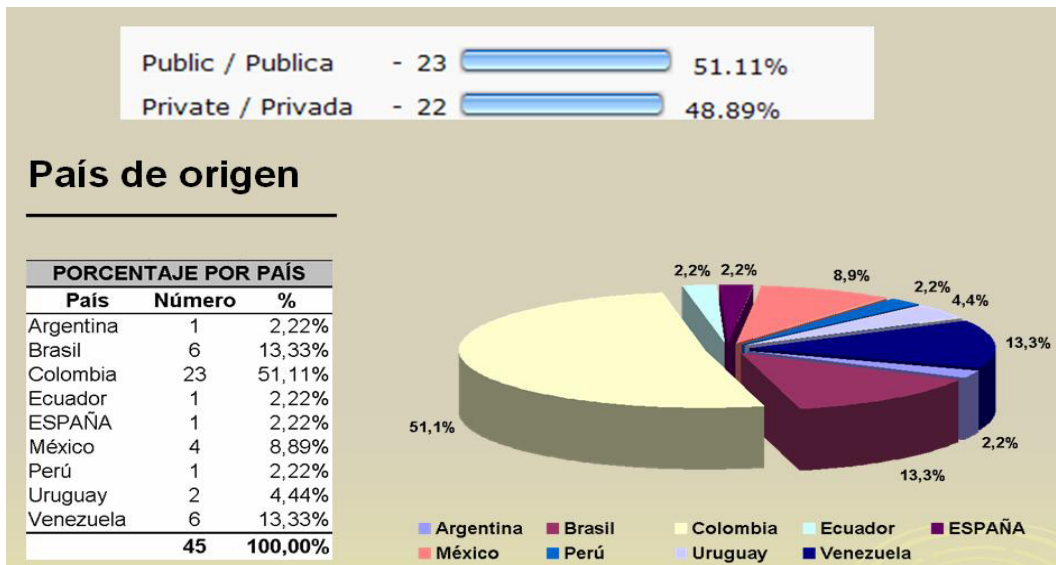


Figura 1.1. Resultados del Análisis<sup>2</sup>.

Entre algunos resultados que se obtuvieron fue la cantidad de áreas de especialidad por biblioteca, siendo el área de Administración y Economía la que poseía el mayor porcentaje de participación, seguida del área Ciencia y Tecnología y Ciencias de la Salud. Otro resultado también obtenido fue el número de documentos disponibles para la investigación, tomando en cuenta el número de estudiantes por universidad, donde las universidades que tienen entre 10.000 y 20.000 estudiantes participan más en esta iniciativa.

Actualmente la Universidad Técnica Particular de Loja (UTPL), es una de los centros de estudios superiores del Ecuador, que forma parte de esta iniciativa.

<sup>2</sup> Recuperado desde <http://liblink.istec.org/modules.php?name=Content&pa=showpage&pid=77>



### 1.3 Estructura

En la actualidad con el incremento de usuarios de bibliotecas digitales es indispensable que estas posean una arquitectura robusta, donde integren diferentes elementos que reflejen el esquema básico de una biblioteca digital. A continuación en la Fig.1.2 se presenta una arquitectura de la biblioteca digital.

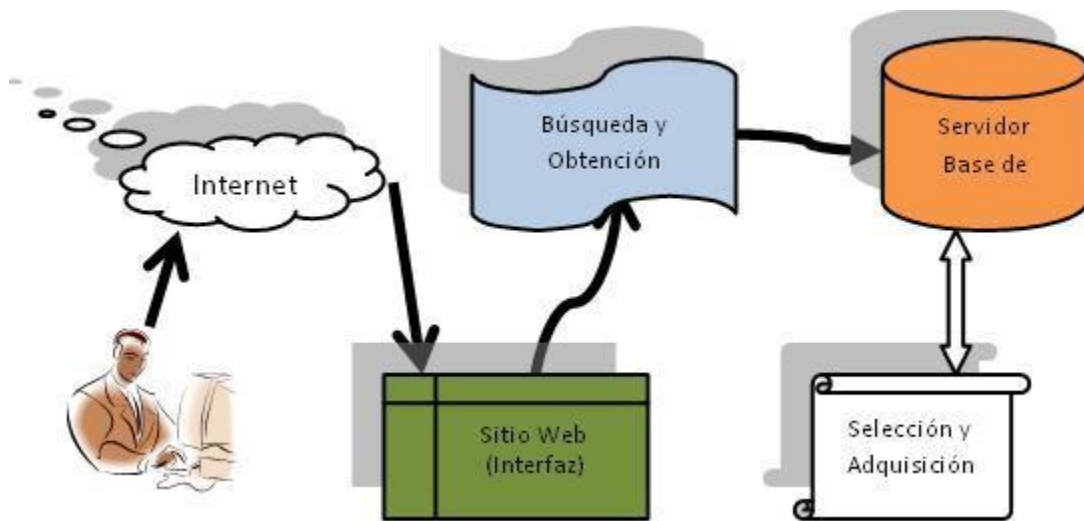


Figura 1.2. Esquema de una arquitectura básica de una biblioteca digital.

**Internet.-** el Internet provee de una interfaz amigable al usuario y por medio de este se accede desde cualquier lugar a una biblioteca virtual, y a través de este sitio Web se realiza la búsqueda.

**Búsqueda y obtención.-** este componente integra elementos para la búsqueda como motores de búsqueda y directorios de índice, así como también elementos para la obtención de información como las conocidas “herramientas de recuperación de información”.



La diferencia que existe entre un motor de búsqueda y un directorio de índice, es que los directorios sólo buscan resultados basándose en el título de la página y su descripción, mientras que un motor de búsqueda realiza sus búsquedas en bases de datos a través de palabras claves.

**Motores de búsqueda o buscadores.-** responden a las peticiones realizadas por el usuario sobre un determinado tema. Son los tradicionalmente conocidos como Google, Lycos, AltaVista,... Estos recorren la Web buscando una determinada palabra.

**Directorios de índices.-** son directorios estructurados jerárquicamente por categorías.

Existe una gran variedad de herramientas usadas y conocidas para realizar la recuperación de información en biblioteca virtual, entre las que se encuentra Apache Lucene<sup>3</sup>, Nutch<sup>4</sup>, Terrier<sup>5</sup>, Lemur<sup>6</sup>,... De éstas, las más utilizadas son Apache Lucene y Lemur.

**Apache Lucene.-** es un API<sup>7</sup> para recuperación de información de código libre desarrollado en Java. Este nos permite la búsqueda de archivos .doc, .pdf, .xls, .ppt, .rtf, .html, .txt, .xml,... Lucene tiene versiones para lenguajes Perl, C#, Ruby y C++, y en el proceso de búsqueda incluye operadores lógicos (AND, OR, NOT, +, -).

**Lemur.-** es un conjunto de herramientas (toolkit) de código libre desarrollado en lenguaje C y C++ para sistemas Linux aunque también corre en Windows, para facilitar el modelado del lenguaje y

<sup>3</sup> <http://lucene.apache.org/>

<sup>4</sup> <http://lucene.apache.org/nutch>

<sup>5</sup> <http://ir.dcs.gla.ac.uk/terrier>

<sup>6</sup> <http://www.lemurproject.org/>

<sup>7</sup> Application Programming Interface (Interfaz de Programación de Aplicaciones)



recuperación de información. Incluye tecnologías como recuperación distribuida, con preguntas estructuradas, resúmenes automáticas, filtrado y clasificación.

**Base de datos**<sup>8</sup>.- los servidores de bases de datos son los lugares donde está almacenada la información que posee la biblioteca virtual, y a los cuales se accede en forma directa a través de un conjunto de programas.

Existe una gran variedad de bases de datos utilizadas para almacenar información y servir de soporte para las bibliotecas virtuales, se puede mencionar las siguientes:

**PostgreSQL**<sup>9</sup>.- es un Sistema de Bases de Datos que fue desarrollado en 1977, y es de código abierto. Actualmente es una de las bases de datos más avanzada del mundo, que posee varias características como:

- **Alta concurrencia.**- es decir, permite el acceso a otros usuarios a la misma tabla sin necesidad de bloqueos.
- **Gran variedad de tipos nativos.**- es decir, posee soporte a números de precisión arbitraria, texto de largo ilimitado, figuras geométricas, direcciones IP versiones (IPv4<sup>10</sup> e IPv6<sup>11</sup>).
- **Un disparador o trigger.**- se da cuando ocurre una acción específica realizada por algún proceso o petición dentro de la base de datos.

<sup>8</sup> [http://es.wikipedia.org/wiki/Base\\_de\\_datos](http://es.wikipedia.org/wiki/Base_de_datos)

<sup>9</sup> <http://www.postgresql.cl/>

<sup>10</sup> Es la versión 4 del Protocolo de Internet (Internet Protocolo)

<sup>11</sup> Es la versión 6 del Protocolo de Internet (Internet Protocolo)



**Oracle**<sup>12</sup>.- es una herramienta cliente/servidor para la gestión de Bases de Datos, tiene una gran potencia aunque el costo es elevado.

Características:

- Soporte de Transacciones.
- Estabilidad.
- Escalabilidad.
- Multiplataforma.

**MySQL**<sup>13</sup>.- es un sistema de gestión de bases de datos relacional, bajo código libre, su sistema de diseño multihilo permite soportar una gran carga de forma muy eficiente.

Características

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP,...).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.

**Selección y Adquisición.**- el documento requerido, es seleccionado del lugar donde se encuentre, para presentarlo a través de la interfaz web.

---

<sup>12</sup> <http://www.oracle.com>

<sup>13</sup> <http://mysql.com/>





## **1.4 Funcionalidades de la Biblioteca Digital**

- Acceso a las bases de datos: la persona que desea realizar alguna tarea profesional, tiene a su disposición el acceso a diferentes bases de datos de cualquier parte del mundo con cualquier cantidad de información.
- Consulta de libros electrónicos (e-books), revistas, diccionarios, almanaques y otros recursos digitales.
- Servicios personalizados de cada usuario: si el usuario desea registrarse, la biblioteca digital le brinda servicios personalizados, tales como: informe de nueva bibliografía, noticias de diferente índole, anuncios,...

## **1.5 Biblioteca digital de la UTPL**

La Universidad Técnica Particular Loja (UTPL) dispone de una Biblioteca Virtual (BV), para la investigación durante la formación académica de estudiantes y apoyo a docentes investigadores. Esta cuenta con importantes enlaces como son: bases de datos, servicios de base de datos, recursos digitales, bibliotecas virtuales, catálogos de libros, solicitud de bibliografía y enlaces a otras fuentes de información.

### **1.5.1 Servicios de Bases de Datos**

Un servicio de Bases de Datos es un portal que incluye varias bases de datos en las que el usuario puede elegir de acuerdo a su interés la búsqueda de información.

La UTPL mantiene convenios con diferentes organismos que proveen estos servicios, en ellos se encuentra una infinidad de publicaciones, libros, revistas, documentos,... en diferentes formatos. Para realizar estas búsquedas se presentan diferentes opciones como son:



búsqueda básica, búsqueda avanzada, búsqueda por temas y búsqueda por publicación.

Las bases de datos pueden ser categorizadas como de acceso libre y de acceso restringido. A las de acceso libre (Opendoar, Doaj, Ohiolink Etd, J-Stage, Ifpri, CgvLibrary) se puede ingresar desde cualquier lugar, mientras que a las de acceso restringido (ProQuest, ESBCOHost, Infotrac, Informe, Hinari) se lo puede hacer solamente desde la red interna de la UTPL.

A continuación se describen los servicios de bases de datos disponibles en la biblioteca virtual de la UTPL:

#### **ProQuest<sup>14</sup>**

Este servicio ofrece 30 bases de datos que proporcionan acceso a más de 125 millones de páginas digitales a nivel mundial, posee recursos en múltiples áreas del conocimiento como: arte, literatura, ciencias sociales, tecnología, medicina.

#### **Esbcohost<sup>15</sup>**

EBSCO suministra acceso a información electrónica en diferentes áreas del conocimiento y cuenta con miles de revistas electrónicas. Sus bases de datos proporcionan información de diferentes áreas del conocimiento. Permite el acceso a seis bases de datos científicas y a una interfaz avanzada para realizar búsquedas.

---

<sup>14</sup> <http://proquest.umi.com/login/ipauto>

<sup>15</sup> <http://search.epnet.com/>



### **Gale**<sup>16</sup>

Informe de Thomson GALE, base de datos con material bibliográfico en diferentes áreas del conocimiento, se encuentra con una extensa publicación de más de 4.000 publicaciones en texto completo, recopiladas en investigaciones realizadas a nivel mundial.

Permite realizar una búsqueda en cuatro formatos, como son: búsqueda básica, búsqueda por guía de temas, búsqueda por publicación y búsqueda avanzada.

### **Informe Académico**<sup>17</sup>

Es una base de datos multidisciplinaria, diseñada para satisfacer la necesidad de conocimiento de los usuarios. Está compuesta por más de 240 publicaciones en texto completo en todas las disciplinas académicas. Contiene revistas y publicaciones realizadas por las más prestigiosas Universidades e Institutos de Investigación de Iberoamérica y Canadá, cuenta con 4.166.612 artículos con una reciente actualización.

### **Hinari**<sup>18</sup>

Ofrece acceso al mayor número de revistas de biomedicina y otros campos afines de forma semi-gratuita, a instituciones públicas de países en vías de desarrollo. Cuenta con 8 bases de datos relacionadas con temas de la salud de varias organizaciones. Posee más de 3.800 revistas que están disponibles para las instituciones en materia de salud. Cuenta con más de 113 países asociados, las búsquedas se puede realizar en diferentes bases de datos de

---

<sup>16</sup>[http://find.galegroup.com/ips/infomark.do?type=static&page=BasicSearch&prodId=IPS&version=1.0&userGroupName=utpl\\_cons&source=gale](http://find.galegroup.com/ips/infomark.do?type=static&page=BasicSearch&prodId=IPS&version=1.0&userGroupName=utpl_cons&source=gale)

<sup>17</sup>[http://find.galegroup.com/itx/infomark.do?type=static&page=BasicSearch&prodId=IFME&version=1.0&userGroupName=utpl\\_cons&source=gale](http://find.galegroup.com/itx/infomark.do?type=static&page=BasicSearch&prodId=IFME&version=1.0&userGroupName=utpl_cons&source=gale)

<sup>18</sup> <http://www.who.int/hinari/en/?lang=sp>



distintas regiones del mundo, en idiomas disponibles como: español, francés, portugués. Permite una personalización y configuración de búsquedas.

### **Wiley InterScience<sup>19</sup>**

Cuenta con 11 base de datos, tiene un número superior a más de 850 revistas. La mayoría de estos artículos y revistas son publicaciones realizadas por diferentes universidades y profesionales que aportan de manera voluntaria para crear e integrar un mundo basado en el conocimiento. Cuenta con información de varios temas relacionados con las ciencias, entre estos temas podemos citar medicina, ciencias sociales y humanidades.

### **Ohiolinketd Search<sup>20</sup>**

Es una base de datos que posee una amplia información sobre distintos tópicos relacionados con las ciencias del conocimiento, que son publicadas por universidades y colegios. Cuenta con acceso a disertaciones y tesis, 46 millones de libros y otros materiales en la biblioteca, millones de artículos electrónicos, 12.000 revistas electrónicas, acceso a 140 bases de datos electrónicas de investigación, 25.000 libros electrónicos, miles de imágenes, vídeos y sonidos, 13.750 tesis y disertaciones de estudiantes.

### **Cg Vlibrary<sup>21</sup>**

Cuenta con 134 bases de datos. Es un grupo consultivo para la investigación agrícola internacional, y mantiene una alianza con organizaciones públicas, privadas y fundaciones. Cuentan con 15 centros de investigación entre los que se encuentran universidades y

---

<sup>19</sup> <http://www.difusion.com.mx/wiley/>

<sup>20</sup> <http://www.ohiolink.edu/etd/search.cgi>

<sup>21</sup> <http://vlibrary.cgiar.org/>



revistas especializadas que trabajan para mejorar la calidad agrícola. Mantienen una publicación de 1.814 documentos que son el fruto de investigaciones realizadas, y se encuentran disponibles en diferentes formatos.

### 1.5.2 Bases de datos

Estas nos permiten realizar búsquedas de cualquier tipo, y no se puede elegir el área de conocimiento o base de datos específica, ya que presentan un portal común.

#### **Ifpri**<sup>22</sup>

IFPRI está orientado a proporcionar soluciones agropecuarias, ya que realiza investigaciones científicas para mejorar la calidad de productos agropecuarios en varios países. Cuenta con 58 miembros, principalmente los países subdesarrollados o en vías de desarrollo, con temas relacionados como la agricultura, la ganadería, la selvicultura, la pesca, la política, y los recursos naturales.

#### **J-Sage**<sup>23</sup>

Patrocinado por el organismo japonés de ciencia y tecnología, poseen revistas científicas en todas las áreas del conocimiento. Tiene actualmente 455 revistas (194.018 artículos), 112 actualizaciones, 10 informes y 42 informes de los JST disponible en versión inglés y japonés.

---

<sup>22</sup> <http://www.ifpri.org/spanish/pubs/spubs.htm>

<sup>23</sup> <http://www.jstage.jst.go.jp/browse>



### **Ebrary<sup>24</sup>**

Es una empresa dedicada a brindar servicio de documentación relacionada con, libros, revistas,... a bibliotecas digitales. Contiene información sobre diferentes áreas del conocimiento.

### **1.5.3 Repositorios**

Estos repositorios son una colección de recursos accesibles a través de una red digital; aquí se pueden incluir varios tipos de recursos con información variada sobre diferentes temas. Puede estar esa información organizada mediante metadatos que permiten describir dichos documentos electrónicos.

### **OpenDOAR<sup>25</sup>**

El OpenDOAR ofrece un servicio de acceso libre a una gran cantidad de información en diversas áreas del conocimiento. Para lograr este objetivo existen varios grupos de usuarios e investigadores, dispersos en todas las universidades del mundo los que aportan con tesis, maestrías e investigaciones realizadas. Dichos documentos están organizados por regiones o continentes, permitiéndole al usuario ubicar la información de acuerdo al lugar.

### **Doaj<sup>26</sup>**

Presta un servicio de revistas relacionados con temas científicos y académicos en texto completo, disponibles en diferentes idiomas, con acceso gratuito. Actualmente posee 3.117 revistas en su base de datos y 171.354 de artículos relacionados con todas las áreas del conocimiento como: matemáticas, ciencias sociales, economía, física

---

<sup>24</sup> <http://site.ebrary.com/lib/bibliotecautpls>

<sup>25</sup> <http://www.opendoar.org/countrylist.php>

<sup>26</sup> <http://www.doaj.org/>



astrología,... Mediante esto promueve un fuerte incremento del conocimiento de sus usuarios.

#### **1.5.4 Bibliotecas Virtuales**

Actualmente la UTPL propone a los estudiantes y docentes investigadores la utilización de otras bibliotecas virtuales, ya sean nacionales e internacionales, con información relacionada con diferentes ciencias, se pueden encontrar libros, revistas, publicaciones académicas,...

Entre ellas se tiene:

##### **Biblioteca Jurídica UTPL<sup>27</sup>**

Sirve de apoyo a estudiantes y docentes del área jurídica de la UTPL y al público en general. Se puede obtener información sobre códigos, leyes y reglamentos que rigen la constitución política del estado ecuatoriano.

##### **National Academy Press<sup>28</sup>**

Creada por el Consejo Nacional de Academias para publicar informes cada año, en diferentes áreas del conocimiento, cuenta con más de 200 libros en una amplia variedad de temas como: ciencia, ingeniería y salud que pueden ser descargados de forma gratuita.

##### **Biblioteca Virtual Miguel de Cervantes<sup>29</sup>**

Es una interesante biblioteca virtual en donde los amantes de la literatura y poesía pueden encontrar una variedad de obras escritas por reconocidos autores hispanoamericanos.

---

<sup>27</sup> <http://www.utpl.edu.ec/biblioteca/bibliotecaJuridica/>

<sup>28</sup> <http://www.nap.edu/>

<sup>29</sup> <http://www.cervantesvirtual.com/>



### **Congreso de los Estados Unidos<sup>30</sup>**

Cuenta con información acerca de personajes históricos y leyes del congreso de los EE.UU, posee una gran variedad de libros, que están disponibles para congresistas y ciudadanos en general.

### **Educar<sup>31</sup>**

Es un portal bajo el patrocinio del Estado Argentino, donde se encuentra una gran variedad de temas, educativos, foros, debates, noticias de actualidad, recomendado para usuarios de todas las edades, incluso niños.

### **Bibliotecas Universitarias y de Investigación Españolas<sup>32</sup>**

Es un portal del Consejo Superior de Investigaciones Científicas, posee una documentación amplia de todas las investigaciones realizadas, cuenta con bases de datos y publicaciones actualizadas de temas científicos, así mismo cuenta con una red de bibliotecas con información relevante en ciencias de la vida.

### **Biblioteca Pública de Ciencia<sup>33</sup>**

Es una organización sin fines de lucro y el acceso es libre. En esta biblioteca el estudiante y el profesor pueden encontrar información sobre prácticas o procesos médicos, e investigaciones realizadas recientemente.

---

<sup>30</sup> <http://www.loc.gov/index.html>

<sup>31</sup> <http://www.educ.ar/educar/site/educar/>

<sup>32</sup> <http://www.csic.es/cbic/otros/cat/webuni.html>

<sup>33</sup> <http://www.plos.org/>





### **Pensamiento Ecuatoriano**<sup>34</sup>

Es una organización sin fines de lucro que promueve el aprendizaje social individual y colectivo, por ello está comprometido con el desarrollo humano. Podemos encontrar información acerca de autores ecuatorianos sobre el tema educativo, tanto a nivel nacional como internacional.

### **Virtual Library**<sup>35</sup>

Posee información de varias partes del mundo en todos los ámbitos del conocimiento. Ideado y concebido por Tim Berners-Lee, creador de la Web semántica. En este portal la información se encuentra organizada por áreas.

### **Molecular and Cellular Biology**<sup>36</sup>

Portal financiado por la Universidad de Harvard. Abarca una amplia gama de temas que incluyen bioquímica, biofísica, genética, biología celular, inmunología molecular, así como neurobiología y biología estructural. Cuenta con gran información en sus bases de datos. Es interesante para estudiantes y docentes dedicados a investigaciones en ciencias biológicas.

### **Virtual Library Chemistry Section**<sup>37</sup>

Es un portal financiado por la Universidad de Liverpool, posee más de 1.000 enlaces a diferentes sitios Web, con información relacionada con la química.

---

<sup>34</sup> [http://www.fronesis.org/ecuador\\_bibvirtual.htm](http://www.fronesis.org/ecuador_bibvirtual.htm)

<sup>35</sup> <http://vlib.org/>

<sup>36</sup> <http://www.mcb.harvard.edu/BioLinks.html>

<sup>37</sup> <http://www.liv.ac.uk/Chemistry/Links/links.html>



### **Biblioteca Virtual de la Salud (BVS)**<sup>38</sup>

Este proyecto está conformado por los países sudamericanos y España, tiene a disposición material relacionado con la medicina natural y tradicional, en base a investigaciones realizadas por diferentes autores cubanos.

### **Biblioteca Virtual de Educación**<sup>39</sup>

Es un portal donde se puede encontrar información del ámbito educativo, de estudios realizados por varias instituciones brasileñas.

### **Biblioteca Virtual en Bioética**<sup>40</sup>

Cuenta con el patrocinio de diversos países sudamericanos y ofrece diversos recursos de información en bioética, a nivel nacional, regional e internacional.

### **Biblioteca Virtual Mundo Hispano**<sup>41</sup>

Biblioteca virtual que posee temas literarios hispanoamericanos, españoles y literatura universal de varios autores, incorpora libros de libre acceso y descargas.

### **Internet Public Library**<sup>42</sup>

Esta base de datos patrocinada por la Universidad de Michigan, contiene información en inglés sobre distintas áreas del conocimiento, como: educación, negocios, cultura, ya sea en papers, diccionarios,...

---

<sup>38</sup> <http://bvs.sld.cu/E/basebilingrafica.html>

<sup>39</sup> <http://bve.cibec.inep.gov.br/>

<sup>40</sup> <http://bioetica.bvsalud.org/html/es/home.html>

<sup>41</sup> <http://www.angelfire.com/pa/bibliooven/>

<sup>42</sup> <http://www.ipl.org/reading/books/>



### **Online Books<sup>43</sup>**

Esta biblioteca permite el acceso a libros en inglés los cuales tienen variada información como: historias, terror, romance, física, astrología,... Permite la lectura on-line, así como descargas y compras.

### **Voice of the Shuttle<sup>44</sup>**

Cuenta con variada información en inglés sobre diferentes áreas del conocimiento, como por ejemplo historia, arquitectura, danza, arqueología, negocios,...

### **Biblioteca Virtual Universal<sup>45</sup>**

Esta biblioteca posee información académica dirigida a todos los niveles de formación, cuenta con una hemeroteca, posee 16.129 obras digitalizadas, las que están disponibles para ser descargadas.

### **Comunidad Literaria y Textos en Línea<sup>46</sup>**

Posee documentos, libros, obras literarias, on-line, de América Central, Sudamérica.

### **Psicología Científica<sup>47</sup>**

Es una publicación que presenta en Internet trabajos de investigación en español como: artículos científicos, ensayos, reseñas y entrevistas en audio, relacionados con el desarrollo de las diferentes áreas de la psicología.

---

<sup>43</sup> <http://www.onlinebooks.com/>

<sup>44</sup> <http://www.ipl.org/reading/books/>

<sup>45</sup> <http://www.biblioteca.org.ar/>

<sup>46</sup> <http://www.bibliotecasvirtuales.com/>

<sup>47</sup> <http://www.psicologiacientifica.com/bv/index.html>



### **Biblioteca Dominico-Virtual<sup>48</sup>**

Creada para fomentar la educación, posee más de 1.000 páginas en diversas áreas del conocimiento, cuenta con documentos de literatura, historia, revistas, recursos biológicos, etc.

### **Biblioteca Virtual de Tecnología Educativa<sup>49</sup>**

Cuenta con información relacionada con la inclusión de las tecnologías en la educación superior a través de libros, publicaciones y tesis doctorales.

### **Mathematics WWW Virtual Library<sup>50</sup>**

Portal en inglés patrocinado por la Universidad de Florida, contiene varios recursos y publicaciones relacionados con las matemáticas.

### **Cátedra UNESCO de e-learning<sup>51</sup>**

Centro de Recursos digitales e-learning. En este portal Web se integran los contenidos y servicios relacionados con la gestión e-learning, permite profundizar en la innovación educativa.

### **Biblioteca Electrónica Cristiana<sup>52</sup>**

Biblioteca que cuenta con recursos digitales en diferentes formatos en el área de la Teología para profundizar los conocimientos teológicos, para ello dispone de recursos digitales en diferentes formatos. Portal en idioma inglés que contiene información en diversas áreas del conocimiento.

---

<sup>48</sup> <http://www.bibliotecavirtual.com.do/>

<sup>49</sup> <http://tecnologiaedu.us.es/bibliovir.htm>

<sup>50</sup> <http://www.math.fsu.edu/Virtual/>

<sup>51</sup> <http://cv.uoc.es/tren/trenacc/web/DIMAX.CLIENT/dimaxweb.Pagina?pantalla>

<sup>52</sup> <http://vlib.org/>



### 1.5.5 Recursos Digitales

En esta sección se ofrecen enlaces a recursos seleccionados de diferentes temáticas como:

- Ingeniería y tecnología (robótica, arquitectura, construcciones, diseño, agricultura y química,...).
- Arte (música, bailes, fotografía, diseño web, software musical, escultura).
- Medicina y salud (medicina, cuidado de la salud, enfermedades infecciosas, primeros auxilios).
- Humanidades (historia, cristianismo, estudios bíblicos, filosofía, ética social, religión).
- Ciencias de la Vida (agua, plantas, botánica, biotecnología, datos biológicos, genética, especies en extinción).
- Lengua y Literatura (poesía, idiomas, ciencia ficción, literatura).
- Matemática y Computación (álgebra, lenguajes de programación, bases de datos, programación, sistemas operativos, diseño web, algoritmos, comercio electrónico, diccionario de computación).
- Ciencias Sociales (leyes, política, historia y sociología).
- Ciencias Físicas (bioquímica, astrología, climatología, cartografía, física nuclear, energía nuclear, satélites, termodinámica, física, química).



### **1.5.6 Requerimientos de la Biblioteca Virtual de la UTPL**

- La existencia de gran cantidad de información, sin orden y categorización eficiente.
- Fuentes de información inutilizadas por falta de acceso de estudiantes y docentes investigadores.
- Al realizar consultas, las búsquedas son redundantes lo que ocasiona pérdida de tiempo.



# CAPÍTULO II

## **ONTOLOGÍAS**



## **2 ONTOLOGÍAS**

### **2.1 Introducción**

Ontología es una antigua materia que proviene del campo filosófico. A lo largo de la historia ha tenido diferentes definiciones, que contribuyeron a identificarla con la metafísica, que estudia la esencia de seres y las características básicas de la realidad como un todo.

Según lo expuesto por Blas L. y otros, (2008), etimológicamente proviene del griego “ontos”: ser y “logos”: tratado o estudio, por lo tanto, ontología se refiere al estudio del ser y sus propiedades esenciales.

Las ontologías actualmente son materia de investigación, desarrollo, y aplicación en temas relacionados con la computación y la representación del conocimiento.

Empiezan a ser utilizadas a finales de los 80s en el ámbito de la Inteligencia Artificial como una forma para compartir y reutilizar el conocimiento existente. Y es a mediados de los 90s donde se empieza a darle aplicabilidad en la Web.

Hoy en día ha sufrido un impulso debido al desarrollo de la Web Semántica, donde la idea principal es que la información almacenada en la web sea interpretada no solo por humanos sino también por un computador.

### **2.2 Concepto**

La primera definición de ontología en inteligencia artificial, fue aportada por Neches con su equipo de trabajo, citado por (García, 2004, p.2), quien define y delimita la ontología como “un instrumento que define los términos básicos y relaciones a partir del vocabulario de una área





específica, así como las reglas para combinar dichos términos y las relaciones para definir extensiones de vocabularios”.

En el transcurso del tiempo han ido surgiendo otras definiciones, tales como la de Gruber Tom (1995), que afirma que una ontología [(Traducción al español) es una especificación explícita de una conceptualización].

Sin embargo Borst (1997), citado en Martin María (2004 a), modificó ligeramente la definición agregando el término “compartido” a la definición de Gruber, así, define a las ontologías como “la especificación formal de una conceptualización compartida”.

El equipo compuesto por Studer, y otros (1998), citado en Martin María (2004 b), agregaron expresividad a los términos expuestos por Gruber y Borst explicando el sentido de conceptualización explícita, formal y compartida.

- **Es explícita** porque expresa conceptos, propiedades, relaciones, funciones,.... y restricciones mediante axiomas y reglas que conforman la ontología, permitiendo su razonamiento.
- **Es formal** porque está especificado en un lenguaje entendible por las máquinas.
- **Conceptualización** porque es la representación abstracta de un determinado conocimiento mediante la integración de los conceptos más relevantes.
- **Compartida** se refiere al conocimiento que estará en la ontología será de común acuerdo, es decir aceptado por el grupo.



De todas estas definiciones se puede concluir que una ontología es una representación o descripción formal, clara y precisa, de una conceptualización de un determinado dominio, entendiéndose por dominio el ámbito en el cual se va a representar la ontología, que se encuentra estructurada de manera lógica (modelado de las clases que conforman el dominio) y matemática (al empleo o uso de reglas de inferencia, para poder inferir conocimiento a la ontología).

### **2.3 Ontología o Taxonomía**

En forma general no está muy clara la diferencia que existe entre una ontología y una taxonomía, debido a que algunos expertos concuerdan en que una ontología no es más que una taxonomía, ya que describe una representación del conocimiento de manera poco profunda.

Para Tallarico M. (2008), una ontología difiere de una taxonomía, en que una ontología tiene una estructura interna mucho más explícita que una taxonomía, además de que refleja cierto consenso entre las clases que la conforman.

Las ontologías típicas que se pueden encontrar y que son utilizadas en la web, están conformadas por taxonomías conjuntamente con reglas de inferencia.

**Taxonomía.-** permite la organización y representación de clases, subclases, relaciones, permitiendo agregar propiedades a las clases, así como su respectiva herencia entre clase y subclase. Éstas son una poderosa forma de representación del conocimiento.



**Reglas de Inferencia.-** estas le añaden una capacidad adicional a las taxonomías, ya que les permiten generar conocimiento. Por lo general el formato de una regla es 'si', "entonces". Ej:

{Si mañana hace sol entonces no llueve}

## 2.4 Elementos

Las ontologías proporcionan la vía para representar el conocimiento de un determinado dominio. Según Gruber (1993) citado en Gómez Asunción (n.f.), propuso utilizar técnicas de IA, dentro de una ontología, y definió los siguientes elementos:

- **Conceptos.-** es lo que se intenta representar, algo tangible (objetos) o intangibles (función, acción).
- **Relaciones.-** son las interacciones y enlaces entre los conceptos del dominio. Las uniones que establecen son del tipo subclase-de, parte-de, conectado-a,...
- **Funciones.-** tipos especiales de relaciones donde se identifican elementos mediante el cálculo de una función que implica a varios elementos de la ontología. Ejemplo: Cantante canta Canción(es)  
Función: Cantidad\_canciones, de acuerdo al número de canciones que grabe.
- **Instancias.-** Son los elementos de una clase (conceptos).
- **Axiomas o reglas de inferencia.-** Son reglas sobre relaciones que deben cumplir las clases de una ontología.



## 2.5 Tipos

Existen variedad de clasificaciones de ontologías, algunos autores las categorizan de acuerdo a su alcance o aplicabilidad, otros como Sowa (2000), las clasifican entre ontologías terminológicas y formales.

Para Mizoguchi, y otros (1995), las ontologías se clasifican por dependencia de contexto, considera cuatro categorías: ontologías de contenido, de indexación, de comunicación y meta-ontologías.

Por otro lado, según Gómez-Pérez (1999), citado en Martín María (2004 a), las ontologías son clasificadas por el tipo de dominio que modelan o por sujeto de contextualización.

De acuerdo con el nivel de reutilización Heist Van y otros (1997) citado en Gómez Asunción (n.f.), las clasifican como: genéricas, de representación, de dominio, y ontologías de aplicación.

Atendiendo a todos estos criterios de clasificación podemos indicar los siguientes tipos de ontologías, definidos por Sowa, Mizoguchi, Gómez-Pérez y Heist Van:

- **Ontologías Terminológicas.-** permite simplicidad en la especificación del modelado. Definen los términos y sus relaciones, identificando las relaciones existentes entre subtipo y súper-tipo. Esto trae como consecuencia una información vaga del modelado.
- **Ontologías Formales.-** tienen sus categorías restringidas por axiomas y definiciones expresadas en lógica formal o en algún otro tipo de lenguaje que puede ser procesado por la computadora.



- **Meta-Ontologías.-** ontologías cuyos conceptos son utilizados para representar más ontologías.
- **Ontologías de contenido.-** parecidas a las ontologías genéricas porque permiten la reutilización de su conocimiento.
- **Ontologías de indexación.-** estas ontologías, en una representación del conocimiento, permiten la recuperación del conocimiento compartido mediante agentes.
- **Ontologías de comunicación.-** permiten dar respuestas a preguntas específicas, para ello utilizan agentes en su estructura.
- **Ontologías de dominio.-** estas ontologías permiten la representación de un dominio en particular. Su principal objetivo es el reúso de ontologías para diferentes aplicaciones.
- **Ontologías generales o de sentido común.-** describen conceptos generales o básicos, por ejemplo: cosas, eventos, tiempo,...
- **Ontologías genéricas.-** los conceptos que se definen en estas ontologías se consideran genéricos, es decir, reutilizables en diferentes dominios.
- **Ontologías de tareas.-** describen un vocabulario común para representar una actividad o una acción.
- **Ontologías de aplicaciones.-** permiten representar las necesidades de los usuarios, con respecto a una aplicación específica.



- **Ontologías de Representación.-** estas ontologías proporcionan el vocabulario necesario para representar ontologías utilizando un determinado paradigma de representación del conocimiento.

## 2.6 Ventajas / Desventajas

### Ventajas

- Suministran una forma potente y eficiente de representar y compartir el conocimiento en una determinada área, para ello utiliza un vocablo común.
- Admiten el uso de un formato de intercambio de la representación del conocimiento, ya que cada vez se va potenciado más por la utilización de estándares y lenguajes de definición de ontología.
- Es una fuente de conocimiento, ya que implica la reusabilidad del mismo.
- La inclusión de ontologías en sistemas en funcionamiento permiten organizar su información.

### Desventajas

- Están sujetas a un dominio determinado.
- Dependen de la interpretación que el desarrollador haga del dominio.
- Podría surgir ambigüedad en determinados términos, al momento de integrar muchas ontologías.



- Al no tener claro el dominio de la aplicación, la creación y utilización de la ontología puede resultar costosa.

## 2.7 Metodologías

Para el diseño de una ontología en particular es necesario contar con una o varias metodologías, de acuerdo al dominio que se desea representar, ya que estas brindan soporte durante todo este proceso. Entre ellas podemos destacar algunas como:

### a) Metodología de Uschold y Gruninger TOVE <sup>53</sup>

Según Uschold y Gruninger (1996), ésta considera la construcción de un modelo formal de conocimiento, identificando los escenarios en los cuales se aplicará la ontología, mediante la formulación de preguntas en lenguaje natural llamadas cuestiones de competencia. Una ventaja es que aprovecha la riqueza del lenguaje natural para transformar lo informal a lo formal.

Las preguntas de competencia son cuestiones a las que la ontología debe responder.

Hay que seguir las siguientes fases:

- Captura de escenarios motivadores o prometedores.
- Formulación de preguntas informales en base a escenarios.
- Especificación de la terminología de la ontología en un lenguaje formal, como puede ser lógica de primer orden.
- Formulación de preguntas formales de competencia, utilizando la terminología formal.

---

<sup>53</sup> Toronto Virtual Enterprise



- Especificación de los axiomas y definiciones para los términos de la ontología en el lenguaje formal.
- Establecer las condiciones para caracterizar la integridad de ontología.

#### **b) Metodología CyC**

La metodología CyC según Valencia R., (2005 a) “consiste en extraer manualmente la información del conocimiento común que está implícito en diferentes fuentes. Una vez extraída la información y cuando sea suficiente para la ontología, se la adiciona usando herramientas de procesamiento de lenguaje natural o aprendizaje computacional”.

Los pasos a seguir son los siguientes:

- Extracción y codificación manual del conocimiento del dominio de la ontología.
- Representación del lenguaje natural en una herramienta de software.
- Realizar la codificación de la ontología en herramientas y lenguajes para permitir su representación.

#### **c) Metodología Uschold y King**

Esta metodología fue presentada por Uschold & King en 1995. Fue desarrollada en base a la experiencia de desarrollo de ontologías. Los autores hacen hincapié en la importancia de la documentación y evaluación de la ontología y dejan abierta la posibilidad de utilizar otras ontologías.

Los pasos a seguir son los siguientes:





- Identificar propósito.
- Construcción de la Ontología
  - Capturar la ontología.
  - Codificación.
  - Integrar ontologías existentes.
- Evaluación.
- Documentación.

#### **d) Metodología Sensus**

Según Valencia R. (2005 c), es un método de construcción de ontologías, que parte de una rama de una ontología más general y especializada para obtener una ontología nueva. Es decir consiste en crear ontologías específicas de dominio a partir de términos denominados semillas<sup>54</sup> que representan un dominio en particular, permitiendo al usuario encontrar los términos más relevantes. A partir de este término se generan más conceptos, obteniendo al final un esqueleto del dominio del conocimiento a representar, es decir una ontología más general.

Posee los siguientes pasos:

- Tomar una serie de términos como semillas.
- Enlazarlos manualmente. Incluir todos los conceptos en el camino que va de la raíz a los conceptos semilla.
- Añadir nuevos términos relevantes del dominio.

#### **e) Metodología On-To-Knowledge**

Esta metodología se utiliza para la "organización de la información disponible electrónicamente, para mejorar la calidad de la gestión del

---

<sup>54</sup> Semillas.- elementos bases para genera clases



conocimiento en organizaciones grandes y distribuidas. Esto permite un soporte automático para la adquisición, mantenimiento y la obtención de grande volúmenes de información” Schnurr y otros, (2000 a).

Este proceso de desarrollo ontológico nos permite tener una visión clara ya que nos proporciona una guía para el desarrollo y mantenimiento de ontologías. Se basa en el análisis y diseño de casos de uso y se considera la identificación de los objetivos que deberían alcanzar las aplicaciones.

### **Fases de la metodología OTK**

OTK es una metodología utilizada cuando la ontología que se pretende desarrollar constituye el insumo principal de un software. Este método permitirá el desarrollo de la ontología de manera conjunta con la aplicación, E.Ramos (comunicación personal, 11 de abril, 2008). Consta de las siguientes fases: estudio de factibilidad, comienzo, refinamiento, evaluación y mantenimiento. Cabe indicar que la fase de refinamiento es donde ocurre el perfeccionamiento de la taxonomía, ya que a partir de esta se desarrollará la ontología. Lo interesante de OTK, es que involucra una etapa bien importante del desarrollo ontológico como es la evaluación de la ontología en todas y cada una de las etapas del desarrollo. La Fig. 2.2 muestra más en detalle las actividades que se realizan en cada fase

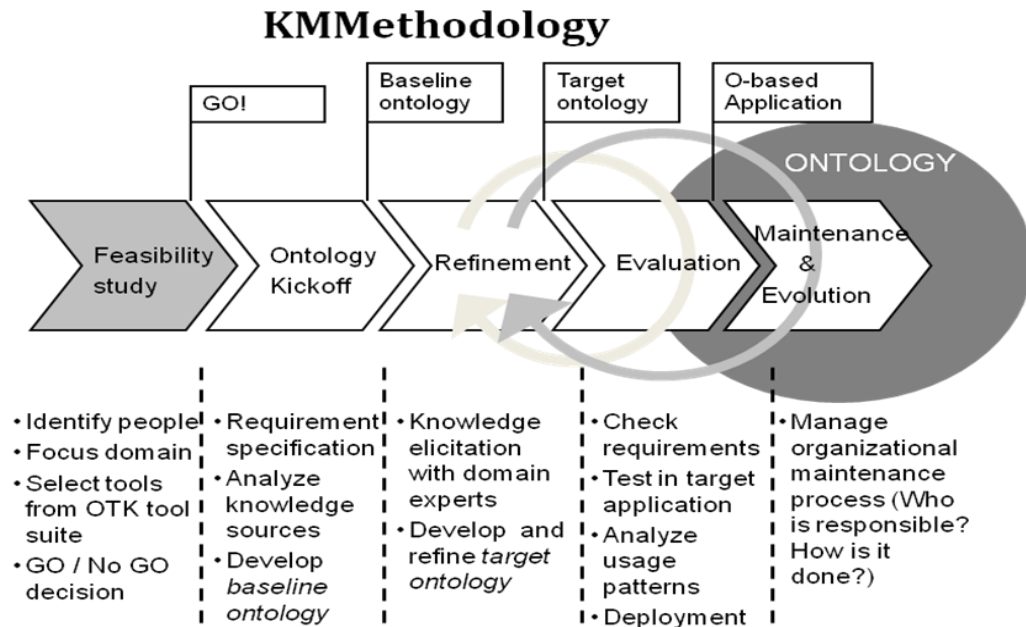


Figura.2.1 Ciclo de Vida Metodología OTK<sup>55</sup>

Según lo expuesto por Schnurr y otros (2000b) esta metodología consta de las siguientes fases:

#### **Fase 1: Estudio de factibilidad**

Aquí se realiza un estudio detallado de gerencia, es decir, las posibilidades concretas de éxito o fracaso, ya que se identifican las oportunidades potenciales y los posibles problemas y soluciones.

- **Identificar problemas y oportunidades**

Se realiza un estudio minucioso de los problemas tanto económicos, de viabilidad, como de tecnología, además se analiza

<sup>55</sup> Tomado de Otk.presentation disponible en [www.ontoknowledge.org/download/otk.presentation.ppt](http://www.ontoknowledge.org/download/otk.presentation.ppt)



las potenciales oportunidades de solución a problemas determinados.

- **Identificar a las personas involucradas**

Otorgar un rol y una tarea a cada persona dentro del grupo de desarrollo.

- **Seleccionar herramientas de desarrollo**

Se llegará a un acuerdo de las herramientas a utilizar durante el desarrollo y explotación de la ontología, se las seleccionará de acuerdo a parámetros como licencia, costos, portabilidad,...

## **Fase 2: Comienzo**

Una vez que se ha cumplido con la fase anterior se comienza el desarrollo de la ontología. En esta fase se deberán cumplir con varios pasos fundamentales, los que se describen a continuación:

- **Objetivo de la ontología**

Esto nos indica donde queremos llegar con la ontología, es decir cuál es el propósito de su desarrollo.

- **Dominio y Alcance**

Se define el dominio que cubrirá la ontología, especificando el número de conceptos, el nivel de granularidad,... Esto se logra mediante el análisis realizado en la fase de estudios de factibilidad del proyecto.

- **Aplicaciones de apoyo o soporte de la ontología**

El desarrollador en base al análisis realizado en el estudio de factibilidad determina el propósito de la ontología, además se le ofrece al usuario la interfaz que tendrá la ontología.



- **Fuentes de Conocimiento**

El diseñador de la ontología analiza las diferentes fuentes de conocimiento que pueden ser personas, documentos, páginas Web,...

- **Casos de uso y escenarios**

Se describe los potenciales casos de uso y escenarios en los cuales la ontología, como su contextualización, tendrá que optar o dar solución a un problema, vistos desde el punto de vista del usuario.

- **Preguntas de competencia**

En sí, es el uso real de la ontología a las posibles preguntas a las cuales debe responder, las que son formuladas por el usuario.

### **Fase 3: Refinamiento**

En esta fase se desarrolla la primera versión taxonómica de la ontología, basada en el conocimiento adquirido en la fase anterior. Esta tiene las siguientes etapas.

- **Recolección de semillas para la taxonomía**

La taxonomía en una ontología es la columna vertebral, por ello se empieza con la recolección de semillas, las que dan origen a la taxonomía mediante la generación de conceptos y su respectiva jerarquización, luego se procede a identificar relaciones,...

- **Conceptualización y Formalización**

Se debe tener clara la representación de la fuente de conocimiento como son, conceptos, atributos, relaciones,... para así poder usar un lenguaje ontológico.



- **Herramienta de apoyo para formalizar la ontología**

Herramientas en las cuales el lenguaje semi-formal representado en una hoja de diseño, se lo puede formalizar mediante el uso de herramientas de representación del conocimiento.

**Fase 4: Evaluación**

Examinar si la ontología satisface los requerimientos especificados en el documento de requerimientos.

- Verificar la capacidad de respuesta a las preguntas de competencias.
- Chequear el funcionamiento de la aplicación desde el punto de vista de navegabilidad, búsqueda de conceptos y relaciones, entre otros aspectos.

**Fase 5: Mantenimiento**

- Reunir los cambios identificados
- Actualizar la ontología según las reglas especificadas

**f) Metodología Methontology**

Esta metodología fue desarrollada en la Universidad Politécnica de Madrid, permite la construcción de ontologías desde cero o mediante la reutilización de ontologías existentes. Esto permite un elevado nivel de detalle, seguido de un ciclo de vida basado en la evolución de prototipos y técnicas particulares para realizar cada actividad.

Por otro lado, Methontology está dirigida al desarrollo del modelo conceptual del dominio, es extremadamente detallada y al aplicarla de manera correcta garantiza una ontología reutilizable y bastante completa.



### **El ciclo de vida de Methontology**

Las actividades que se realizan en este ciclo de vida son:

- Actividades de administración: estas actividades se dan en forma continua durante todo el proceso de desarrollo y construcción de la ontología, y abarca la identificación de tareas a realizar, el tiempo y los recursos necesarios a utilizar, para ello se realizan dos actividades: control y aseguramiento de la calidad. Mediante el control se garantiza terminar las fases en los tiempos establecidos, y el aseguramiento de la calidad garantiza un producto aceptable y con una buena documentación.
- Actividades de desarrollo: en esta fase se realizan actividades de pre-desarrollo, desarrollo y post-desarrollo de la ontología.

En la fase de pre-desarrollo se estudia la viabilidad del proyecto así como en donde se utilizará la ontología.

En la fase de desarrollo se realizan los siguientes pasos.

- Especificación.
- Conceptualización.
- Formalización.
- Implementación.

En la fase de post-desarrollo se realizan actividades concernientes con el mantenimiento y actualización de la ontología.

- Actividades de Apoyo: se ejecutan de manera paralela durante todas las actividades de desarrollo, tienen los siguientes pasos:
  - Adquisición de conocimiento.



- Integración,
- Evaluación, documentación
- Manejo de configuración.

Estas actividades deben ser evaluadas rigurosamente para evitar la propagación de errores, especialmente en la conceptualización, que es la más importante en este proceso ya que un error conlleva a un fracaso. Las actividades de adquisición de conocimiento, integración y evaluación son consideradas etapas importantes durante el ciclo de vida de esta metodología. La Methontology *“propone un ciclo de vida de construcción de la ontología basado en prototipos evolutivos, porque esto permite agregar, cambiar y remover términos en cada nueva versión.”* Ramos y Gil (n.f.).

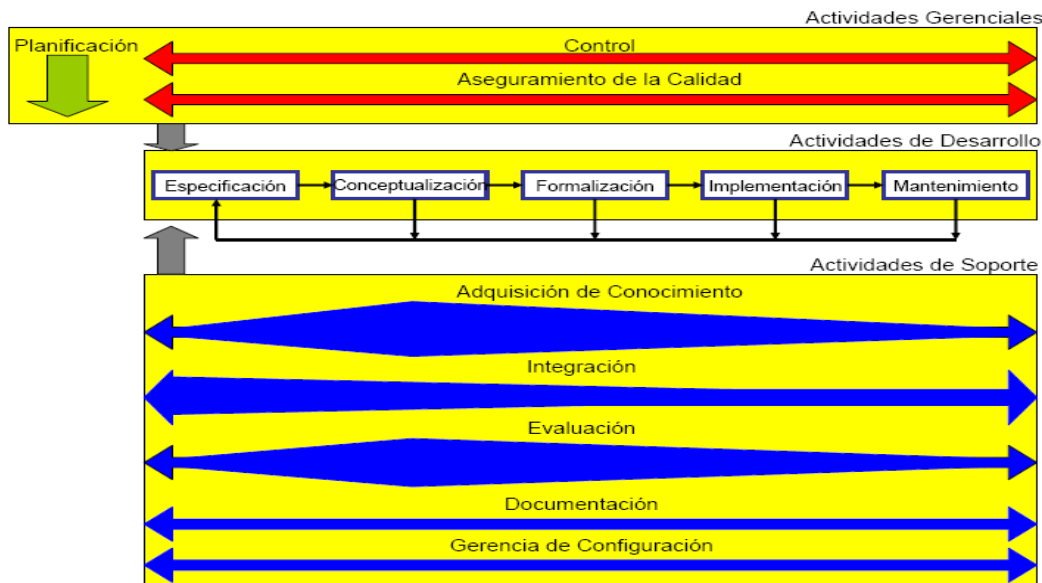


Figura 2.2 Ciclo de vida Methontology<sup>56</sup>

<sup>56</sup> Tomado de Ontology Building Methodologies and Tools - Schiappelli. Disponible en [http://www.dsi.uniroma1.it/~estrinfo/sws/2003\\_09\\_19/Ontology%20Building%20Methodologies%20and%20Tools%20-%20Schiappelli.ppt](http://www.dsi.uniroma1.it/~estrinfo/sws/2003_09_19/Ontology%20Building%20Methodologies%20and%20Tools%20-%20Schiappelli.ppt)





## **Fases de las actividades de desarrollo de la metodología Methontology**

Las actividades de desarrollo de la metodología de diseño ontológico Methontology según, Ramos Esmeralda (2007), consta de varias etapas, las cuales se detallan a continuación:

### **Fase 1: Especificación**

Esta fase consiste en realizar una especificación donde se señale varios puntos relacionados con la administración el proyecto.

- **Alcance**

Este paso determina el área que cubrirá la representación del conocimiento mediante la ontología.

- **Objetivos**

Especificar los objetivos para el usuario y la ontología, que se desean alcanzar durante su desarrollo.

- **Propósito**

El por qué la construcción de la ontología, a qué problema le dará solución.

- **Usuarios finales de la ontología**

Quienes harán uso de la representación del conocimiento.

### **Fase 2: Conceptualización**

Consiste en la transición de una representación informal de conocimiento a una representación semi-formal, mediante la representación de tablas, diagramas, relaciones y reglas, las mismas que servirán de interpretación para su implementación.



- **Construir el glosario de términos**

Es el glosario de términos relevantes del dominio de la aplicación, las mismas que son categorizadas como conceptos, relaciones, instancias,... Estas se describen en lenguaje natural.

- **Construir la taxonomía de conceptos**

La taxonomía es una representación jerárquica de los conceptos que posee el dominio del conocimiento a representar.

- **Construir un diagrama de relaciones binarias**

El objetivo es describir las relaciones que existen entre los conceptos de la taxonomía.

- **Construir el diccionario de conceptos**

Este diccionario contiene los conceptos del dominio de la aplicación, sus relaciones, instancias, atributos de clases y atributos de instancias a un lenguaje semi-formal.

- **Definir las relaciones binarias en detalle**

Se describen las relaciones binarias y para cada una de ellas se deben especificar el nombre de la relación, conceptos fuente y destino, cardinalidad y relación inversa.

- **Definir los atributos de clases en detalle**

Estos atributos de la clase representan la característica propia del concepto. Se lo puede representar en una tabla donde constan los siguientes campos: el nombre del atributo, concepto donde es definido, el valor que posee y su cardinalidad.



- **Definir los atributos de instancia en detalle**

Los atributos de instancias son aquellos que describen las instancias de un concepto, y los valores que toman pueden ser diferentes para cada una de ellas. Para cada atributo de instancia, se debe especificar: nombre, concepto al que pertenece, tipo de valor, rango de valores y cardinalidad.

- **Definir los axiomas formales**

En la ontología los axiomas formales sirven para definir sentencias que siempre son ciertas en el proceso de modelado del conocimiento, así como también para verificar la consistencia de la ontología. Se los representa en una tabla en la cual se especifica nombre del axioma, descripción del mismo, los conceptos que intervienen y las entidades.

- **Definir las reglas**

Son las que infieren conocimiento de la ontología ya que le permiten razonar y deducir posibles respuestas acorde al modelo que se implementó en el diseño y construcción de la ontología.

- **Definir las instancias**

Son los diferentes elementos que forman parte de una determinada clase del dominio. Las instancias son los diferentes objetos de una clase en particular. Por ejemplo de la clase *Animales*, sus instancias serían gato, perro, caballo, león,...

### **Fase 3: Formalización**

En esta fase se debe realizar la transformación del modelo conceptual, a un modelo formal o semi-computable, es decir tener claro cuáles son los conceptos importantes y definitivos que van en una ontología.



#### **Fase 4: Implementación**

Una vez que se ha obtenido el modelo definitivo de la ontología, se debe realizar la codificación o implementación de dicho modelo ontológico en un lenguaje formal (Ontolingua, XOL, OIL, DAML, OWL,...).

#### **Fase 5: Mantenimiento**

Esta actividad permite la actualización y corrección de la ontología, cuando se producen cambios en la fuente de información o requerimiento a nivel de usuarios.

Dentro de las actividades de soporte de la metodología, cabe destacar la *evaluación*, y *documentación*, ya que estas actividades se realizan a manera de soporte durante el desarrollo de una ontología. Con la evaluación se determina la correcta extracción de la información en la ontología, para ello se aplicará el método OntoQA, orientado a evaluar la población de clases y la estructura de la ontología. La documentación detalla los pasos y actividades realizadas en cada etapa del desarrollo de la ontología.

### **2.8 Análisis de las metodologías**

A continuación, en la Tabla 2.1 se presenta un análisis comparativo de las metodologías, el mismo que se enfoca en las siguientes características:

- Ciclo de vida propuesto: es importante ya que éste comprende todos los procesos que se realizan desde la concepción, construcción y mantenimiento de la ontología.



- Las Herramientas de soporte: herramientas ontológicas, que permiten tener claro el nivel de expresividad que se puede tener al pasar de lo informal a lo formal.
- Modelado: este permite tener una idea sobre cómo se realiza la extracción de conceptos.
- Nivel de abstracción: permite la correcta extracción de conceptos, lo que permite una elevada granularidad de la ontología, evitando la formación de islas<sup>57</sup>.
- El uso de ontología base es importante ya que esto permite analizar otras ontologías existentes y anexarlas a la ontología que está en proceso de construcción.

<b>Características</b> <b>Metodologías</b>	<b>Ciclo de vida Propuesto</b>	<b>Herramientas de soporte</b>	<b>Modelado</b>	<b>Nivel de Abstracción</b>	<b>Uso de Ontología de Base</b>
<b>Methontology</b>	Secuencial con prototipos de desarrollo y actividades de administración y apoyo paralelo al desarrollo	ODE WebODE OntoEdit Protégé 2000	Identificación directa de conceptos y organización taxonómica	Elevado	No
<b>Grüninger y Fox</b>	Secuencial	No especificada	Preguntas de competencia	Alto	No
<b>CyC</b>	Secuencial	No especificada	Extracción manual	Alto	No
<b>Uschold y King</b>	No especificada	No especificada	Identificación directa de conceptos	Normal	No
<b>OTK</b>	Incremental y cíclica con prototipos de	OntoEdit Corporum	Casos de uso Semilla	Bajo	No

<sup>57</sup> Isla.- acumulación excesiva de información en ciertos lugares del dominio en representación



	desarrollo				
<b>Sensus</b>	No propuesto	No especificada	Semilla	Normal	Si

Tabla 2.1 Análisis de las metodologías de desarrollo de ontologías

Además de las características presentadas en el cuadro comparativo, y considerando la descripción anterior de cada una de las metodologías, se observa que todas las metodologías de una u otra forma abarcan cuatro pasos fundamentales:

1. **Identificar el propósito y el alcance.-** en este punto se debe determinar el dominio de aplicación, los objetivos de interés del dominio, y las tareas que va a realizar la ontología, es decir se debe contestar a la interrogante ¿para qué va a construirse la ontología?
2. **Construcción de la ontología.-** es la representación clara del escenario a modelar a través de un lenguaje formal, se debe analizar la reutilización de ontologías existentes y definirse las posibles ontologías a utilizar. Este paso a su vez puede dividirse en dos fases o etapas: captura y codificación.

**Captura.-** es identificar los conceptos y relaciones claves en el dominio de interés en lenguaje natural. Incluye los siguientes pasos:

- Proponer definiciones no ambiguas de conceptos y de sus relaciones.
- Identificación de términos claves para referirse a estos conceptos y relaciones.

**Codificación.-** es representar el conocimiento adquirido en el paso anterior en un lenguaje formal, se siguen los siguientes pasos:



- Seleccionar el lenguaje de representación adecuado.
  - Codificación de la ontología en el lenguaje seleccionado.
  - Integración de ontologías existentes: cuáles y de qué forma se va a reutilizar el conocimiento de dichas ontologías.
3. **Evaluación.**- en esta fase ya se tiene un diseño definitivo de la ontología, existen muchos métodos de evaluación como OntoQA y así determinar el nivel de extracción de información. También se evalúa la consistencia taxonómica, además se puede considerar si la ontología puede ser reutilizable en el futuro para posibles aplicaciones.
4. **Documentación y reutilización.**- la documentación se hace de forma paralela a los puntos anteriores. Debe considerar:
- La justificación de las decisiones tomadas.
  - La evaluación.
  - Instrucciones adicionales para su uso.

Una especificación más clara de la interacción entre las fases de desarrollo de una ontología, se muestra en la siguiente figura:

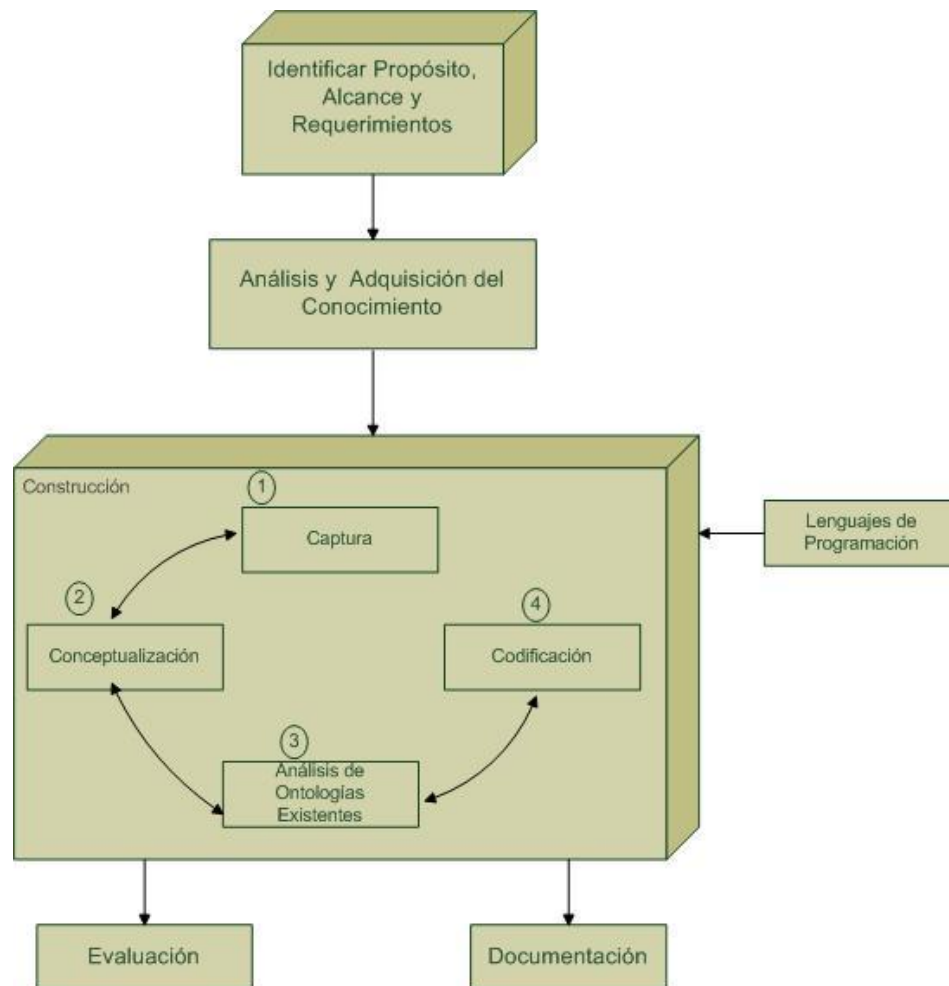


Figura 2.3 Fases durante el ciclo de vida de una ontología

Luego del análisis desarrollado para el diseño y construcción de la ontología de la biblioteca virtual de la UTPL, se propone seguir la metodología Methontology, ya que nos permite un desarrollo basado en prototipos que pueden cambiar de acuerdo a la evolución y sobre todo un nivel elevado de detalle en su conceptualización, aspectos importantes que garantizan la calidad de la ontología.





## 2.9 Habituales errores en taxonomías y posibles anomalías del diseño

En el diseño y construcción de ontologías es importante la captura de errores. Gómez-Pérez (1999 2001), citado en Muhammad Fahad (n.f) identifica tres clases principales de errores en las taxonomías como son: errores de inconsistencia, errores de completitud y errores de redundancia.

### 2.9.1 Errores de Inconsistencia

Existen tres errores que causan la inconsistencia y ambigüedad en una taxonomía y por ende en una ontología: Circulatory errors, Partition errors y Semantic inconsistency errors.

- **Circulatory errors.-** este error se da cuando se define una clase o superclase de sí mismo, en cualquier nivel jerárquico de la ontología. Por ejemplo cuando se define como subclase páginas web de la clase fuentes de información y fuentes de información está definida como subclase de páginas web.
- **Partition errors.-** este tipo de error se da principalmente cuando existe una descomposición de superclase a subclases, es decir cuando todas las características de las subclases son independientes y no se solapan entre sí, esto conduce a la descomposición disjunta. Para evitar todo esto se debe tener claro las clases, sus atributos e instancias, ya que una instancia no puede pertenecer a dos clases, o una instancia de una clase no puede pertenecer a la subclase. Por ejemplo cuando la clase contenido tiene como instancia el nombre de la instancia perteneciente a la clase materia.



- **Semantic inconsistency errors.-** este error se produce cuando se hace una clasificación taxonómica jerárquica de clase de una ontología de manera incorrecta, es decir cuando se tiene un concepto y éste tiene una subclase y ésta subclase en realidad no tiene nada que ver con la clase. Por ejemplo si la clase servicio de base de datos tuviese como subclase la clase revista, teniendo en cuenta que estas son totalmente diferentes aunque tengan algunos atributos en común.

### 2.9.2 Errores de completitud

Este tipo de errores se da cuando al realizar un análisis se pasa por alto ciertos aspectos importantes de los conceptos. Esto puede causar una cierta ambigüedad afectando a los mecanismos de inferencia, las causas pueden ser:

- **Incompleta clasificación.-** cuando se pasan por alto algunos conceptos que se considera que no son relevantes para la representación del dominio. Por ejemplo en la ontología desarrollada se pasa por alto el concepto área, esto causa que no se pueda diferenciar la pertenencia de determinada carrera, por consiguiente no se le pueden asignar bases de datos con información específica.
- **Errores de Partición.-** cuando se omiten axiomas sobre la clasificación de conceptos, esto influye en la disminución del razonamiento y por ende limita el poder de inferir conocimiento. En la ontología si se omitiese el axioma



todas las carreras tienen materias esto causará una inconsistencia en la ontología.

- **Omisión del Conocimiento Disjunto.**- este se produce cuando existe un concepto con varias subclases y a éstas se omite el incluirlas como clases disjuntas. Por ello todas las clases hermanas tienen que ser consideradas como disjuntas.
- **Omisión del Conocimiento Exhaustivo.**- la causa de este error es cuando las ontologías no tienen un límite, lo que causa una exhaustiva creación de clases y subclases.

### 2.9.3 Errores de redundancia

Son causados cuando la información se infiere más de una vez en las relaciones existentes en la ontología.

**Las redundancias de Subclase\_de, Subpropiedad-de las relaciones e Instancia\_de.**- se producen cuando existe una relación directa e indirecta entre las clases y sus respectivas subclases, es decir existe una doble inferencia.

### 2.10 Diseño y Construcción

Cuando se piensa desarrollar y construir una ontología, es necesario tener en cuenta el ámbito en que se la va a aplicar, para lo cual se debe seleccionar una buena herramienta de diseño y construcción, además del análisis, la(s) metodología(s) y los criterios.



### 2.10.1 Criterios

Para la definición de criterios, existen diferentes aportes de algunos autores, los cuales coinciden en algunos de ellos. V. R. Benjamins (n.f), establece algunas características que debe tener una ontología (claridad, coherencia, completitud, extensibilidad, mínimo compromiso ontológico y diversificación de ontologías).

Según Gruber citado en Gómez Pérez (n.f.c), los principios para diseñar una ontología son: claridad y objetividad, completitud, coherencia, máxima extensibilidad monótona, principio de distinción ontológica, diversificación, estandarización y minimización.

La mayoría de estudios realizados citan de una u otra manera a los principios de diseño de Gruber (1993), y algunos adicionan otros que van relacionados. Así podemos detallar los más relevantes:

**Claridad.-** la ontología debe tratar de comunicar lo que realmente significan sus términos. Al momento de plantear las definiciones, éstas deben ser claras y objetivas, y ser escritas en un lenguaje natural, es decir algo entendible.

**Coherencia.-** la ontología debe hacer deducciones que sean consistentes con sus definiciones.

**Extensibilidad.-** se debe tener en cuenta, nuevos usos que se puedan dar a la ontología, que permitan en un futuro extensiones y especializaciones.

**Precisión.-** se debe tener la menor cantidad de supuestos del entorno que estamos modelando, sin que con ellos se quite la representación del conocimiento. En otras palabras, las consultas



que realice la ontología, deberán tener la capacidad de recuperar información pertinente a la consulta realizada.

### 2.10.2 Lenguajes para construir ontologías

**RDF.-** según Lamarca María (2008), RDF es un lenguaje para especificar metadatos. Una descripción RDF es un conjunto de sentencias o declaraciones, y a una proposición se conoce como una tripleta, porque está compuesta de 3 elementos: sujeto, predicado y objeto. Es posible representar esta tripleta en forma de grafos dirigidos, donde el sujeto y objeto son nodos, mientras que el predicado es un arco.

Además una tripleta se representa mediante nodos conectados por líneas con etiquetas, donde los nodos representan recursos y las líneas con etiquetas representan las propiedades de esos recursos.

**KIF.-** para Iglesias Moro (2000), KIF es un lenguaje para el intercambio de conocimiento (lenguaje natural o lenguaje de programación) entre programas diferentes. Diseñado para intercambiar conocimiento entre sistemas de computación distinta, de diferentes lenguas.

Características:

- Lenguaje formal basado en lógica de primer orden para representar conocimiento.
- Permite definir objetos, funciones, y relaciones. Contiene cuantificador universal y existencial.
- No depende de intérpretes.



- Contiene lógicamente a lenguajes de Bases de Datos y a lenguajes como Prolog.

Sintaxis:

- Caracteres
- Letras mayúsculas
- Letras minúsculas
- Dígitos
- Caracteres alfabéticos
- Caracteres especiales

Lexemas:

- Especiales
- Palabras (words)
- Referencias a caracteres
- Strings de caracteres
- Blocks de caracteres

Expresiones:

- Términos
- Sentencias
- Reglas
- Definiciones

**SHOE.-** fue el primer lenguaje de etiquetado para diseñar ontologías en la web. Según García R. (2000), SHOE permite definir clases y reglas de inferencia, pero no negaciones o disyunciones.

Soporta adquisición del conocimiento aumentando la web con etiquetas que suministran semántica.



La estructura básica son las ontologías, que definen las reglas que dicen qué tipo de afirmaciones pueden hacerse y qué clase de conclusiones pueden derivarse de esas afirmaciones, y las instancias, que realizan las afirmaciones basándose en esas reglas.

**DAML + OIL.-** en el manual Lenguaje de codificación de ontologías DAML + OIL, este lenguaje procede de la unificación de dos lenguajes: DAML y OIL. Una ontología DAML+OIL, tiene varios componentes, algunos de los cuales son opcionales y otros se pueden repetir. Además tiene una cabecera, elementos clase, elementos propiedad e instancias. Está basado en RDF, utilizando su modelo y sintaxis. Este lenguaje extiende el RDF Schema (rdfs) permitiendo describir tipos de objetos y relaciones entre ellos de manera más completa.

#### Características

- La cardinalidad de los elementos y propiedades pueden ser especificadas globalmente y localmente.
- Las propiedades y rangos pueden especificarse global o localmente.
- Los tipos de datos básicos pueden ser especificados desde el XML Schema.

**OWL<sup>58</sup>.**- lenguaje de etiquetado semántico para publicar y compartir ontologías en la Web. Permite definir el significado de términos en vocabularios y las relaciones entre ontologías. Es una extensión del lenguaje RDF. Está diseñado para usarse cuando la información contenida en los documentos necesita ser procesada por programas

---

<sup>58</sup> OWL Web Ontology Language



o aplicaciones. Se estructura en capas y puede ser adaptado a las necesidades de los usuarios.

### **2.10.3 Herramientas de diseño y construcción**

Son herramientas que sirven para desarrollar, reutilizar y mantener ontologías de diferente índole. La siguiente tabla 2.2 muestra un análisis comparativo.





NOMBRE	DESCRIPCIÓN	ALMACENAMIENTO EN BASES DE DATOS	CARACTERÍSTICAS DEL MODELADO	PLATAFORMAS	FORMATO IMPORTACIÓN EXPORTACIÓN
Protégé	Posee interfaz amigable. Puede ser extendida con elementos gráficos como tablas, diagramas, para acceder a otras ontologías. El acceso a la base del conocimiento puede ser a través de un applet. La comprobación automática de consistencia de la ontología se la hace mediante plugins.	Oracle MySQL Microsoft SQL Server Microsoft Access	Herencia múltiple y jerarquías de relación soporte de especificación de instancias, axiomas con restricciones	Multiplataforma, ya que está desarrollada o en Java, soporte JDK 1.4.0 o superior	RDF(S) XML, Schema RDB
OILEd	Editor gráfico de ontologías que permite construir ontologías utilizando DAML+OIL. Basado en Protégé. Las clases están definidas en función de sus súper clases y las restricciones de sus propiedades. Permite editar ontologías, también puede trabajar con grandes bases de conocimiento. No soporta multiusuario.	No soporta almacenamiento de datos.	Se da mediante axiomas estrictivos de DAML. Permite una creación de metadatos	Multiplataforma, ya que esta desarrollada o en Java, soporte JDK 1.4.0 o superior	RDFS SHIQ
Onto Edit	Soporta el desarrollo y mantenimiento de ontologías de forma gráfica. Permite representación y modelado de conceptos, relaciones y axiomas. En base a plugins se permite extender su funcionalidad.	Oracle, MySQL, Microsoft SQL Server DB2, PostgreSQL	Permite herencia múltiple y jerarquías de relaciones	Multiplataforma, ya que esta desarrollada o en Java, soporte JDK 1.4.0 o superior	DAML+OIL RDF Schema OXML Frame- Logic



<p>Ontolin gua Server</p>	<p>Permite crear, editar, modificar y usar ontologías. Sistema cliente-servidor. Posee un motor de validaciones el cual trabaja en tiempo real. Acceso a base del conocimiento se da vía web a través de un navegador de Ontolingua.</p>	<p>Se almacenan en una base de datos relacional Access, Oracle</p>	<p>El modelo permite incluir relaciones de ontologías, como pueden ser clases subclases y relaciones.</p>	<p>Multiplataforma, ya que esta desarrollada o en Java, soporte JDK 1.4.0 o superior</p>	<p>Ontolingua OKBC LOOM EPIKIT GENERICFR AME</p>
<p>Jena</p>	<p>Ofrece un entorno para RDF, esquemas RDF y OWL. Desarrollado por Hewlett Packard, permite leer, recorrer y modificar grafos tanto RDF como OWL desde un programa Java. Permite guardar las ontologías en RDF textual, en formato de base de datos.</p>	<p>Oracle, Microsoft Access, My Sql</p>			<p>RDF RDFS OWL</p>

Tabla 2.2 Análisis comparativo de las diferentes herramientas de diseño de ontologías



#### **2.10.4 Análisis del lenguaje y las herramientas de diseño y construcción**

Después de haber realizado un análisis de cada una de los lenguajes y las herramientas de diseño, se ha considerado utilizar como mejor alternativa para el diseño y construcción de la ontología para la Biblioteca Virtual de la UTPL, la herramienta PROTEGE conjuntamente con el lenguaje de construcción RDF, debido a que este lenguaje puede ser utilizado con el razonador RDF API de Jena.

De las herramientas mencionadas, la más difundida es Protégé ya que entre sus principales ventajas es que facilita diseñar y construir ontologías en varios formatos (Rdf, Rdfs, Owl,...)

##### **Principales características**

- Es de código abierto.
- Permite la representación del conocimiento de forma clara y sencilla.
- Tiene compatibilidad con la mayoría de las bases de datos.
- Mediante grafos se puede observar el árbol ontológico que se está diseñando.
- Define clases OWL mediante lógica expresada en OWL.
- Conexión con razonadores externos para validar el modelo mediante clasificadores de lógica descriptiva, como puede ser Racer Pro.
- Edición y creación de instancias de las clases creadas en la ontología.
- Instalación sencilla e incluye todas las librerías necesarias para su correcto funcionamiento.
- Tiene una interfaz amigable y sencilla de manejar.
- Trabaja con ontologías OWL y RDF, permitiendo exportación a otros formatos.



## 2.11 Evaluación de la calidad de la ontología

Un aspecto importante es la medición de la calidad de la ontología. Existen varias herramientas que nos permiten realizar este proceso. La mayoría de estas lo hacen de una manera genérica, es decir independientemente de la herramienta de representación, el lenguaje de desarrollo y la metodología aplicada durante este proceso.

La mayoría de autores concuerdan en cinco puntos a evaluar según lo expuesto por Strasunskas y Tomassen (n.f.), como son: la sintaxis, el vocabulario, la estructura, la población de clases, estadísticas de uso y metadatos.

**La sintaxis.-** este método de evaluación verifica si la ontología es sintácticamente correcta. Esta evaluación es importante en ontologías de aplicación ya que con una buena sintaxis, se puede tener un marco eficiente para poder procesar la información.

**El vocabulario.-** este análisis permite establecer la coherencia entre el dominio a representar y la ontología e incluye la taxonomía, las relaciones y las propiedades, es decir, analiza qué tan bien está representado el modelo lógico.

**La estructura.-** consiste en evaluar la estructura taxonómica frente a la estructura de relaciones semánticas. Mediante esta evaluación se puede determinar la diversidad y riqueza de la relación.

**La población de clases.-** se realiza un análisis para saber qué tan bien ha sido poblada la base de conocimientos. Esto se hace mediante un análisis de comparación realizado entre el número total de instancias con el número total de las clases.



**Estadísticas de uso y metadatos.**- esta evaluación se basa en el nivel de anotación de la ontología, es decir los metadatos y sus elementos. Para ello se pueden tomar puntos básicos como por ejemplo: el nivel de satisfacción del usuario, la procedencia de la información, el porcentaje de uso de la ontología,...

Cabe destacar que cada herramienta de evaluación tiene su propio método.

## 2.12 Método de evaluación OntoQA

En la actualidad a medida que las ontologías ganan terreno en el mundo web, se hace necesario métodos de evaluación. Existen varios métodos que se especializan en la evaluación de ontologías, y así resolver problemas que se presentan al momento de desarrollar dichas ontologías, uno de ellos es OntoQA.

OntoQA es un método de evaluación que implementa una serie de métricas, que serán desarrollados más adelante. Analiza la estructura de la ontología, su población y relaciones existentes. Ésta se puede aplicar antes o después de la formalización ontológica, al momento de evaluar determinados aspectos de la ontología y sus posibilidades de representación del conocimiento. Además permite tomar decisiones sobre el desarrollo, basados en la calidad de la ontología.

OntoQA está dividido en dos categorías como son:

1. Métricas de la estructura de la ontología (schemas) y
2. Métricas de instancias.
  - Métricas para la base de conocimientos,
  - Métricas de las clases.



Según Tartir y otros (n.f.) exponen las siguientes métricas:

**Métricas de la estructura de la ontología.**- aunque no se puede saber con exactitud si el diseño de la ontología está correcto, mediante estas métricas se pueden conocer cifras que indican la riqueza de la representación, tomando en cuenta la anchura, profundidad y la herencia, permitiendo evaluar el diseño de la ontología y su potencial forma de representación del conocimiento, y para lo cual tenemos las siguientes métricas.

- **Riqueza de relación (Relationship Richness).**- mediante este indicador se podrá saber el nivel de diversidad existente entre las diferentes relaciones en la taxonomía, partiendo de la premisa que no sean únicamente relaciones clase subclase.

Aplicando siguiente fórmula: si el resultado es cero (0), indica que existe un nivel muy pobre, es decir se darían relaciones clase subclase o denominadas relaciones Is-A, si por el contrario, el resultado es cercano a uno (1), las relaciones son mucho más ricas, ya que no existirían relaciones clase subclase sino una clase con otras clases, conocidas como relaciones non-Is-A.

$$RR = \frac{|P|}{|IsA| + |P|}$$

|P|: Number of non-IsA relationships

|IsA|: Number of IsA relationships



- **Riqueza de Atributos (Attribute Richness).**- por el número de atributos que pertenecen a una determinada clase, se puede determinar la riqueza del diseño, ya que los atributos nos dan información adicional de las instancias y mediante esto se transmite más conocimiento a la ontología. Aplicando la siguiente formula si tenemos un resultado elevado, esto permite saber que la ontología posee información suficiente para realizar una buena representación del conocimiento, caso contrario, si es bajo, se dice que aporta con poca información.

$$AR = \frac{|A|}{|C|}$$

|A|: Number of literal attributes

|C|: Number of classes

- **Riqueza de Herencia (Inheritance Richness (fan-out)).**- es fundamental determinar el nivel de agrupación en una ontología, ya que permite tener herencia, amplitud y los distintos niveles de profundidad en una clase padre. O simplemente notar si la ontología es horizontal o plana la ontología tiene un pequeño número de herencia, y cada clase tiene un número relativamente grande de subclases. En cambio, una ontología vertical contiene un gran número de herencia, los niveles de las clases tienen un pequeño número de subclases.

$$IR = \frac{\sum_{c_j \in C} |H^c(c_j, c_i)|}{|C|}$$

|H<sup>c</sup>(c<sub>j</sub>, c<sub>i</sub>)|: Number of subclasses of Class C<sub>i</sub>

|C|: Number of classes



### Métricas de instancias

Para evitar la pérdida de conocimiento, es muy importante la correcta ubicación de las instancias dentro de las clases, ya que permiten un uso eficaz de la ontología. Una correcta distribución de los datos permite mejorar la representación del conocimiento.

#### a) Métricas para la base de conocimientos

- **Riqueza de clase (Class Richness).**- es un indicador acerca de cómo están distribuidas las instancias en las clases. Este principio realiza una comparación entre el número de clases existentes en la ontología y las clases instanciadas, y si el número es muy bajo, se llega a concluir que no existe una correcta base de conocimientos, por el contrario, si el porcentaje es cercano al 100%, la base de conocimiento cuenta con información necesaria.

$$CR = \frac{|C'|}{|C|}$$

|C'|: Number of used classes

|C|: Number of defined classes

- **Promedio de instancias en todas las clases de la ontología (Average Population).**- es una guía de comparación entre el número de instancias y el número de clases. Es aplicable durante el proceso de desarrollo y cuando existe confusión entre el número de clases e instancias, por ejemplo, si el número de instancias es igual al de las clases.

La siguiente fórmula muestra cómo encontrar el promedio: si el resultado promedio de instancias por clase es bajo,





indica que el número de instancias por clase es insuficiente para una efectiva representación del conocimiento, pero si dicho resultado es elevado, quiere decir que es óptimo para representar el conocimiento. Pero en este caso puede existir una excepción, ya que dependiendo del dominio a representar, el promedio puede ser muy bajo o elevado.

$$P = \frac{|I|}{|C|}$$

|I|: Number of instances

|C|: Number of defined classes

#### b) Métricas de las clases

- **Importancia (Importance).**- mediante este indicador se puede tener una idea global sobre la distribución de instancias sobre una clase, permitiendo al desarrollador identificar las zonas más pobladas y las zonas donde se necesita más información. Esto define el total de instancias de un subárbol de una clase con respecto al número total de instancias de la base de conocimiento. Al aplicar la fórmula el resultado será una comparación de la cobertura real de instancias en la ontología con las que se tiene actualmente.

$$Imp = \frac{|C_i(I)|}{|I|}$$

|C<sub>i</sub>(I)|: Number of instances defined for class C<sub>i</sub>

|I|: Number of instances



- **Riqueza de la herencia (Inheritance Richness).**- mediante este indicador se puede ver en detalle la distribución de la información en el subárbol o subclases de una clase, lo que nos permite observar la correcta categorización tanto de clases como subclases en la base de conocimientos de la ontología. Este promedio define el número de subclases que posee una clase en relación con el número de nodos en las subclases. Si el resultado es bajo, significa que se está utilizando unas pocas clases con relación al esquema ontológico, en contraposición, si el resultado es elevado quiere decir la utilización de un número elevado de relaciones.

$$IR_{C_i} = \frac{\sum_{C_j \in C_i} |H^c(C_k, C_j)|}{|C_i|}$$

$C_i$ : Classes belonging to the subtree rooted at  $C_i$

$|H^c(C_k, C_j)|$ : Number of subclasses of Class  $C_i$

- **Riqueza de relación (Relationship Richness).**- este indicador muestra una visión clara de cómo se produce el proceso de extracción de información en la ontología. Esta pauta proporciona el nivel de uso de las propiedades de cada clase que actualmente están siendo utilizadas por las instancias, es decir, el número de relaciones utilizadas por las instancias pertenecientes a una determinada clase comparadas con el número de relaciones definidas para dicha clase. Al aplicar la fórmula el resultado será la representación del nivel de uso de la base de conocimiento.



$$RR_{C_i} = \frac{\left| \text{Distinct}(P(I_i, I_j)) : I_i \in C_i(I), I_j \in C_j(I), \forall C_j \in C \right|}{\left| P(C_i, C_j) \right|}$$

$P(I_i, I_j)$ : Relationships between instances  $I_i$  and  $I_j$ .

$C_i(I)$ : Instances of class  $C_i$ .

$C_j(I)$ : Instances of class  $C_j$ .

$C$ : Defined classes

$P(C_i, C_j)$ : Relationships between instances  $C_i$  and  $C_j$ .



# CAPITULO III

## **DISEÑO Y CONSTRUCCIÓN DE LA ONTOLOGÍA PARA LA BIBLIOTECA VIRTUAL DE LA UTPL**



### **3 DISEÑO Y CONSTRUCCIÓN DE LA ONTOLOGÍA PARA LA BIBLIOTECA VIRTUAL (BV) DE LA UTPL**

#### **3.1 Introducción**

En este capítulo se hablará de la implementación de la metodología escogida. Se describirán los pasos de cada una las fases de las actividades de desarrollo y cómo fueron implementadas.

También se hablará de las métricas que sirven o son de gran ayuda para evaluar la calidad del diseño del modelo ontológico.

Por último se hará una descripción detallada de los modelos lógicos de la ontología (clases, subclases, instancias, atributos).

#### **3.2 Metodología para el desarrollo**

Como se mencionó en el capítulo anterior, se ha escogido como metodología de desarrollo para el diseño y construcción de una ontología, la metodología conocida como Methontology.

Esta metodología comprende las fases de: especificación, conceptualización, formalización, implementación y mantenimiento, siendo la última fase desarrollada en trabajos futuros, ya que la ontología propuesta, solo abarca hasta su implementación.

#### **3.3 Aplicación de Methontology en el desarrollo**

##### **3.3.1 Especificación**

###### **Alcance**

El dominio es el área del conocimiento que envuelve la BV, en nuestro caso se refiere a las bases de datos, repositorios, recursos digitales y otras bibliotecas, que ofrece la UTPL en la página web de la biblioteca virtual. Se plantea utilizar la ontología para recomendar enlaces a los alumnos que ingresan al entorno EVA.



### **¿Para qué usaremos la ontología?**

- Para realizar recomendaciones personalizadas de enlaces, de acuerdo al perfil académico de cada estudiante y/o profesor.
- Para un mejor aprovechamiento de los recursos virtuales disponibles en la biblioteca virtual.
- Para minimizar los tiempos de búsqueda empleado por los estudiantes al momento de realizar consultas académicas.

### **Objetivos**

Diseñar y construir una ontología, para brindar recomendaciones de enlaces de biblioteca virtual de la UTPL a los usuarios del entorno EVA.

### **Usuarios finales de la ontología**

Los usuarios finales de la ontología serán los profesionales en formación, los docentes investigadores y público en general.

### **Identificar a las personas involucradas**

Las personas involucradas durante el desarrollo de la ontología son docentes investigadores, profesionales en formación, administrador de la BV y personal de la Unidad de Virtualización.

### **Especificar los requerimientos funcionales**

- Las recomendaciones se las harán en base al contenido de las asignaturas.
- Los enlaces presentados al estudiante se deben extraer desde los distintos recursos digitales de la biblioteca virtual, sean estas bases de datos, revistas, bibliotecas virtuales,...



- La herramienta seleccionada para la administración de la ontología, debe proveer funciones para la adición de nuevos elementos, y la edición y supresión de los existentes.
- El tiempo de respuesta para presentar las recomendaciones debe ser prudencial.

### **Especificar los requerimientos no funcionales**

- El software a utilizar será:
  - PHP.- lenguaje de programación empleado para la codificación de:  
Empleo del modelo ontológico en el razonador, las consultas hechas a la ontología, el bloque añadido al EVA y la interfaz de presentación al usuario.
  - MySQL.- conjuntamente con el lenguaje PHP, se utilizará para realizar las pruebas respectivas, para luego poder implementarlas en el EVA.
  - Protégé.- este software será utilizado para realizar el diseño y la construcción del modelo lógico de la ontología como tipo de archivo RDF.
- Se utilizará el lenguaje OWL para la implementación de la ontología.
- El tiempo de respuesta deberá ser prudencial, que no demore demasiado al momento de brindar respuestas (enlaces) al estudiante.



- Permitir múltiples usuarios simultáneos, quienes deben tener acceso al sistema, usarlo fácilmente y recibir respuesta con rapidez

### **Analizar las fuentes de conocimiento**

Se ha realizado un análisis al conjunto de información y datos disponibles electrónicamente, tomando en cuenta la estructura de la biblioteca virtual de la UTPL. En dicho análisis se ha identificado los términos importantes para este determinado contexto:

Como fuentes de conocimiento se tiene:

- Bases de datos
- Recursos Digitales
- Otras bibliotecas virtuales
- Registros de estudiantes
- Registros de materias UTPL
- Pensum académico

### **Casos de uso y escenarios**

Estos nos permiten tener una representación gráfica del comportamiento del sistema desde el punto de vista de los usuarios. Estos análisis nos ayudan a describir qué es lo que el sistema debe hacer.

### **Casos de Uso**

Mediante el caso de uso en la Fig. 3.1, se representa la secuencia de interacciones que se realizan entre el sistema y sus actores, con lo cual se comprende la comunicación que existe entre estos durante el desarrollo de la ontología, tomando en cuenta desde un principio la fase de inicio hasta la fase de construcción, en una herramienta de representación del conocimiento.





Figura 3.1. Diagrama de casos de uso



### Diagrama Conceptual

Este diagrama nos permite identificar los conceptos que conforman la ontología y así aumentar el nivel de comprensión del dominio que se ha de representar. En la figura 3.2 se puede tener una idea clara de las relaciones existentes en la ontología.

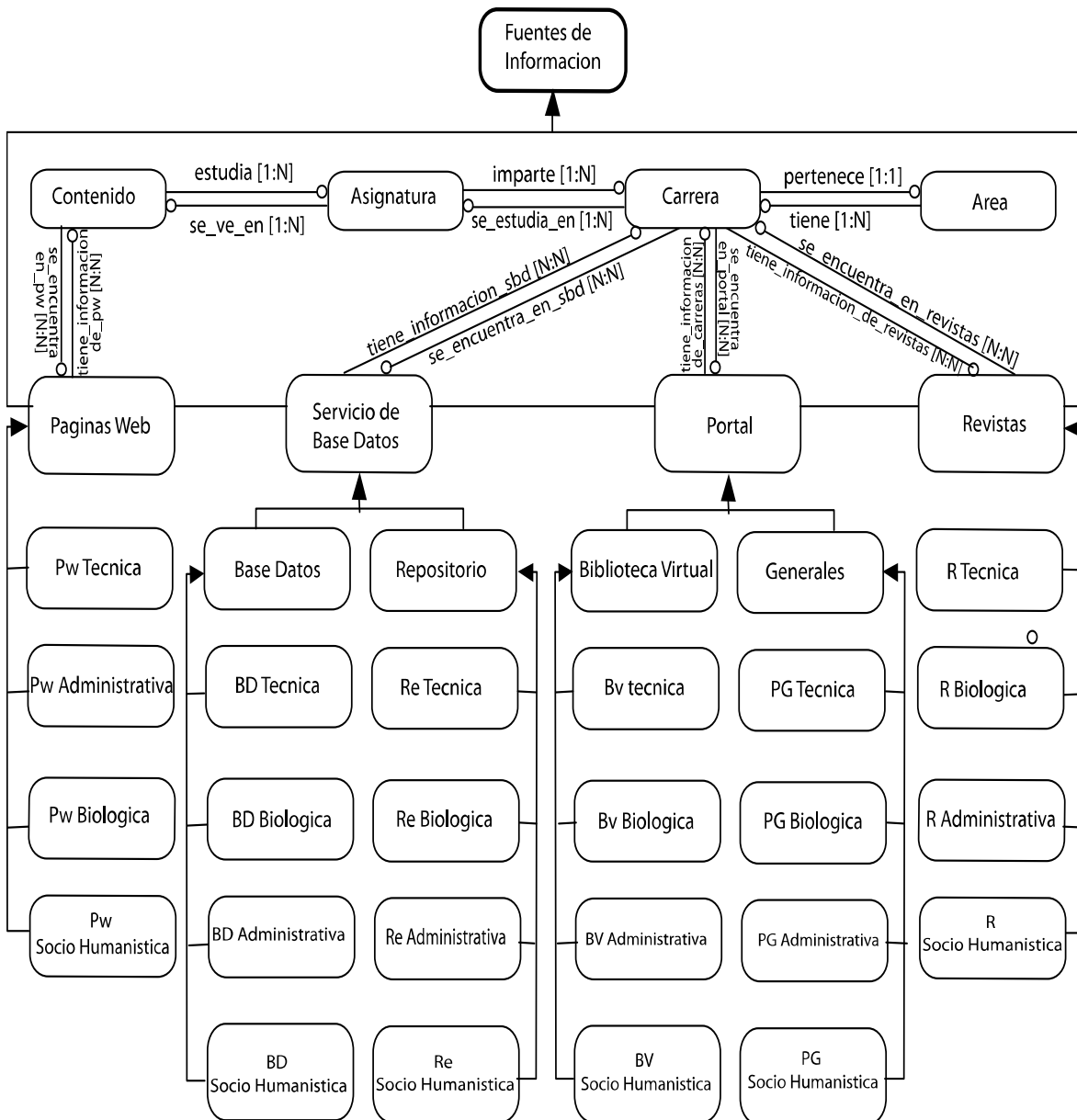


Figura 3.2. Diagrama conceptual



### 3.3.2 Conceptualización

Mediante una revisión detallada de la información disponible tanto en material digital como impreso, se trata de obtener un conocimiento consensuado del dominio de la ontología, así como las relaciones existentes entre cada uno de sus conceptos.

#### Construir un glosario de términos

- En la tabla 3.1 se presenta una parte del glosario de términos correspondiente a la ontología. (La tabla completa se encuentra en el Anexo VI).

<b>Nombre del Concepto</b>	<b>Sinónimo</b>	<b>Descripción</b>	<b>Tipo</b>
Asignatura	Materia Disciplina Curso	Materias que son dictadas en periodos académicos	Clase
Administración en Gestión Pública		Relaciones publicas y mas	Instancia
Se estudia en	Estudiar Aprende	Relación entre Asignaturas y su Contenido	Relación

Tabla 3.1 Extracto de glosario de términos

#### Construir la taxonomía de conceptos

Para realizar este paso, se toma en cuenta los términos definidos en el glosario de términos. En la siguiente figura 3.3 se muestra la taxonomía de conceptos utilizados en la ontología.

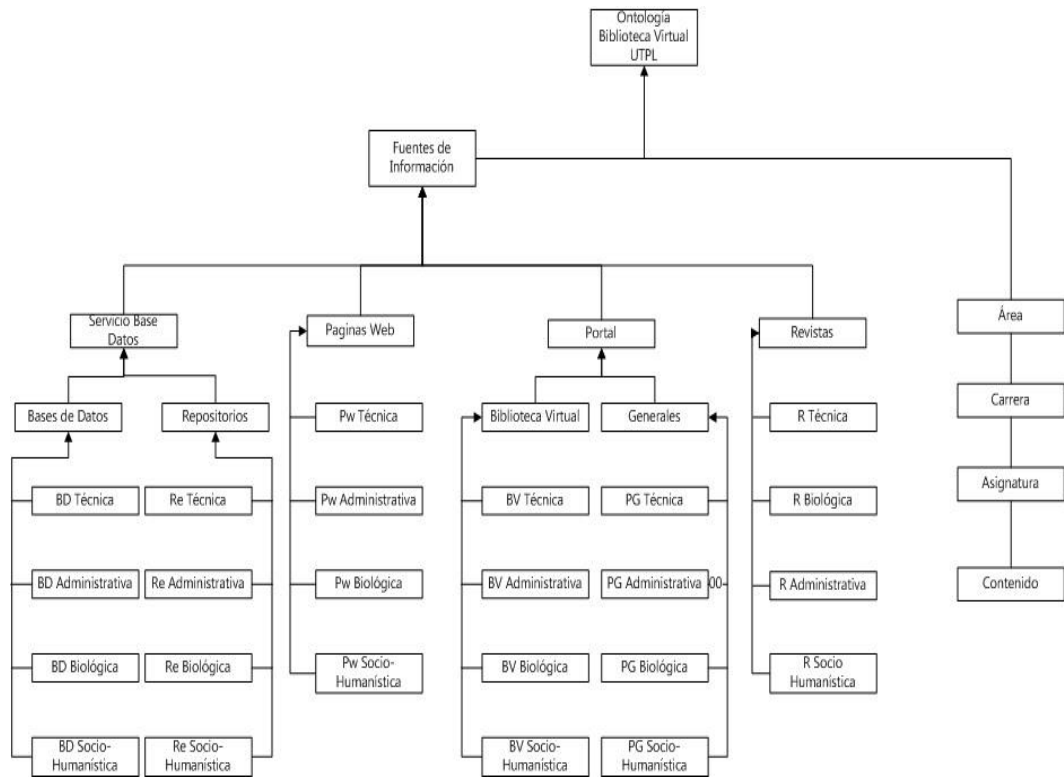


Figura 3.3 Taxonomía de conceptos

### Construir un diagrama de relaciones binarias

En la figura 3.4 se muestra las relaciones binarias utilizadas en la ontología.

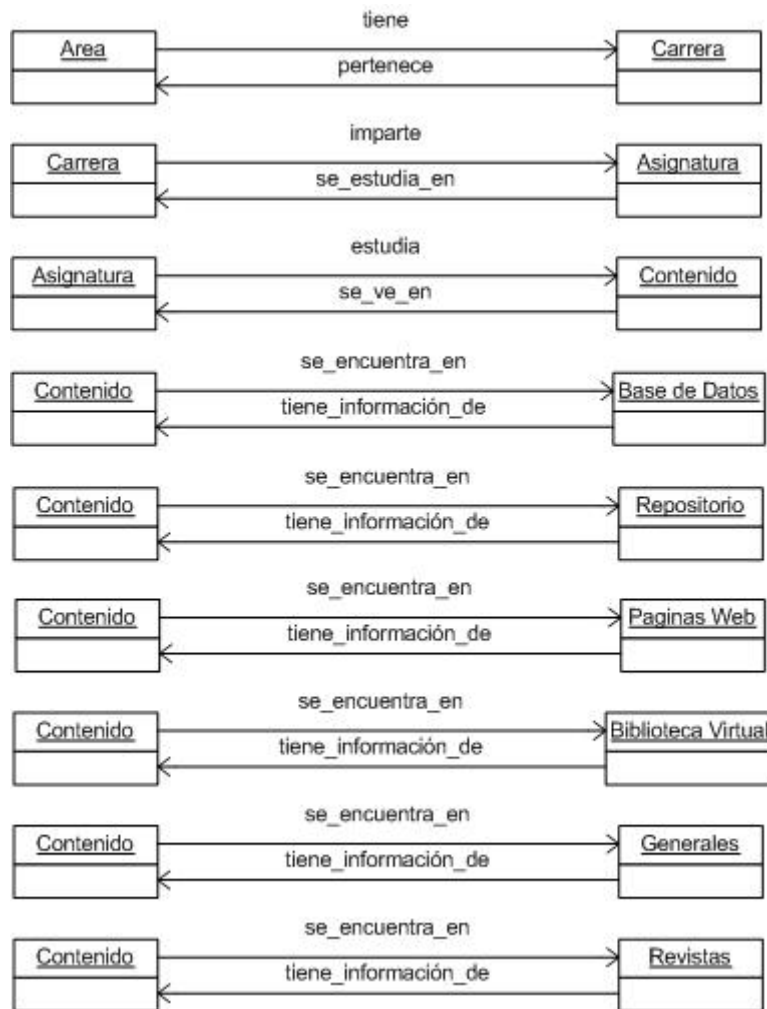


Figura 3.4 Relaciones binarias

### Construir el diccionario de conceptos

La tabla 3.2 muestra un extracto de diccionario de conceptos utilizados en la ontología. (La tabla completa se encuentra en el Anexo VII).



Nombre del concepto	Instancias	Atributos de la clase	Atributos de la instancia	Relaciones
Asignatura	Zoología General Writing III Balance de Materia y Energía Iniciación a la Lecto- estructura Archivística Escultura V Bioquímica Administración en Salud	Nombre_asignatura  oferta_académica  ciclo  modalidad  tipo	Pregrado Postgrado Pregrado y Posgrado Clásica - Abierta Virtual Semipresencia l Clásica y Abierta Troncal de Carrera Formación Básica o Comunes De Carrera Genéricas de Carrera Complementarias Libre Configuración Gestión Productiva	se_estudia_en (carreras) estudia (contenidos)

Tabla 3.2 Extracto del diccionario de conceptos



### Definir las relaciones binarias en detalle

En la tabla 3.3 se muestran las relaciones binarias que se encuentran en la ontología. Estas nos sirven para especificar el nombre, conceptos fuente y destino, cardinalidad y relación inversa.

<b>Nombre Relación</b>	<b>Concepto Origen</b>	<b>Concepto Destino</b>	<b>Cardialidad</b>	<b>Relación Inversa</b>
Tiene	Área	Carrera	1:N	Pertenece
Imparte	Carrera	Asignatura	1:N	se_estudia_en
Estudia	Asignatura	Contenido	1:N	se_ve_en
Se_encuentra_en	Contenido	Base de Datos	1:N	tiene_información_de
Se_encuentra_en	Contenido	Repositorios	1:N	tiene_información_de
Se_encuentra_en	Contenido	Páginas web	1:N	tiene_información_de
Se_encuentra_en	Contenido	Biblioteca Virtual	1:N	tiene_información_de
Se_encuentra_en	Contenido	Generales	1:N	tiene_información_de
Se_encuentra_en	Contenido	Revistas	1:N	tiene_información_de

Tabla 3.3 Relaciones binarias en detalle

### Definir los atributos de clases en detalle

La tabla 3.4 consta de los atributos en detalle, la cual tiene los siguientes elementos: nombre, concepto donde es definido, tipo de valor, valor y cardinalidad.

<b>Nombre</b>	<b>Concepto donde es definido</b>	<b>Tipo de Valor</b>	<b>Cardinalidad</b>
Nombre Asignatura	Asignatura	String	1:1
Modalidad	Asignatura	String	1:N
Tipo	Asignatura	String	1:1
Descripción	Asignatura	String	1:1
Nombre Carrera	Carrera	String	1:1



Nombre Área	Área	String	1:1
Oferta Académica	Asignatura	String	1:1
Ciclo	Asignatura	String	1:1
Idioma	Fuentes de Información	String	1:N
Acceso	Fuentes de Información	String	1:N
Descripción	Fuentes de Información	String	1:1
Búsqueda	Portal	String	1:N
Servicios	Portal	String	1:N
Issn	Revista	Integer	1:1
Entidad Editora	Revista	String	1:1
Volumen	Revista	String	1:N
Fecha de Publicación	Revista	Integer	1:N
Autor	Páginas web	String	1:N
Tema	Páginas Web	String	1:N
Disciplina	Servicio de Base de datos	String	1:N
Propietario	Servicio de Base de datos	String	1:N

Tabla 3.4 Atributos de las clases en detalle

### Definir los axiomas formales

En la tabla 3.5 se describen los axiomas formales utilizados en la ontología. En dicha tabla se describe la relación existente entre las clases, clase origen, clase destino y la respectiva regla.

Relación	Clase origen	Clase destino	Nombre regla
Tiene	Área	Carrera	Toda área tiene al menos una carrera
Tiene	Área	Carrera	Una área tiene mínimo una carrera
Pertenece	Carrera	Área	Toda carrera pertenece a un área





Pertenece	Carrera	Área	Al menos una carrera pertenece a un área
Imparte	Carrera	Asignatura	En toda carrera se imparten asignaturas
Imparte	Carrera	Asignatura	En al menos una carrera tiene asignaturas
Estudia	Asignatura	Contenidos	Todas las asignaturas estudian contenidos
Estudia	Asignatura	Contenidos	En al menos una asignatura se estudian contenidos
se estudia en	Asignatura	Carrera	Algunas asignaturas se estudian en carreras
se estudia en	Asignatura	Carrera	Al menos una asignatura se estudian en carrera
se ve en	Contenido	Asignatura	Algunos contenidos se ven en asignaturas
se ve en	Contenido	Asignatura	Al menos un contenido se ve en una asignatura
Se encuentra en pw	Contenido	Páginas web	Algunos contenidos se encuentran en Páginas web
Se encuentra en portal	Contenido	Portal	Algunos contenidos en encuentran en portal
Se encuentran en revistas	Contenido	Revistas	Algunos contenidos en encuentran Revistas
Se encuentra en sbd	Contenido	Servicio de base de datos	Algunos contenidos en encuentran en servicio de base de datos

Tabla 3.5 Axiomas formales

### Definir las reglas

Las reglas detalladas a continuación son necesarias porque describen las condiciones en las que la ontología trabajará y estas reglas constan de un antecedente y un consecuente, bajo el formato “si entonces”.

Las reglas de inferencia son las que permiten inferir (interpretar, deducir) conocimiento por parte de la ontología, de acuerdo al diseño que se haga del dominio en el cual se va a implementar dicha



ontología, es decir a partir de las relaciones que existan entre las diferentes clases

Por ejemplo una regla de inferencia sencilla sería de la siguiente forma:

Loja es una ciudad del Ecuador, Ecuador es un país del continente Americano, por lo tanto se deduce que Loja es una ciudad del continente Americano.

### REGLAS DE INFERENCIA EN LA ONTOLOGIA

Siguiendo la regla de inferencia SILOGISMO HIPOTÉTICO (SH):

$$P \rightarrow Q$$

$$Q \rightarrow R$$

$$P \rightarrow R$$

Se deduce lo siguiente:

Contenido se\_ve\_en Asignatura

Asignatura se estudia en Carrera

Contenido se\_estudia\_en Carrera

Asignatura se\_estudia\_en Carrera

Carrera pertenece Area

Asignatura pertenece Area



Carrera	imparte	Asignatura
<u>Asignatura</u>	<u>estudia</u>	<u>Contenido</u>
Carrera	estudia	Contenido
Area	tiene	Carrera
<u>Carrera</u>	<u>imparte</u>	<u>Asignatura</u>
Area	imparte	Asignatura

Como se puede ver en la regla de inferencia planteada anteriormente, ya podemos conocer a que carrera y a que asignatura pertenece determinado contenido. Entonces, las recomendaciones también abarcarán tanto a la asignatura, o si se prefiere a la carrera.

Contenido	se_estudia_en	Carrera
<u>Carrera</u>	<u>se_ecuentra_en</u>	<u>Fuente de Informacion</u>
Contenido	se_ecuentra_en	Fuente de Informacion

### 3.3.3 Formalización e Implementación

Protégé es una herramienta de software diseñada para construir ontologías. Con la ayuda de esta herramienta se puede desarrollar sistemas basados en el conocimiento, es decir capaces de tomar decisiones en problemas particulares. Además es una herramienta sencilla y potente tanto en su utilización como en portabilidad para realizar aplicaciones Web, y soporta varios frames entre los cuales están: XML Schema, RDF Schema y OWL,...

En la siguiente gráfica 3.5, se describe los conceptos básicos de la ontología para la Biblioteca Virtual de la UTPL:

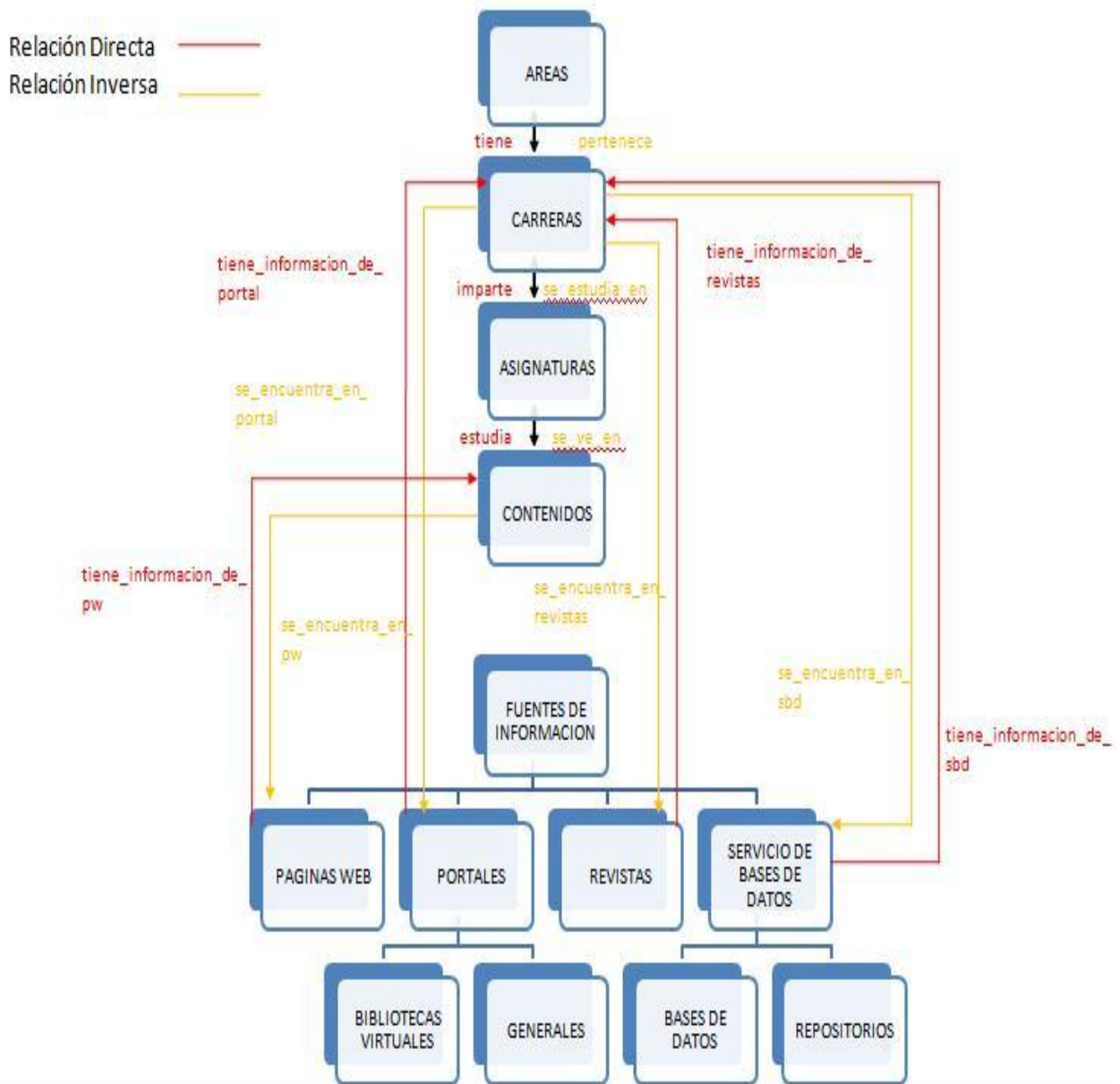


Figura 3.5 Modelo formal de la ontología

Para tener una mejor apreciación de la ontología desarrollada en la herramienta Protégé, en la siguiente figura 3.8, se puede observar el modelo lógico.

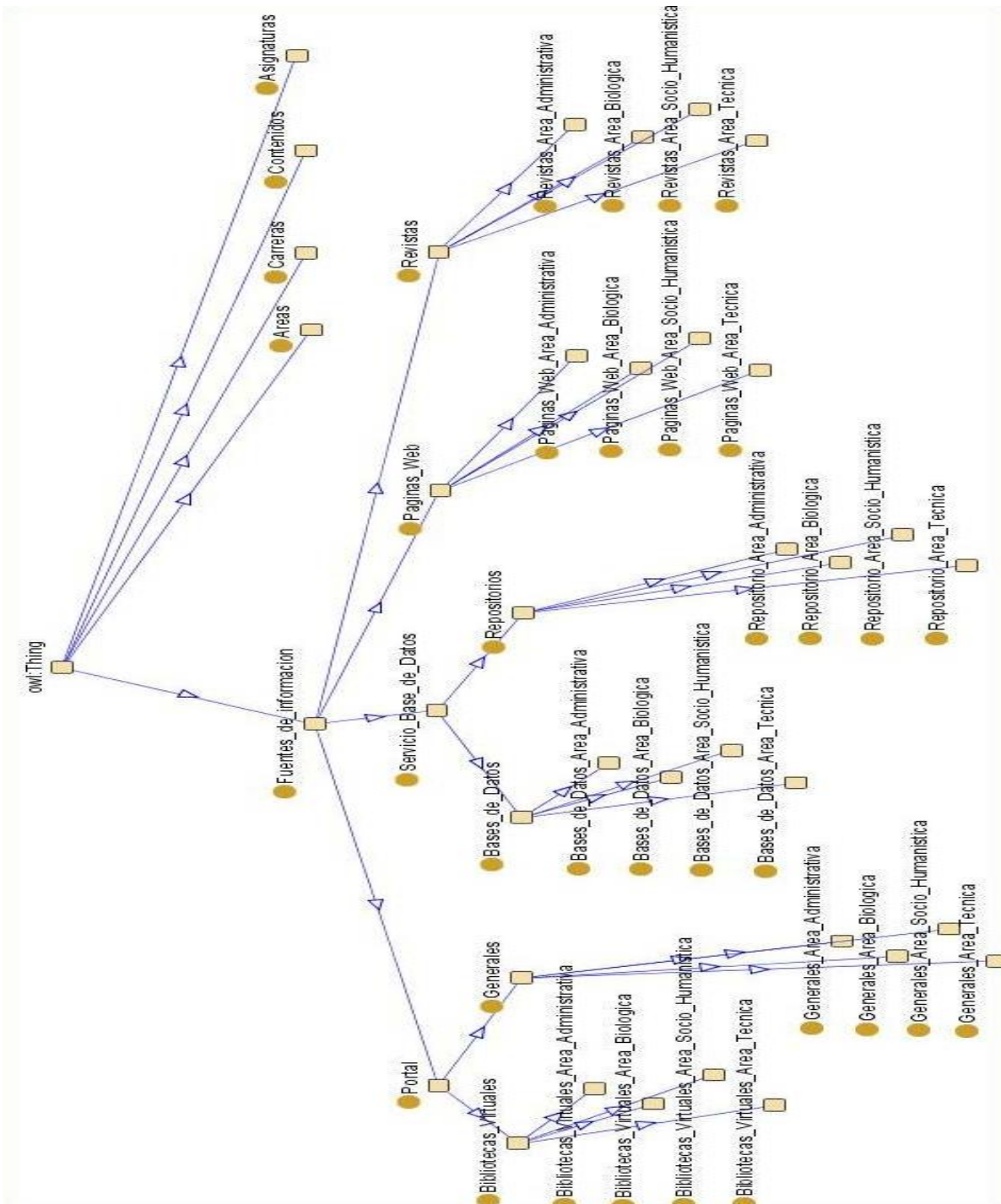


Figura 3.6 Modelo lógico



### 3.3.4 Plan de Pruebas

Para poder descubrir errores en la ontología es necesario realizar pruebas durante su diseño y construcción. Según lo manifestado por Pressman (2005), existen cuatro tipos de pruebas orientados al software como son: pruebas de componentes, de integración, de validación y de sistema.

Tratando de aplicar estas pruebas en ontologías, se utilizarán las pruebas de integración y validación. Las pruebas de integración permiten evaluar el modelo lógico considerando el nivel de representación del dominio, y las pruebas de validación consisten en evaluar los requerimientos funcionales de la ontología, a continuación el detalle de las pruebas:

#### 3.3.4.1 Pruebas de integración

**Objetivo.-** contar con un modelado que permita un funcionamiento real del modelo lógico.

**Procedimiento.-** para realizar esta prueba se toma en cuenta la evolución real que ha sufrido el modelo ontológico, durante su etapa de diseño y construcción. Para ello se crea una tabla donde se listen todos los objetos del diseño ontológico luego se los clasifica si son: clases, atributos o relaciones. A continuación, se identifica si son operativos o no, y finalmente se toma en cuenta el filtrado de resultados.



**Versión V1**

Nombre Objeto	Clase	Atributo	Relaciones	Operativo Funcional		Filtrado de Resultados	
				Si	No	Si	No
Areas	X			X			X
Modalidad	X				X	X	
Materias	X			X			X
Informacion	X			X		X	
Carreras	X			X			X
Posee			X	X		X	
Contiene			X	X		X	
Pertenece			X	X			X
Agrupar			X	X			X
Tipo		X		X		X	
Servicio		X			X		X
Url		X			X		X

Tabla 3.6 Evaluación del primer modelo ontológico

**Observación:** como se puede apreciar en la tabla 3.6, la clase Información no tiene subclases. Por otra parte, la clase Modalidad se relacionaba con materias a través de la relación *posee*. Esto ocasionaba errores ya que existen materias en las modalidades clásica y abierta con el mismo nombre aunque con diferentes contenidos.



**Versión V2**

Nombre Objeto	Clase/ Subclase	Atributo	Relaciones	Operativo Funcional		Filtrado de Resultados	
				Si	No	Si	No
Areas	X			X			X
Carreras	X			X			X
Contenidos	X			X		X	
Asignaturas	X			X			X
Recurso digital	X			X		X	
Fuentes de informacion	X						
Bases de datos	X				X	X	
Bases de datos generales	X						
Bibliotecas virtuales	X				X	X	
Portal	X				X	X	
Repositorio	X				X	X	
Servicio de bases de datos	X				X	X	
Paginas web	X				X	X	
Revistas	X						
Se estudia en			X	X			X





Estudia			X	X			X
Conforma			X	X			X
Tienen			X	X			X
Posee			X		X	X	
Se ve en			X	X			X
Pertenece			X	X			X
Imparte			X	X			X
Pertenece a			X	X			X
Esta compuesta			X		X	X	
Descripcion de		X		X			X
Propietario		X		X			X
Modalidad		X			X	X	
Tipo		X			X	X	
Nombre contenido		X		X			
Mail		X			X		X
Nombre carrera		X		X			X
Idioma		X		X			X
Busqueda		X		X		X	
Servicios		X		X			X
Link		X		X			X



Descripcion		X			X	X	
Volumen		X			X	X	
Autor		X		X			X
Entidad editora		X		X			X
Nombre area		X		X			X
ISSN		X		X			X
Nombre asignatura		X		X			X
Oferta academica		X		X			X
Formato		X			X	X	
Fecha de creación		X		X			X
Ciclo		X			X	X	
Acceso		X		X			X

Tabla 3.7 Evaluación del segundo modelo ontológico

**Observación:** en este modelo ontológico la clase Recurso Digital tiene nueve subclases, de las cuales Fuentes de información, Base de datos, Base de datos generales y Servicio de base de datos, tienen atributos en común, y al utilizar una sola relación los resultados brindados eran redundantes. También existían atributos e instancias con el mismo nombre, lo que ocasionaba duplicidad de información.



**Versión V3**

Nombre Objeto	Clase/Subclase	Atributo	Relaciones	Operativo Funcional		Filtrado de Resultados	
				Si	No	Si	No
Areas	X			X		X	
Carreras	X			X		X	
Contenidos	X			X		X	
Asignaturas	X			X		X	
Fuentes de información	X			X		X	
Paginas web		X		X		X	
Portal		X		X		X	
Revistas		X		X			X
Servicio base de datos		X		X			X
Paginas web area administrativa		X		X		X	
Paginas web area biológica		X		X		X	
Paginas web area socio humanística		X		X		X	
Paginas web area técnica		X		X		X	
Bibliotecas				X			X



virtuales							
Bibliotecas virtuales area administrativa	X			X		X	
Bibliotecas virtuales area biológica	X			X		X	
Bibliotecas virtuales area sociohumanistica	X			X		X	
Bibliotecas virtuales area técnica	X			X		X	
Generales	X			X			X
Generales area administrativa	X			X		X	
Generales area biológica	X			X		X	
Generales area socio humanisticas	X			X		X	
Generales area técnica	X			X		X	
Revistas	X			X			X
Revistas area administrativa	X			X		X	
Revistas area biológica	X			X		X	
Revistas area socio				X		X	



humanísticas	X						
Revistas area técnica	X			X		X	
Servicio Base de datos	X			X			X
Bases de datos	X			X		X	
Bases de datos area administrativa	X			X		X	
Bases de datos area biológica	X			X		X	
Bases de datos area socio humanística	X			X		X	
Bases de datos area técnica	X			X		X	
Repositorios	X			X			X
Repositorios area administrativa	X			X		X	
Repositorios area biológica	X			X		X	
Repositorios area socio humanística	X			X		X	
Repositorio area técnica	X			X		X	



Tiene información de sbd			X	X			X
Estudia			X	X		X	
Se estudia en			X	X		X	
Tiene información de portal			X	X			X
Imparte			X	X			X
Pertenece			X	X			X
Tiene			X	X			X
Se ve en			X	X			X
Se encuentra en pw			X	X		X	
Tiene información de pw			X	X		X	
Tiene información de revistas			X	X		X	
Se encuentra en portal			X	X		X	
Se encuentra en sbd			X	X		X	
Se encuentra en revistas			X	X		X	
Idioma		X		X		X	
Busqueda		X		X		X	
Oferta academica		X		X		X	



Nombre contenido		X		X		X	
Acceso		X		X		X	
Fecha publicación		X		X		X	
Servicio		X		X		X	
Descripcion		X		X		X	
Nombre asignatura		X		X		X	
Titulo		X		X		X	
ISSN		X		X		X	
Nombre area		X		X		X	
Entidad editora		X		X		X	
Autor		X		X		X	
Disciplina		X		X		X	
Tipo		X		X		X	
Propietario		X		X		X	
Modalidad		X		X		X	
Volumen		X		X		X	
Nombre carrera		X		X		X	

Tabla 3.8 Evaluación del primer modelo ontológico

**Observación:** ontología definitiva con clases, atributos, relaciones e instancias, que permite una correcta representación del dominio de la ontología.



### Representación gráfica de la evolución ontológica

A continuación, se detalla el nivel de representación del conocimiento entre las diferentes versiones de los modelos ontológicos.

Versión	Objeto	Atributos	Relación
V1	10	3	4
V2	47	23	10
V3	73	20	14

Tabla 3.9 Resultados de las ontologías

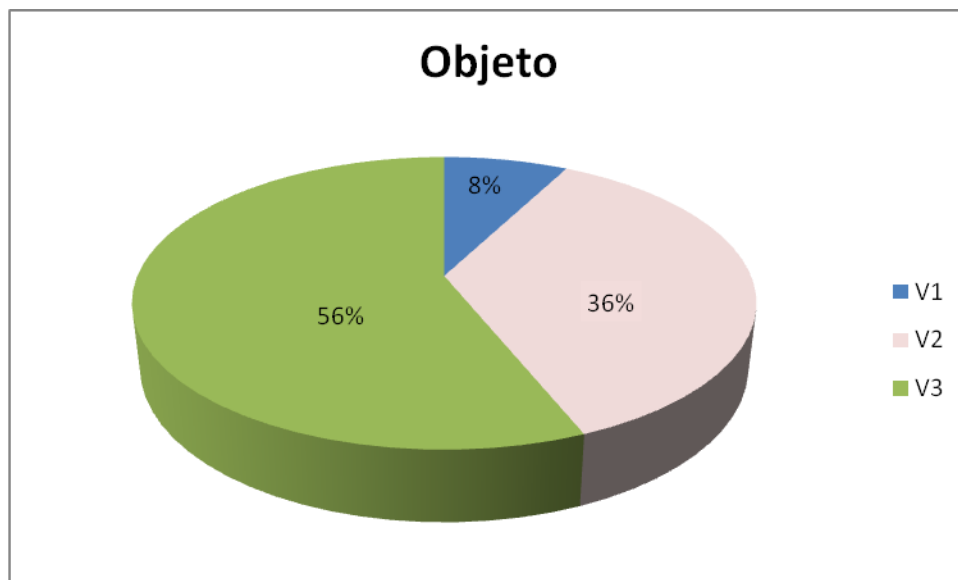


Figura 3.7 Porcentaje de representación del conocimiento en cuanto a objeto



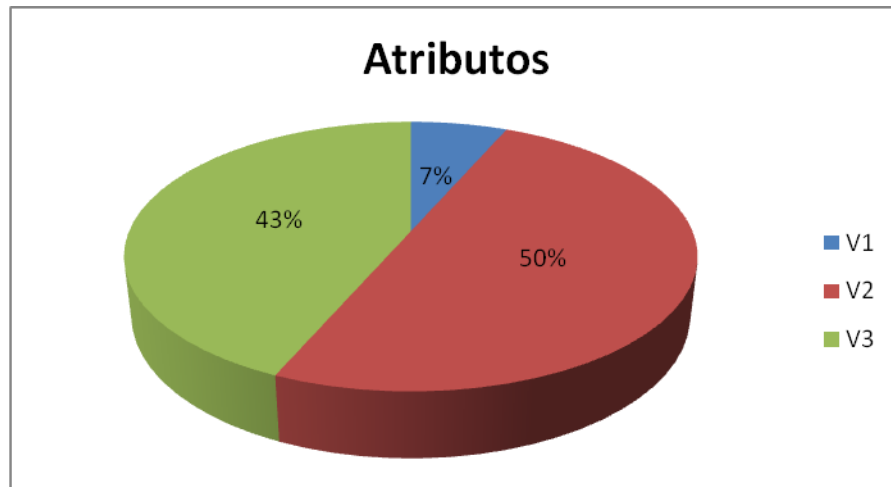


Figura 3.8 Porcentaje de representación del conocimiento en cuanto a atributos

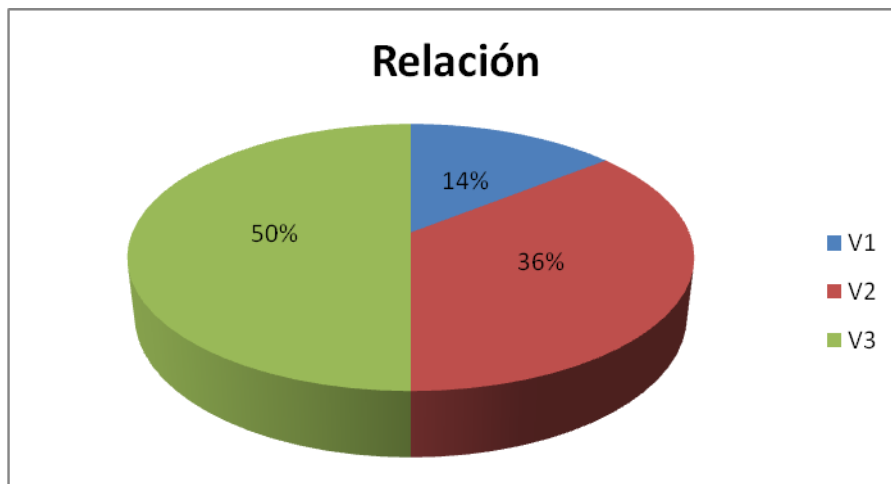


Figura 3.9 Porcentaje de representación del conocimiento en cuanto a relaciones

Como se puede observar en las figuras anteriores, a medida que se va haciendo un refinamiento de la ontología, va creciendo el número de objetos, atributos y relaciones.



### **3.3.4.2 Pruebas de validación**

**Objetivo:** determinar si la ontología cumple con los requerimientos funcionales contemplados en el proyecto.

#### **Requerimiento 1**

Las recomendaciones se las harán en base al contenido de las asignaturas.

#### **Requerimiento 2**

Los enlaces presentados al estudiante se deben extraer desde los distintos recursos digitales de la biblioteca virtual, sean estas: bases de datos, revistas, bibliotecas virtuales,...

#### **Requerimiento 3**

La herramienta seleccionada para la administración de la ontología, debe proveer funciones para la adición, edición y supresión de los elementos.

#### **Requerimiento 4**

El tiempo de respuesta para presentar las recomendaciones debe ser prudencial.

#### **Procedimiento**

Para el requerimiento 1 y 2, se procede a comprobar las relaciones existentes en la ontología, y los resultados a presentarse deben estar de acuerdo a las fuentes bibliográficas existentes en la Biblioteca Virtual, esta prueba se la realiza a nivel de desarrolladores. Para cumplir con el requerimiento 3, se utilizó la herramienta Protégé, que



permite la edición de ontologías. Para el requerimiento 4, se realizó pruebas para determinar la aceptación del usuario final, llegando a determinarse la aceptabilidad, facilidad de uso, confiabilidad y tiempo de respuesta al momento de utilizar la ontología, para esto se utiliza una encuesta que se aplica a cinco estudiantes.

### Requerimiento 1 y 2

Carrera	Contenido	Fuentes de información							
		SBD		Portal		Revistas		PW	
		Si	No	Si	No	Si	No	Si	No
Información extraída para las carreras		X		X		X			
	Información extraída de contenidos							X	

Tabla 3.10 Resultados requerimientos 1 y 2

### Requerimiento 4

En la siguiente tabla se muestra los usuarios que realizaron los casos de pruebas:

Nombre	Carrera	Modalidad	Materia	Cento asociado
Jessica Eulalia Jima Narvaez	Informática	Abierta	Sistemas IV	Cariamanga
Jose Fernando Llerena Gomez	Informática	Abierta	Matematicas discreta	Loja
Francisco Marcial Acosta	Economia	Abierta	Macroeconomia I	Quito



Mejia				
Italo Adrian Perez Hurtado	Economia	Abierta	Microeconomia I	Cuenca
Victor Felix Barrezueta Bermeo	Economia	Abierta	Matematicas	Ambato

Tabla 3.11 Lista de usuarios

Los resultados obtenidos se muestran a continuación:

	<b>Excelente</b>	<b>Muy Bueno</b>	<b>Bueno</b>	<b>Regular</b>
Tiempo en espera de solicitud	20%	40%	40%	0%
La información es confiable	25%	50%	25%	0%
Adecuada interfaz de presentación al usuario	60%	20%	20%	0%
Información útil en la interfaz de usuario	40%	40%	20%	0%

Tabla 3.12 Resultados requerimiento 4

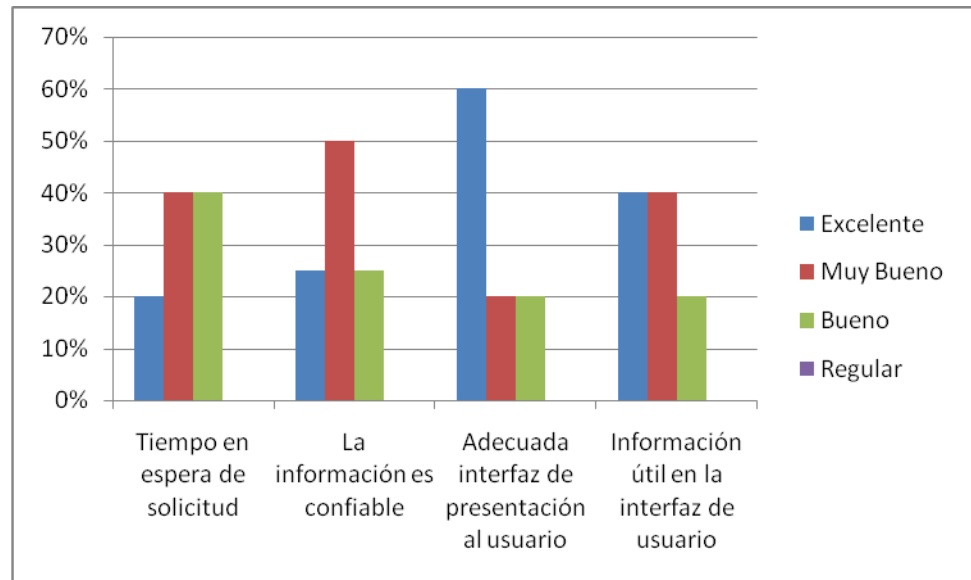


Figura 3.10 Resultados obtenidos

Como se puede observar en la figura 3.10, existe un muy bueno nivel de aceptación por parte de los usuarios, al igual que el nivel de información, con respecto al tiempo de espera, este depende del tipo de conexión que se esté utilizando. Debemos recordar que las personas encuestadas pertenecen a diferentes partes del país.

### 3.3.5 Aplicación de OntoQA

En la ontología se aplicaron las siguientes métricas, con la finalidad de comprender y controlar los elementos como: clases, instancias, relaciones,... que constituyen la base de conocimientos y determinar el nivel de representación de la ontología.



### Métricas de estructura

#### a) Riqueza de relaciones

Como se puede observar en la tabla 3.13, el resultado es cercano al 0.4, lo que indica la existencia baja de relaciones entre diferentes clases de la ontología, esto se da debido a que esta ontología representa un dominio específico. La existencia de relaciones entre diferentes clases son pocas pero suficientes para realizar la representación ontológica.

Ontología	Relaciones non Is-a	Relaciones Is-a	RR
<b>Biblioteca Virtual</b>	14	24	0.368

Tabla 3.13 Resultado de relaciones de riqueza

#### b) Riqueza de atributos

Como se menciona anteriormente, los atributos le dan características adicionales a clases e instancias. La tabla 3.14 indica que el número promedio de atributos por clase es, 4.6, una de las causas de este valor bajo es porque la ontología representa un dominio muy pequeño, lo que causa la existencia baja de atributos por clase, por ello se tomó en cuenta únicamente las clases principales que tienen atributos, descartando las clases que heredan atributos, evitando así repetición de clases con atributos heredados.

Ontología	Atributos	Número de clases	RA
<b>Biblioteca Virtual</b>	42	9	4.6

Tabla 3.14 Riqueza de atributos en la ontología



### c) Riqueza de herencia

Mediante la riqueza de herencia, en la tabla 3.15 se visualiza la estructura de la ontología, cuyo valor es 2.64, este valor es bajo, por lo tanto tiene un pequeño valor de subclase por clase, con un nivel de profundidad reducido. Esto indica que es una ontología horizontal, es decir posee información general sobre el dominio representado, ya que no se cuenta con información específica de la base de conocimiento, o un nivel elevado de detalle.

Ontología	Clase	Numero de Subclase por clase	Total Clases	IRs
<b>Biblioteca Virtual</b>	Owl:Thing	5	<b>14</b>	<b>2.64</b>
	Fuentes de	4		
	Área	0		
	Contenido	0		
	Asignatura	0		
	Carrera	0		
	Portal	2		
	Biblioteca Virtual	4		
	Generales	4		
	Servicio de base	2		
	Repositorio	4		
	Bases de Datos	4		
	Pag Web	4		
	Revistas	4		
	<b>Sumatoria</b>	<b>37</b>		

Tabla 3.15 Riqueza de herencia en la ontología

### Métricas de las instancias

#### a) Riqueza de clase

La riqueza de clase de la ontología es del 70%, mediante esto se puede decir que existe una aceptable riqueza de clase en la base de conocimiento, es decir, aunque exista una riqueza de relaciones media y la ontología posea información general del



dominio (como se vio en las medidas anteriores), esta tiene una alta representación del dominio del conocimiento.

<b>Ontología</b>	<b>Clases instanciadas</b>	<b>Clases existentes</b>	<b>CR</b>
<b>Biblioteca Virtual</b>	26	37	70%

Tabla 3.16 Riqueza de clases

**b) Promedio de instancias en todas las clases de la ontología**

Mediante este indicador en la tabla 3.17, se puede observar el promedio de población por clase en la ontología, con un valor, 15.21, este resultado es bajo debido a dos razones:

- Existen dos clases asignatura y contenido que no están instanciadas en su totalidad, la primera posee cuatro instancias, y la clase contenido aunque tiene un valor que supera los cien elementos es bajo, al momento que se instancien todas las asignaturas, su valor se incrementará.
- Y la segunda, las instancias de la clase área siempre será bajo, ya que la universidad posee solo cuatro áreas académicas, en cuanto a las fuentes de información estas siempre serán bajas ya que es información que casi no tiene la capacidad de variar.





Ontología	Clases	Número de Instancias	Clases existentes	CR
<b>Biblioteca Virtual</b>	Area	4		
	Asignatura	5		
	Carrera	30		
	Contenido	155		
	Paginas Web Area Administrativa	30		
	Paginas Web Area Tecnica	27		
	Bibliotecas Virtual Area	11		
	Bibliotecas Virtual Area Biologica	13		
	Bibliotecas Virtual Area Socio Humanistica	26		
	Bibliotecas Virtual Area Tecnica	12		
	Generales Area Administrativa	5		
	Generales Area Biologica	11		
	Generales Area Socio Humanistica	7		
	Generales Area Tecnica	5		
	Revistas Area Administrativa	7		
	Revistas Area Biologica	3		
	Revistas Area Socio Humanistica	13		
	Revistas Área Técnica	4		
	Base de datos Área Administrativa	34		
	Base de datos Área Biológica	53		
	Base de datos Área Socio	39		
	Base de datos Área Técnica	31		
	Repositorio Área Administrativa	7		
Repositorio Área Biológica	12			
Repositorio Área Socio Humanística	13			
Repositorio Área Técnica	10			
<b>Sumatoria</b>		<b>563</b>	<b>37</b>	<b>15.2</b>

Tabla 3.17 Promedio de instancias por clase

### Métricas de clase

#### a) Importancia

En la siguiente tabla 3.18, se puede observar de forma clara donde faltan instancias, es decir existe poca población para realizar una efectiva representación del conocimiento. Al aplicar la fórmula se denota lo siguiente:

- La clase Área tiene un valor bajo, debido a que la universidad tiene cuatro áreas académicas de estudio, y



Asignatura, posee solo cinco instancias, las que se incrementaran al momento de instanciar con todas las asignaturas dictadas por área académica. Todo esto se soluciona al momento de poner en producción la ontología, con información real o total.

- La clase Fuentes de Información tiene un valor elevado, esto se da porque está instanciada en su totalidad.

<b>Ontología</b>	<b>Clase</b>	<b>Número de Instancias</b>	<b>Total Instancias</b>	<b>CR</b>
Biblioteca Virtual	Área	4	563	0.007
	Carrera	30	563	0.053
	Contenido	155	563	0.275
	Asignatura	5	563	0.008
	Fuentes de Información	369	563	0.655

Tabla. 3.18 Importancia de la ontología

#### **b) Riqueza de la herencia**

En la siguiente tabla 3.19, se puede apreciar los valores al aplicar esta métrica. Estos resultados son bajos, ya que la base de conocimiento de la ontología representa un dominio específico. Tomando en cuenta que las clases con nodos terminales (hojas), son las que tienen un mayor nivel de agrupamiento de subclase por clase, en los niveles superiores el nivel de agrupamiento varía de acuerdo con el número de herencia que posean.



<b>Ontología</b>	<b>Clases</b>	<b>Número de subclase por árbol</b>	<b>Numero de nodos por subárbol</b>	<b>IRc</b>
Biblioteca Virtual	Fuentes de Información	32	9	3.55
	Portal	10	3	3.33
	Biblioteca Virtual	4	1	4
	Generales	4	1	4
	Servicio de base de datos	10	3	3.33
	Repositorio	4	1	4
	Bases de Datos	4	1	4
	Pág. Web	4	1	4
	Revistas	4	1	4

Tabla 3.19 Representación de riqueza de herencia

### **c) Relaciones de riqueza**

En el cuadro 3.20, se puede observar de manera clara el nivel de relaciones utilizadas por las clases. Las clases Asignatura y Area son las que tienen el mayor número de relaciones utilizadas por las instancias entre las que conforman la base de conocimiento, en contraste con las clases Pág. Web, Revistas, Servicio de base de datos y Portal, las que tienen relaciones únicamente definidas y no utilizadas debido a la naturaleza de la ontología. Tomando en cuenta el contexto de la ontología, ya que es de dominio, se puede concluir que al momento de realizar la extracción de la información para recomendar un enlace, las clases más utilizadas son Area y Asignatura.



<b>Ontología</b>	<b>Clases</b>	<b>Relaciones Definidas</b>	<b>Relaciones Utilizadas</b>	<b>RRc</b>
Biblioteca Virtual	Área	2	2	1
	Carrera	10	2	0.5
	Asignatura	4	4	1
	Contenido	4	1	0.25
	Pág. Web	2	0	0
	Revistas	2	0	0
	Servicio de base de datos	2	0	0
	Portal	2	0	0

Tabla 3.20 Representación relaciones de riqueza.



# CAPITULO IV

## **RECOMENDACIONES DE ENLACES DE LA BIBLIOTECA VIRTUAL**



## **4 RECOMENDACIONES DE ENLACES DE LA BIBLIOTECA VIRTUAL**

### **4.1 Introducción**

En este capítulo, se hablará de la forma como se ha integrado la ontología con el Entorno Virtual de Aprendizaje (EVA) de la UTPL.

La herramienta de construcción empleado, es Protégé con su formato de archivo RDF<sup>59</sup>, conjuntamente con el lenguaje de consultas, llamado RDQL<sup>60</sup>.

Además se describirá el módulo que fue creado en la interfaz para poder aplicar la ontología y se hablará de la importancia de obtener el pènsun académico del usuario del EVA, ya que partir de éste, se puede ofrecer una oferta de recomendaciones.

Complementando lo anteriormente dicho, se debe señalar que el aspecto principal para que entre en funcionamiento la ontología es la materia, ya que las recomendaciones se presentarán de acuerdo a la carrera que pertenezca dicha materia y a su variedad de contenidos.

### **4.2 Lenguaje de consultas empleado en la aplicación**

El lenguaje de consulta que se empleo para el desarrollo de esa aplicación fue RDQL, ya que es un lenguaje que sirve para consultar documentos RDF utilizando una sintaxis similar al lenguaje SQL.

#### **Sintaxis**

```
SELECT vars  
FROM documents  
WHERE Expressions
```

---

<sup>59</sup> Resource Description Framework

<sup>60</sup> RDF Data Query Language



*AND Filters*

*USING Namespace declarations*

SELECT.- permite definir una lista de variables que retornará la consulta. A cada variable se la escribe anteponiéndole el signo de interrogación (?), y puede ser de tipo alfanumérico combinada con un guión (\_).

FROM.- permite especificar entre signos (<doc.rdf>) la ruta o URL del documento RDF a consultar.

WHERE.- Esta es la parte más importante de la expresión RDQL, donde se indica las limitaciones que la tripleta RDF (sujeto, predicado, objeto) debe cumplir para que sean devueltas en la consulta. El sujeto, predicado y objeto pueden ser una variable o un URI.

AND.- especifica las expresiones booleanas, expresiones de igualdad, etc.

USING.- declara todos los *namespace* para las propiedades RDF, además se utiliza conjuntamente con el prefijo *FOR*

### **4.3 Funcionamiento de la aplicación**

Para el correcto funcionamiento de la ontología, se debe realizar lo siguiente:

- Cuando un usuario hace clic sobre determinada materia o asignatura que cursa, esta tiene un identificador denominado COURSEID, este es un valor numérico que tiene la materia en cada



ciclo académico, es único, no se repite y está almacenado en la tabla mdl\_course\_utpl.

- A partir del courseid de la asignatura, se podrá obtener su PASID, que es el número como se identifica a las diferentes materias en la UTPL. Para obtener su valor realizamos una consulta a la tabla señalada anteriormente, de la siguiente manera:

```
$pasid=get_record("course_utpl","courseid",$id);  
$idpas=$pasid->pasid;
```

- Y cuando determinado usuario ingrese a cualquiera de las asignaturas en las que se encuentra matriculado, se le mostrará una ventana emergente, que es la que tiene las recomendaciones académicas que se le ofrecen por intermedio de la ontología (ver su código en Anexo II).

The screenshot shows a web browser window with two tabs. The active tab is titled 'Recomendaciones Académicas - Mozilla Firefox' and displays a URL: [http://evadev.utpl.edu.ec/eva19/course/Recomendations/recomendador\\_definitivo.p](http://evadev.utpl.edu.ec/eva19/course/Recomendations/recomendador_definitivo.p). The page content includes a header with the university logo and name, and a list of recommended items under the 'Por Contenidos' tab. The items are:

- Literal(" Aplicaciones y funciones lineales
- " )Literal(" Algebra Lineal
- " )Literal(" Algebra Lineal
- " )Literal(" Funciones lineales
- " )Literal(" Arbol binario
- " )Literal(" Arbol expansion minimo
- " )Literal(" Desigualdades cuadraticas

A 'Buscar' button is visible at the bottom of the popup window.





- Adicionalmente se ofrece la alternativa de utilizar un buscador (ver su código en el Anexo III), que como se explica en la sección 4.4.3 Limitaciones de la solución, permite hacer otro tipo de búsquedas.

## **4.4 Modelo de comunicación Ontología – EVA**

### **4.4.1 Casos de uso**

Como se puede apreciar en la figura 4.2, se tiene a un usuario, que desea ingresar al EVA (puede ser un profesional en formación o docente investigador). Para este fin debe poseer un nombre de usuario (username) y una clave (password).

En el momento que ingresa a dicho entorno, en el explorador de internet se le mostrará las materias en las que se encuentra matriculado. Una vez que ha ocurrido esto, es cuando entra en funcionamiento la ontología, ya que para poder recomendarle sugerencias de índole académico, debemos saber a plenitud que tipo de materias cursa el usuario.

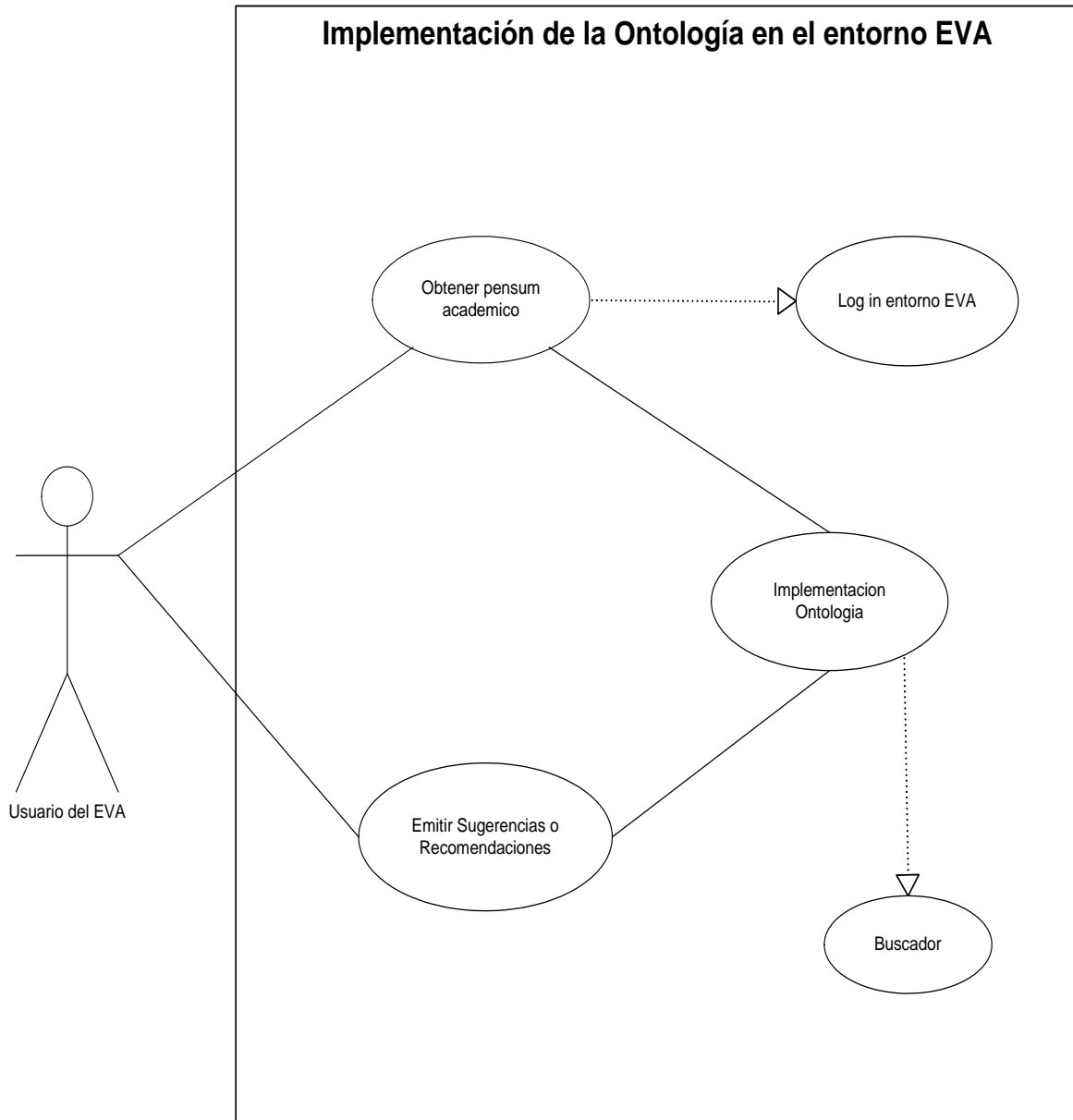


Figura 4.2. Casos de uso Modelo de comunicación Ontología-EVA



#### 4.4.2 Integración del módulo de recomendaciones al EVA

Para poner en funcionamiento la ontología y por ende las ofertas de recomendaciones, se agregó un bloque (ver Anexo IV) o módulo en el servidor de pruebas del EVA, para que cuando determinado usuario ingrese, pueda utilizarlas.

Además se brinda la opción de un buscador (ver Anexo III), donde se podrá realizar búsquedas sobre bases de datos, portales y revistas, de acuerdo a la carrera, y búsquedas sobre páginas web (recursos digitales) de acuerdo a los contenidos de la asignatura.



En la siguiente figura 4.3, se aprecia el bloque de recomendaciones creado en el entorno EVA.

Figura 4.3. Bloque de recomendaciones en el entorno EVA

#### 4.4.3 Limitaciones de la solución

Al utilizar el buscador, las consultas se deben realizar de la siguiente manera:



- Para buscar bases de datos, portales y revistas, las consultas se las deberá hacer a través de la carrera a la que pertenece determinada materia o contenido.
- Se deberá anteponer los prefijos: base, bases, revista, revistas, portal o portales, seguido del nombre de la carrera. Por ejemplo: “bases de datos para la carrera de sistemas informáticos y computación”. O si queremos visualizar todo, solo pondremos la palabra “información”. Por ejemplo: “información sobre sistemas informáticos y computación”.
- Al realizar búsquedas sobre determinada materia o contenido se deberá poner los prefijos: páginas web o página web. Por ejemplo: “páginas web sobre programación orientada a objetos de la carrera de sistemas informáticos y computación”.
- El conjunto de palabras a buscar no debe contener tildes ni signos de puntuación o interrogación o palabras que contengan la letra Ñ, debido a que la herramienta Protégé, está diseñada para ser utilizada en el idioma inglés.



## **DISCUSION**

La UTPL constantemente está investigando e implementando nuevas tecnologías para adaptarlas al proceso enseñanza-aprendizaje. Es así que pone a disposición de cada uno de sus integrantes y público en general, una biblioteca física y una biblioteca virtual. La mayoría de usuarios no le interesa o no le presta la debida atención al segundo tipo de biblioteca, limitándose las diferentes consultas a libros físicos, más no a documentos o libros digitales, publicados y certificados, y que están a disposición en la biblioteca virtual, a la cual puede acceder el personal de la universidad por intermedio del EVA a través de la etiqueta “Biblioteca Virtual”.

El inconveniente que actualmente presenta la biblioteca virtual, es la existencia de algunas fuentes de consulta sin guardar relación con las diferentes carreras y contenidos de asignaturas que se dictan en nuestra universidad.

Con el fin de solucionar en gran medida este problema, se planteó diseñar e implementar una ontología que presente al usuario información que ha sido procesada de una manera semántica, y que esté relacionada con el contenido de cada una de las asignatura. Así mismo, se utiliza un hipervínculo que permite el acceso a la ontología a través del entorno virtual.

Para obtener dicha ontología, se tuvo que seguir ciertos pasos:

1. Se realizó un estudio de cómo está estructurada la Biblioteca Virtual de la UTPL: bases de datos, bibliotecas virtuales, revistas, recursos digitales,...



2. Luego del estudio, se obtuvieron varios modelos ontológicos que surgían conforme avanzaba el desarrollo de la presente investigación. Todo esto para obtener al final un modelo que se adapte a las necesidades o requerimientos que se planeaban cumplir.
3. Una vez obtenido el modelo lógico definitivo de la ontología, se procedió a su implementación en la herramienta Protégé. Cabe indicar que al inicio, el formato de archivo de la ontología era OWL, pero posteriormente se realizó la transformación al formato RDF, debido a que el razonador implementado en la presente solución, solo funciona con este tipo de archivo.
4. Luego se procedió a su codificación e implementación con el razonador de Jena, llamado RDF API.
5. El lenguaje de codificación que se utilizó fue PHP, ya que tanto el razonador como el EVA están implementados en este lenguaje, esto con el fin de minimizar errores y no tener problemas de integración con el EVA.
6. Las consultas a la ontología se las realizó a través del lenguaje de consultas RDQL.
7. Posteriormente, se construyó un módulo o bloque en el EVA, para poder implementar las recomendaciones a través de la ontología. Además este bloque permite presentar la información que posee la ontología en base a ciertos requerimientos dados por los usuarios internos (estudiantes).



8. Por último, el razonador conjuntamente con los archivos (buscador.php, recomendador.php, block\_simplehtml) fueron configurados e incorporado al EVA.

Cabe indicar que la ontología tiene ciertas limitantes, por ejemplo: las búsquedas de bases de datos, portales y revistas, deben ser a través de la carrera. Se debe anteponer los prefijos base(s), revista(s), portal(es) conjuntamente con el nombre de dicha carrera.

Otra limitación es que las palabras a buscar, no deben contener tildes, signos de interrogación o puntuación, ni mucho menos la letra Ñ, debido a que la herramienta Protégé no acepta estos símbolos.

Además, las búsquedas a realizar no ofrecen alternativas de un corrector ortográfico. Es decir, si se realizan búsquedas de una base de datos para determinada carrera, el criterio de búsqueda no deberá contener palabras mal escritas.

Las búsquedas que se realicen sobre determinada materia o contenido, al principio se deberá poner los prefijos: pagina(s) web. Por ejemplo: “paginas web sobre programacion orientada a objetos de la carrera de sistemas informaticos y computacion”.

Por último, como trabajos futuros y hacer una mejora del modelo ontológico aplicado en éste proyecto, se debería contar con un acceso integro a los sitios donde está almacenada la información que poseen las bases de datos, bibliotecas virtuales, portales,... de la biblioteca virtual, para así poder acceder y manipular dicha información de una manera más profunda.



Para que la ontología pueda servir a todos los usuarios del EVA, esta debe ser instanciada con los contenidos de las asignaturas de todas las carreras que se imparten en la UTPL.

La información que posee la ontología es estática, es decir, las diferentes instancias de la clase Fuentes de información no puede crecer, debido a que no existe aún una herramienta que permita alimentar esta información de manera automática.





## **CONCLUSIONES**

- En la actualidad las bibliotecas virtuales cuentan con elevada cantidad de información, sin ningún criterio de orden lógico y semántico, por consiguiente la extracción de información es redundante.
- Las ontologías permiten la obtención de información de un determinado dominio, mediante el uso de clases, relaciones, propiedades, axiomas y reglas de inferencia, reduciendo así las incongruencias sintácticas y semánticas que se producen cuando se trabaja con información dispersa y sin ningún orden específico.
- Al realizar el estudio y análisis comparativo de las metodologías más utilizadas, como herramientas de representación del conocimiento, se llegó a la conclusión que Methontology es la que presenta un nivel de madurez elevado, que imita el ciclo de vida propuesto por IEEE 1074, y un nivel de formalidad y documentación que permite modelar de forma clara el conocimiento abstracto evitando redundancia y ambigüedad en su representación.
- Para que la aplicación de la ontología en un determinado dominio sea lo más satisfactorio posible, su análisis y diseño deben ser realizados meticulosamente; porque si se comete algún error en el modelo ontológico, los resultados serían inadecuados.
- Según la metodología Methontology, en la fase de conceptualización se realiza un detalle del dominio a representar, lo que implica tener acceso a toda la información a modelar. En el caso de la biblioteca virtual de la UTPL, el concepto “bases de datos”, es descrito a un nivel general, debido a



que la información a detalle, es de propiedad de los proveedores de las bases de datos.

- Al realizar un análisis comparativo entre las herramientas para diseño de ontologías se consideró que Protégé es la adecuada, porque permite un elevado nivel de representación, evitando la pérdida de conocimiento desde la contextualización al nivel de formalización.
- Cuando se requiere que una ontología se implemente en una aplicación Java, se debe trabajar con el API de Jena; mientras que, si se trabaja en una aplicación Php, se debe trabajar con el paquete RDF API de Jena, el cuál ha sido diseñado exclusivamente para ser implementado en este lenguaje de programación.
- En el proceso de desarrollo ontológico es importante la evaluación continua, ya que permite determinar falencias cuando se realiza la extracción de información. OntoQA es un método que permite evaluar la estructura de la ontología, así como el nivel de granularidad de la representación del conocimiento.
- El instanciamiento de las clases es estático, debido a que aún no existen agentes inteligentes que alimenten de manera automática la base de conocimiento de la ontología.
- Las recomendaciones de enlaces que se detallan en este proyecto, se realizan en base a asignatura y contenido, sin la utilización de alguna técnica de inteligencia artificial.



## RECOMENDACIONES

*“Y guarden estas palabras  
Que les digo al terminar:  
En mi obra he de continuar  
Hasta dársela concluida  
Si el ingenio o si la vida  
No me llegan a faltar.”*

José Hernández

La vuelta de Martín Fierro, Canto XXXIII

- Se recomienda que la organización de la información que contiene una biblioteca virtual, sea estructurada de una manera semántica con el uso de: metadatos, taxonomías y ontologías.
- Para el proceso de desarrollo de ontologías, se recomienda utilizar metodologías calificadas que permitan obtener un elevado nivel de formalización y así evitar la redundancia en la extracción del conocimiento.
- Cuando el dominio de la ontología es amplio y el diseñador tiene poca experticia, se recomienda utilizar Methontology, ya que la fase de conceptualización permite manejar una adecuada granularidad de la representación del dominio.
- Antes de diseñar, desarrollar y construir una ontología, se debe tener la información necesaria concerniente al dominio que se desee modelar. Caso contrario, no se podría establecer adecuadamente las clases, subclases, instancias, relaciones, restricciones y axiomas.
- Las herramientas que se utilicen en el diseño y construcción de la ontología, deberán seleccionarse de acuerdo al contexto de la aplicación.



- Para la evaluación ontológica, se debe tomar en cuenta el tipo de ontología y el concepto de aplicación.
- Para la implementación de la presente ontología se debe considerar que las versiones de software sean las que se encuentran descritas en el manual del programador.
- En trabajos futuros se recomienda implementar, en base a un estudio más amplio, los llamados agentes inteligentes, ya que permitirán automatizar la extracción de información y la oferta de recomendaciones.



## BIBLIOGRAFÍA

- Blas L., Delgado C., Javier D., *La ontología*. Recuperado Marzo 15, 2008, de <http://www.slideshare.net/punkinlenin/ontologa-457135/>
- Benjamins, R., Gomez, A., *Knowledge-System Technology: Ontologies and Problem-Solving Methods*. Recuperado Febrero 20, 2008, de [www.swi.psy.uva.nl/usr/richard/pdf/kais.pdf](http://www.swi.psy.uva.nl/usr/richard/pdf/kais.pdf)
- Fernández J, (2003 b). *Un Entorno de Integración de Ontologías para el Desarrollo de Sistemas de Gestión de Conocimiento*. Recuperado Enero 10, 2008, desde [FernandezBreis.pdf](#)
- Iglesias M., (2000), *Preguntas frecuentes sobre el Lenguaje de Ontologías Web (OWL) del W3C*, Recuperado Junio, 22, 2008, desde <http://www.w3c.es/Traducciones/es/SW/2005/owlfaq>
- García R., Montes A., Moreno V.,(2004) *Esquemas para Representar el Conocimiento Basado en Ontologías* Pag 1, Recuperado Junio 22, 2008, desde [http://kybele.escet.urjc.es/RedBD/Articulos/Art\\_03.pdf](http://kybele.escet.urjc.es/RedBD/Articulos/Art_03.pdf).
- Gomez A., (s.f. a) *Ontological Engineering: A State of Art*. Recuperado Enero 11, 2008, desde [www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps](http://www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps)
- Gomez A., (s.f b) *Ontological Engineering: A State of Art*. Recuperado Enero 11, 2008 de [www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps](http://www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps)



- Gomez A., (sin año c) *Ontological Engineering: A State of Art*. Recuperado Enero 11, 2008, desde [www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps](http://www.comp.rgu.ac.uk/staff/nw/ExpertUpdate/ontologies.ps)
- Gómez, L., (2008) *Bienvenidos a la Iniciativa Enlace de Bibliotecas*,. Recuperado el 20 de Febrero, 2008, desde <http://liblink.istec.org/>
- Gruber T., (2005). *Ontology*. Recuperado Enero 11, 2008, desde <http://tomgruber.org/writing/ontology-definition-2007.htm>
- Lamarca, M, (2008) *RDF* Recuperado Junio 13, 2008, desde <http://www.hipertexto.info/documentos/rdf.htm>
- Martí, Y, Planelles, E (2002), *Análisis de PROTÉGÉ –2000*, Recuperado Junio 15 2008 de <http://personales.upv.es/ccarrasc/doc/2001-2002/Protégé2000/PROTEGE.htm>
- Martin M., (2004 a), *Sistema de Catalogación de Métricas e Indicadores con Potencia de Web semántica*. Recuperado Febrero 10, 2008, desde <http://gidis.ing.unlpam.edu.ar/home/downloads/Tesis-Martin.pdf>
- Martin M., (2004 b), *Sistema de Catalogación de Métricas e Indicadores con Potencia de Web semántica*. Recuperado Febrero 10, 2008, desde <http://gidis.ing.unlpam.edu.ar/home/downloads/Tesis-Martin.pdf>
- Martin M., (2004 c), *Sistema de Catalogación de Métricas e Indicadores con Potencia de Web semántica*. Recuperado Febrero 10, 2008, desde <http://gidis.ing.unlpam.edu.ar/home/downloads/Tesis-Martin.pdf>



- Mejía, L., (2007) *Encuesta - situación actual de las bibliotecas universitarias en el contexto de la iniciativa enlace de bibliotecas liblink-istec* Recuperado Febrero 20, 2008, desde <http://liblink.istec.org/modules.php?name=Content&pa=showpage&pid=77>
- Mizoguchi R., Vanwelkenhuysen J., Ikeda M. (1995). *Task Ontology for Reuse of Problem Solving Knowledge*. Recuperado febrero 15, 2008, desde <http://www.ei.sanken.osaka-u.ac.jp/pub/miz/KBKS95.pdf>
- Muhammad F., Muhammad A., (S/N) *A Framework for Ontology Evaluation*. Recuperado Mayo 22, 2008, desde <ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-354/p59r.pdf>
- Ramos E., Núñez H., (2007). *ONTOLOGÍAS: componentes, metodologías, lenguajes, herramientas y aplicaciones*. Recuperado Julio 18, 2008, desde <http://dircompucv.ciens.ucv.ve/Documentos/RT-2007-12.pdf>
- Ramos, L. ,Gil, R.,(n.f.) *Hacia un Sistema de Información para Apoyar la Gestión de la Educación a Distancia*,. Recuperado Mayo 05, 2008 desde [www.emis.de/journals/EJDE/conf-proc/06/r1/ramos.pdf](http://www.emis.de/journals/EJDE/conf-proc/06/r1/ramos.pdf)
- Pressman, R. S. (2005). *Ingeniería del software* (6ta. Edición), Ciudad de México, México, McGraw-Hill
- Samir Tartir, I. Budak Arpinar, Michael Moore, Amit P. Sheth, Boanerges Aleman-Meza (n.f.).*OntoQA: Metric-Based Ontology Quality Analysis*. Recuperado Agosto 10, 2008, desde <http://lsdis.cs.uga.edu/library/download/OntoQA.pdf>



- Schnurr H., Sure Y., Studer R., Akkermans H., (2000 a). *On-To-Knowledge Methodology — Baseline Version* recuperado Enero 18, 2008, desde [www.aifb.uni-karlsruhe.de/WBS/ysu/publications/OTK-D15\\_v1-0.pdf](http://www.aifb.uni-karlsruhe.de/WBS/ysu/publications/OTK-D15_v1-0.pdf)
- Schnurr H., Sure Y., Studer R., Akkermans H., (2000 b). *On-To-Knowledge Methodology — Baseline Version*. Recuperado Enero 18, 2008, desde [www.aifb.uni-karlsruhe.de/WBS/ysu/publications/OTK-D15\\_v1-0.pdf](http://www.aifb.uni-karlsruhe.de/WBS/ysu/publications/OTK-D15_v1-0.pdf)
- Strasunskas D., Tomassen S., (n.f). *Web Search Tailored Ontology Evaluation Framework*. Recuperado Julio 25, 2008, desde. <http://folk.ntnu.no/steint/papers/2007-WET-EvOQS.pdf>
- Sowa J., (2000). *Building, Sharing, and Merging Ontologies*. Recuperado Febrero, 20, 2008, desde <http://www.jfsowa.com/ontology/ontoshar.htm>
- Tallarico M., *Uso de ontologías en tareas de recupero de información*, (2008). Recuperado Septiembre, 29, 2008 de <http://usuarios.fceia.unr.edu.ar/~tallaric/tesina/presentacion.pdf>
- Taly, S., & Ariel, F. (2007), *Bibliotecas digitales en Internet*. Recuperado Diciembre 13, 2008, desde [www.ua.es/.../cursos\\_programados/2007/otrasconvocatorias/documentacion/Bibliotecas\\_digitales\\_en\\_Internet.pdf](http://www.ua.es/.../cursos_programados/2007/otrasconvocatorias/documentacion/Bibliotecas_digitales_en_Internet.pdf)
- Uschold F., Gruninger M., (1996) *Ontologies: Principles, Methods and Applications*. Recuperado Marzo, 20, 2008, desde





<http://www.aiai.ed.ac.uk/project/pub/documents/1996/96-ker-intro-ontologies.ps.gz>

- Uschold M., King M., (1995). *Towards a Methodology for Building Ontologies*. Recuperado Marzo 10, 2008, desde <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=FB71F5B8B122DB99F9FC332B96EA1F3F?doi=10.1.1.55.5357&rep=rep1&type=pdf>
- Valencia, R., (2005a), *Un Entorno para la Extracción Incremental de Conocimiento desde Texto en Lenguaje Natural*. Recuperado Febrero, 20, 2008, desde [www.tesisenxarxa.net/TESIS\\_UM/AVAILABLE/TDR-1123105-140512//rvalencia.pdf](http://www.tesisenxarxa.net/TESIS_UM/AVAILABLE/TDR-1123105-140512//rvalencia.pdf)
- Valencia, R., (2005b), *Un Entorno para la Extracción Incremental de Conocimiento desde Texto en Lenguaje Natural*. Recuperado Febrero, 20, 2008, desde [www.tesisenxarxa.net/TESIS\\_UM/AVAILABLE/TDR-1123105-140512//rvalencia.pdf](http://www.tesisenxarxa.net/TESIS_UM/AVAILABLE/TDR-1123105-140512//rvalencia.pdf)



# **ANEXOS**



## **ANEXO I: MANUAL DEL PROGRAMADOR**

### **1. Descripción de la herramienta Protégé**

Es un editor de ontologías que permite desarrollar sistemas basados en el conocimiento. Permite definir las clases, sus atributos e instancias, además que pueden ser desarrolladas en los lenguajes para ontologías como OWL y RDF. La versión utilizada en este proyecto es Protégé 3.3.1

Características: según Martí et al. (2002) Protege tiene las siguientes características:

- Posee una interfaz amigable para el usuario.
- Permite definir una jerarquía de clases que definen un tema o dominio en particular.
- Facilita el trabajar con clases e instancias.
- Además de poseer una interfaz sencilla, posee un entorno donde se puede visualizar en modo grafico la construcción de la ontología.

### **2. Ontología para la Biblioteca Virtual de la UTPL**

#### **2.1. Creación de la Ontología**

Para el presente proyecto de tesis, se ha seleccionado la herramienta Protégé, ya que nos ofrece realizar tanto el análisis como la construcción en un solo entorno. Una vez realizado el análisis del modelado de la ontología, se procede a realizar su construcción. Para esto se debe realizar los siguientes pasos:



### 2.1.1. Creación del archivo OWL/RDF

Hacemos clic en el menú **File**, luego en **New Project**, y aparecerá en pantalla un asistente que nos ayudará en la creación del proyecto.

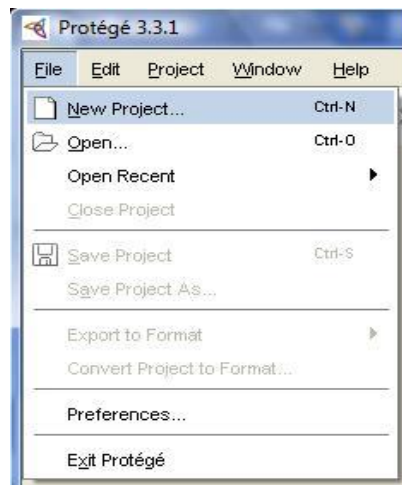
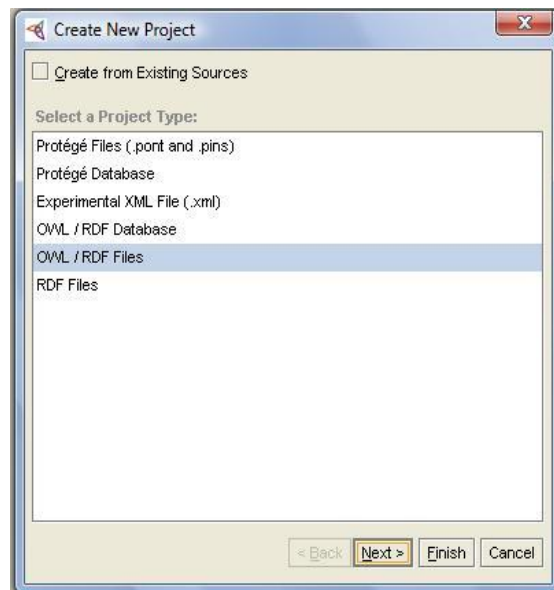


Figura 1





Escogemos como tipo de proyecto al formato **OWL/RDF file**, damos clic en **Next**

Figura 2

Ignoramos la siguiente pantalla, haciendo clic en **Next**. Luego nos aparecerá una nueva pantalla, en donde escogemos el perfil del lenguaje. Por defecto se seleccionará automáticamente **OW DL**, luego damos clic en **Next**.



Figura 3

Acto seguido nos aparecerá una pantalla donde debemos escoger la vista de clases OWL (**OWL Classes View**), por defecto se seleccionará **Logic View**, y damos clic en **Finish**



Figura 4

Y Listo. Hemos creado un proyecto OWL/RDF. Si deseamos darle un nombre al proyecto, presionamos **Ctrl+S**, y nos aparecerá una ventana, donde debemos seleccionar el lugar donde guardaremos el archivo y su nombre. Luego clic en **OK**

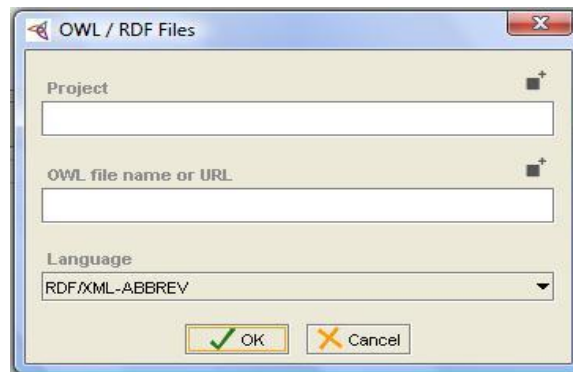


Figura 5

A continuación se detalla cómo crear en Protégé el modelado de la ontología.



### 2.1.2. Creación de Clases

Para la creación de las clases y subclases, nos situaremos en la pestaña **OWL Classes**, como se muestra en la flecha a continuación:



Figura 6

Después damos un clic en el botón que se señala con la flecha. Esto



es para crear una clase(s):



Figura 7

La cantidad de clases que deseemos lo realizamos de acuerdo a nuestro modelo ontológico, en este caso, las clases creadas son las siguientes:

Figura 8

A veces una clase puede contener una o varias subclases, entonces para crear una subclase, nos situamos en la clase que deseemos que tenga subclases y damos clic en el botón que señala la flecha en la figura 9:





Figura 9

Y como se puede apreciar en la figura anterior, en nuestra ontología ya hemos creado un sin número de subclases.

### 2.1.3. Creación de Propiedades Objetc y Datatype

#### 2.1.3.1. Propiedad Object

Una propiedad Object es aquella que nos permite relacionar dos o más clases o subclases entre sí. Tiene un dominio (Domain) que es la(s) clase(s) de donde parte la relación y un rango (Range) que es la(s) clase(s) donde termina dicha relación.



Por ejemplo tenemos 2 clases **Padres** e **Hijos** y la propiedad Object **tienen**, entonces la clase **Padres** (Domain) **tienen Hijos** (Range).

Para crear las propiedades Object se realiza lo siguiente:

Nos situamos en la pestaña **Properties** >> **Object** como lo indica la flecha:

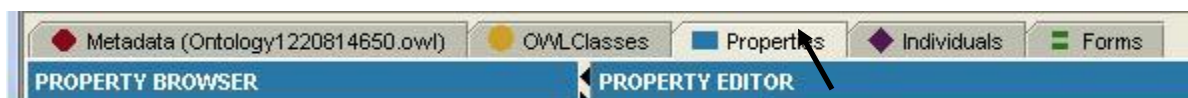


Figura 10

Para crea una propiedad Object, damos clic en el botón donde indica la flecha:

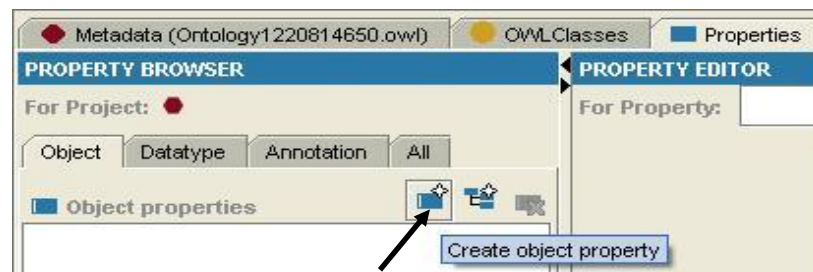


Figura 11

De igual forma, en nuestra ontología, tenemos varias relaciones con sus respectivos dominios y rangos. Para poner el nombre a una relación, nos situamos al lado derecho en **Property Editor**, en **For Property** e introducimos el nombre que deseamos. Y para crear varias subrelaciones, basta con hacer clic en el botón donde indica la flecha:

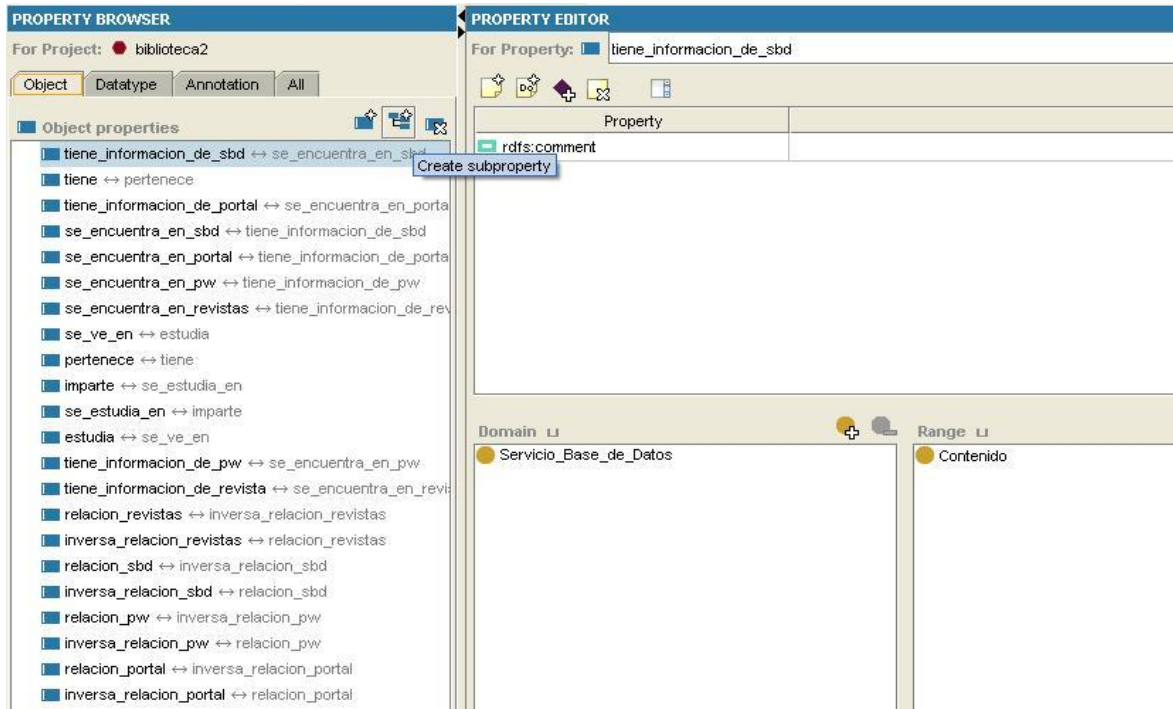
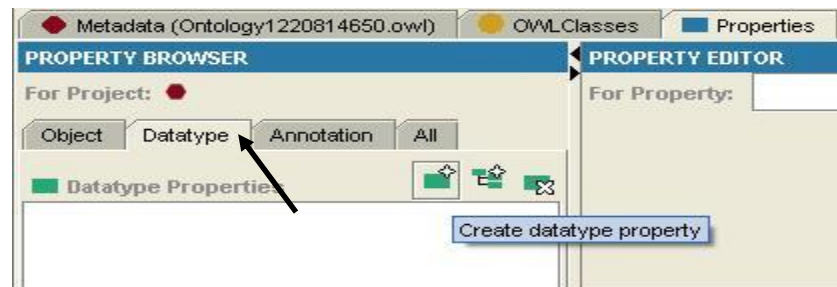


Figura 12

### 2.1.3.2. Propiedad Datatype

Una propiedad Datatype son atributos o características que posee una clase, y tiene valores del tipo float, integer, string,...



Para la creación de una propiedad Datatype, nos situamos en la pestaña **Properties** >> **Datatype**, y damos clic en el botón donde indica la flecha:

Figura 13

Y para poner el nombre a un atributo de una clase, nos situamos en **Property Editor**, en **For Property**.

De la misma manera que una propiedad Object, la propiedad Datatype también tiene:

Un dominio (**Domain**) que es la clase, clases, subclase o subclases que tendrán el atributo dado.

Un rango (**Range**) que es el tipo de valor que tendrá el atributo, este puede ser float, integer, string, boolean, Datetime, date, time.

Y valores permitidos (**Allowed values**) que son los valores específicos que poseerá el atributo.



En nuestra ontología poseemos varios atributos dados a las clases. Como lo muestra la siguiente figura, tenemos a la clase Fuentes de Información (Domain) con un rango (Range) del tipo String con los valores permitidos Español, Inglés y Varios Idiomas.

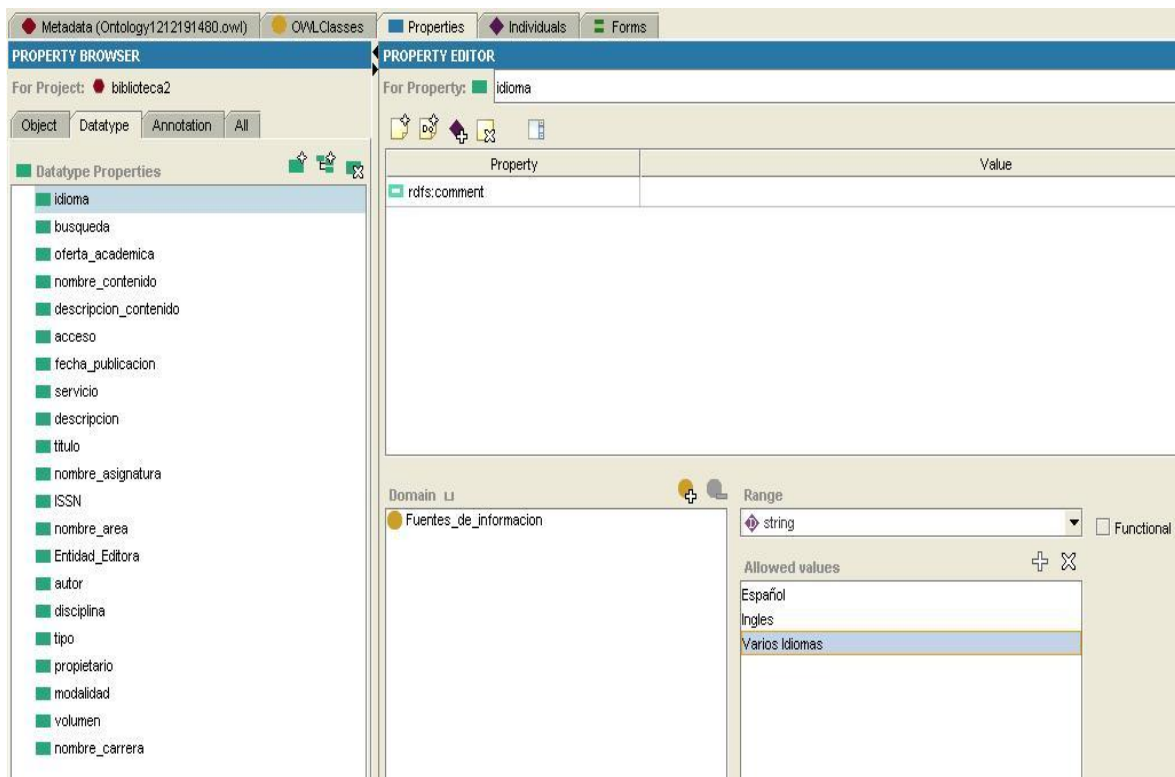


Figura 14

#### 2.1.4. Creación de Instancias

Las instancias son los nombres de los valores que tiene una clase. Para realizar esto, nos situamos en la pestaña **Individuals**, seleccionamos de la parte izquierda en **Class Hierarchy**, la clase



que deseamos que tenga las instancias y damos clic en el botón donde indica la flecha:

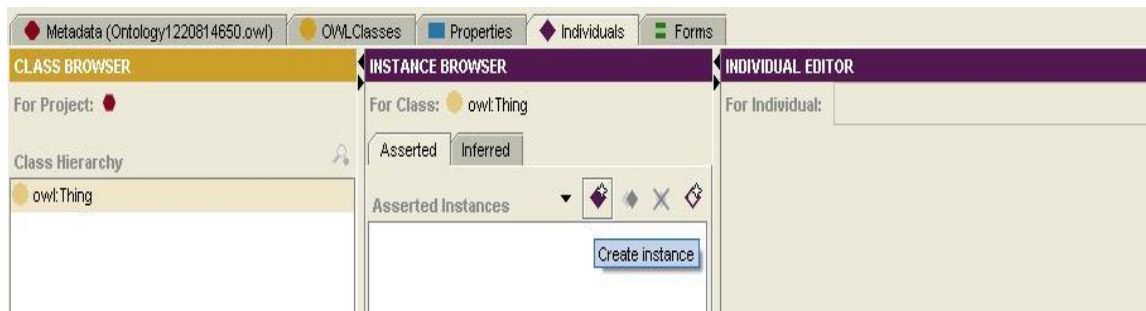


Figura 15

Para colocar el nombre a la instancia, nos situamos al lado derecho en **Individual Editor, For Individual** y ponemos el nombre que deseamos.



Existe la posibilidad de crear otras instancias a partir de una instancia creada anteriormente, además de poder eliminarlas. Para esto existen los dos botones resaltados en el cuadro de la figura, uno para copiar y otro para eliminar instancias respectivamente:

Figura 16

Cabe destacar que una instancia además de heredar los atributos dados a la clase que pertenece, ésta también posee un atributo llamada *RDFS:COMMENT*, que no es más que una posibilidad de ampliar la información que aportará o contendrá la instancia.

En la siguiente figura, se puede ver como la subclase **Bases de datos Area Administrativa** de la clase **Bases de Datos**, contiene los atributos dados a la clase principal **Servicio Base de Datos**, el atributo *rdfs:comment* (en el cuadro) y sus respectivas instancias.

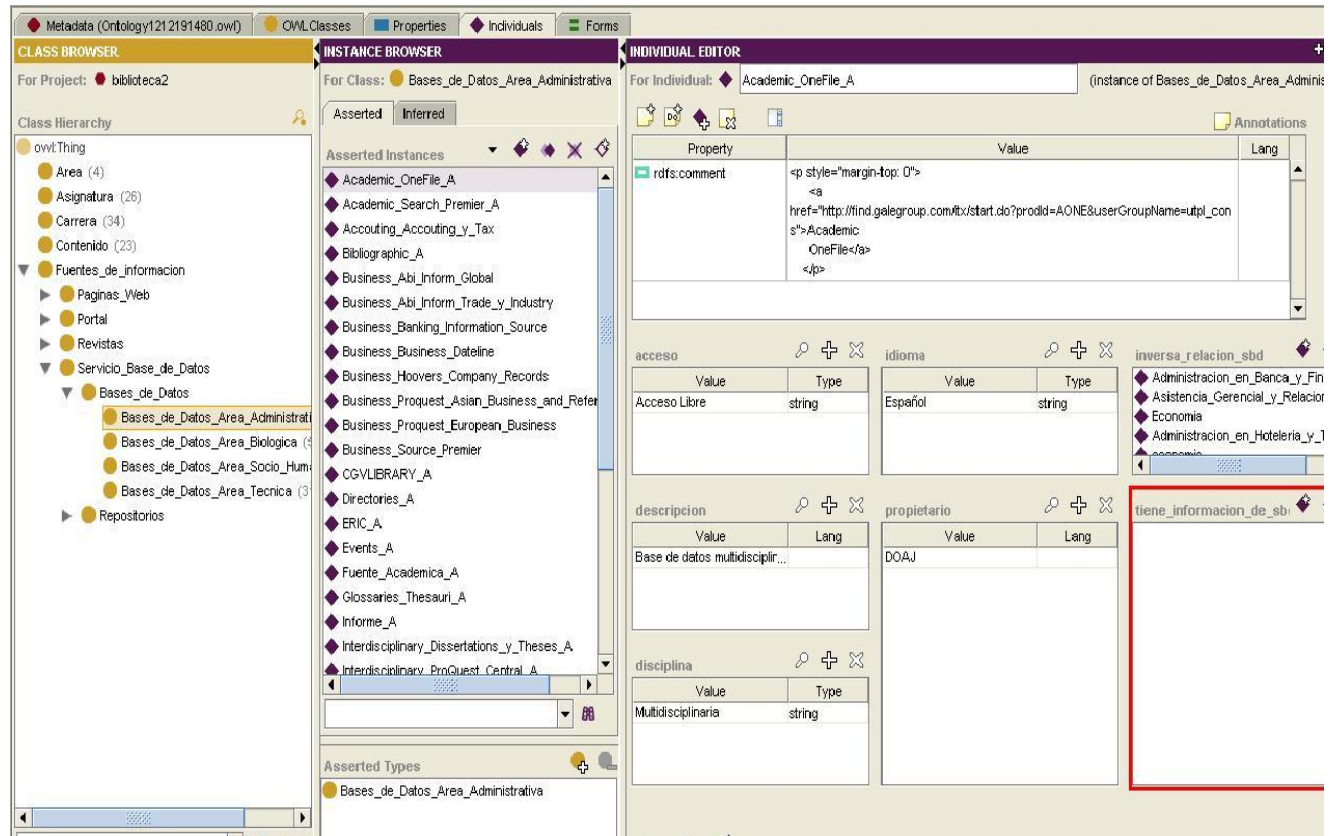
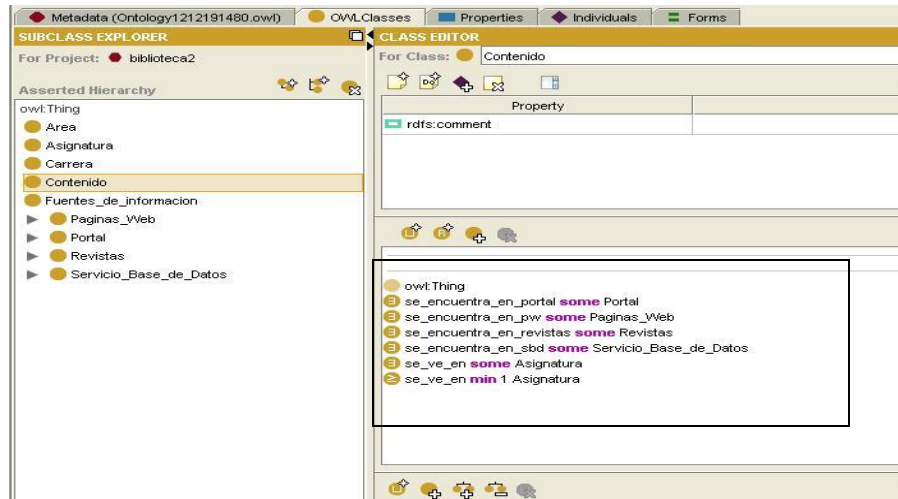


Figura 17

### 2.1.5. Creación de Restricciones

Se utilizan para que una instancia cumpla ciertas condiciones que han sido especificadas en la clase a la que pertenece.





En el siguiente grafico, en el lado derecho (ver recuadro), se observa a la clase **Contenidos** con restricciones

Figura 18

Existen dos formas de restricciones:

### Restricciones con cuantificadores

Cuantificador Existencial ( $\exists$ ) significa “Al menos un(a)”, “Algunos(as)”

En la ontología tenemos a la clase **Area** con la siguiente restricción:

La clase Area tiene “**algunas**” Carrera



Figura 19

Cuantificador Universal ( $\forall$ ) significa “Solo”

En la ontología, no está implementado este cuantificador, pero para poder mostrar un ejemplo, se ha agregado la siguiente restricción:

La clase Area tiene “solo” Carrera, por lo tanto al momento de crear una instancia

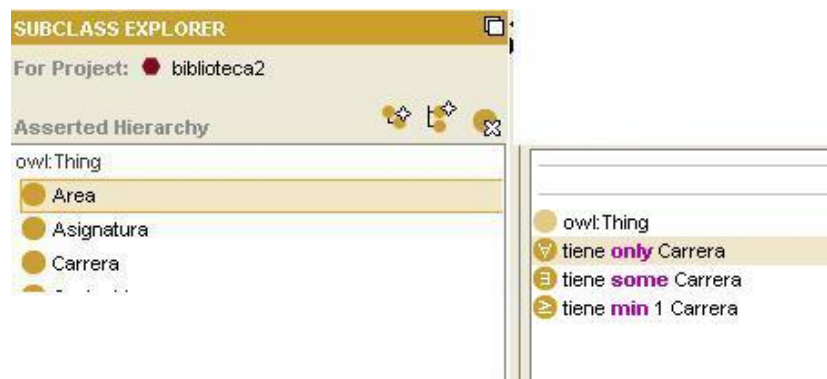


Figura 20



### **Restricciones de cardinalidad**

Exactamente (=) significa **“Exactamente”, “Igual”**. Permiten definir *exactamente* la cantidad de elementos que tendrán las propiedades Object (relaciones) o propiedades Datatype (atributos).

Cardinalidad Máxima ( $\geq$ ) significa **“Máximo (a)”**. Son las que permiten definir *el máximo* número de elementos que tendrán las propiedades Object (relaciones) o propiedades Datatype (atributos).

Cardinalidad Mínima ( $\leq$ ) significa **“Mínimo (a)”**. Definen *el mínimo* número de elementos que tendrán las propiedades Object (relaciones) o propiedades Datatype (atributos).



En la ontología, tenemos la siguiente restricción:

La clase Carrera imparte como **mínimo 1** Asignatura

Figura 21

### 3. Razonador RDF API for PHP v.096

#### 3.1. Herramienta Wamp Server v2.0a

Es un mini-servidor que instala conjuntamente al lenguaje de programación PHP, a las bases de datos MySQL y al servidor Apache.

Para el desarrollo de este proyecto se utilizaron las siguientes versiones:

PHP v 5.2.5

MySQL v 5.0.45

Apache v 2.2.6

**Instalación.-** su instalación es muy sencilla, colocamos la siguiente dirección en el explorador de internet <http://www.wampserver.com> y



descargamos la última versión (pero como ya se menciona, para el presente proyecto se utilizó la versión Wamp Server v2.0a)

A continuación se debe seguir los siguientes pasos:

Name	Date modified	Type	Size	Tags
Paginas php	14/07/2008 23:53	File Folder		
apache_java_tomcat_...	06/03/2008 11:12	Adobe Acrobat D...	77 KB	
Creando un servidor c...	14/03/2008 12:55	MHTML Document	143 KB	
Tutorial Crea tu propio...	14/03/2008 12:49	MHTML Document	1.008 KB	
WampServer2.0a	13/03/2008 18:01	Application	19.919 KB	

1. Una vez descargado el programa, damos clic sobre el archivo .exe

Figura 22

2. Luego aparecerá una pantalla indicando que no debemos instalar la reciente versión sobre una anterior, sino que debemos guardar todos nuestros datos e desinstalar la versión más antigua, para poder instalar la nueva. Después se da clic en Yes y luego en Next.

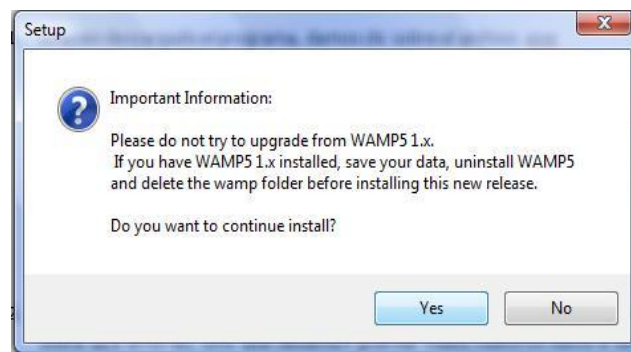




Figura 23

3. Aceptamos las condiciones del contrato y damos clic en Next. Aparecerá una pantalla que indica el lugar donde se va instalar el programa, o se puede seleccionar el lugar donde se desee que se instale el programa, haciendo clic en Browse. Luego un clic en Next y otra vez en Next.



Figura 24

4. Finalmente aparece una pantalla que indica un breve resumen de las características del programa que van a ser instaladas. Luego dar clic en Install.

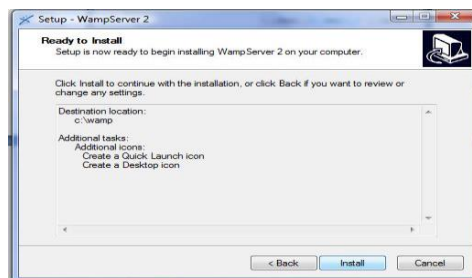
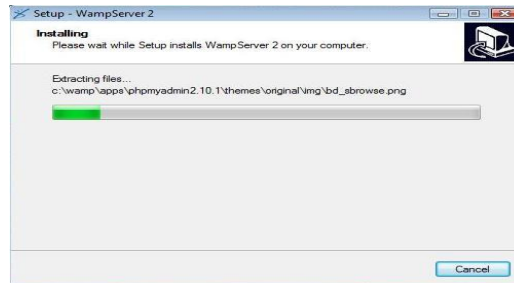


Figura 25



5. Aparecerá una ventana, donde se muestra el progreso de la instalación.

Figura 26

6. Antes de finalizar la instalación, se muestra una pantalla, que indica que debemos ingresar el servidor SMTP (por defecto es localhost y se lo deja tal cual aparece) y un correo electrónico.

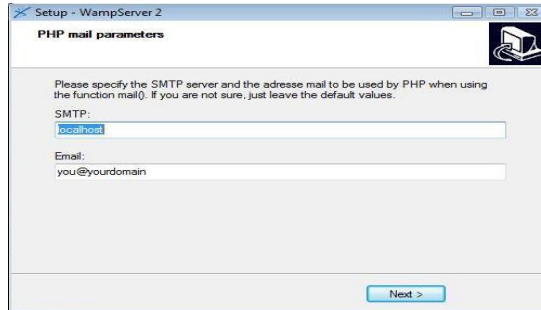


Figura 27

7. Listo, se ha instalado el servidor Wamp Server. Para poder acceder al servidor rápidamente, se deja la casilla **Launch WampServer 2 Now**, activada. Luego dar clic en Finish



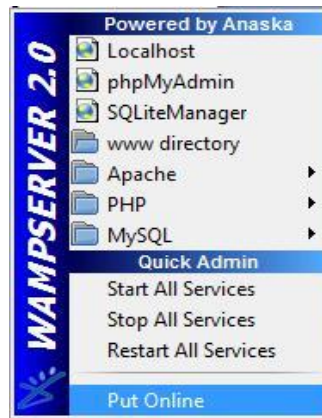
Figura 28

8. En el escritorio se habrá colocado un acceso directo al servidor Wamp Serve, se da clic en ese acceso, y poner atención a la barra de tareas de Windows, ya que en ésta aparecerá un icono similar al que indica la flecha, que quiere decir que el servidor está activo y que se puede comenzar a trabajar.



Figura 30

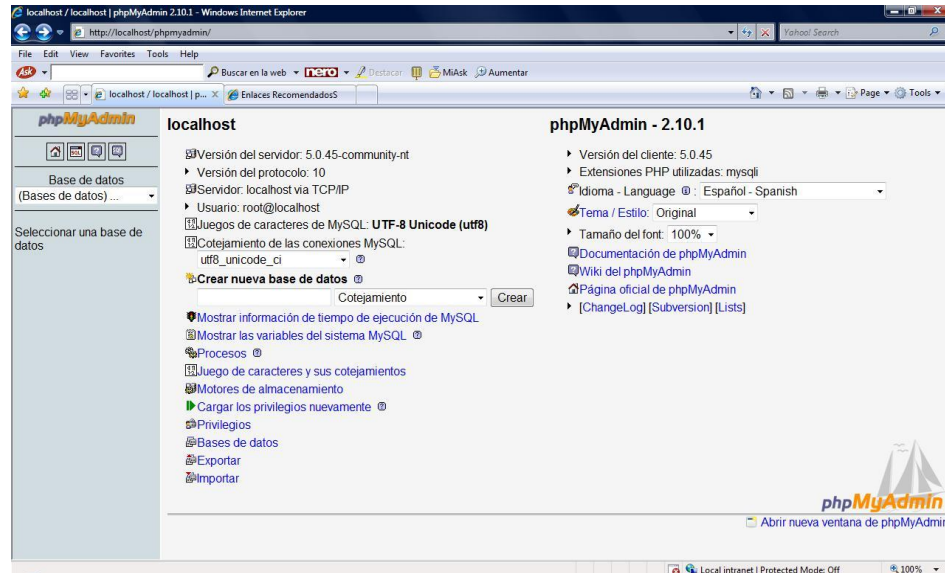




En caso de que no se active automáticamente (el icono tiene un símbolo de color negro en medio), se da un clic con el botón izquierdo del mouse y clic en **Put Online**.

Figura 31

9. Finalmente, para ingresar al servidor Wamp Server, colocamos en el explorador de internet la siguiente dirección:



<http://localhost/phpmyadmin/>, y aparecerá una pantalla similar a la siguiente:

Figura 32

Lo que indica que el servidor ha sido instalado correctamente.

10. Para cargar en el explorador de internet las páginas web que se han creado para el presente proyecto, nos dirigimos al siguiente directorio: C:\wamp, donde se encuentra la carpeta WWW, que es la que contendrá y será el lugar donde se pondrán todos los archivos php que sean utilizados.

### 3.2. Herramienta Zend Studio

También llamado **Zend Development Environment**, es un entorno o plataforma de desarrollo para el lenguaje de programación PHP, que no es gratuita.



Características: según Wikipedia<sup>61</sup>, Zend Studio tiene las siguientes características:

- Soporte para PHP 4 y PHP 5.
- Resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase.
- Inserción automática de paréntesis y corchetes de cierre.
- Sangrado automático y otras ayudas de formato de código.
- Si se sitúa el cursor sobre un paréntesis (corchete) de apertura (cierre), Zend Studio localiza el correspondiente paréntesis (corchete) de cierre (apertura)).
- Detección de errores de sintaxis en tiempo real.
  
- Instalación de barras de herramientas para Internet Explorer y Mozilla Firefox (opcional).
- Soporte para gestión de grandes proyectos de desarrollo.
- Manual de PHP integrado.
- Cliente FTP integrado.
- Soporte para navegación en bases de datos y ejecución de consultas SQL.

Zend Studio fue diseñado para el lenguaje PHP, sin embargo también ofrece soporte para otros lenguajes Web como HTML, Javascript y XML.

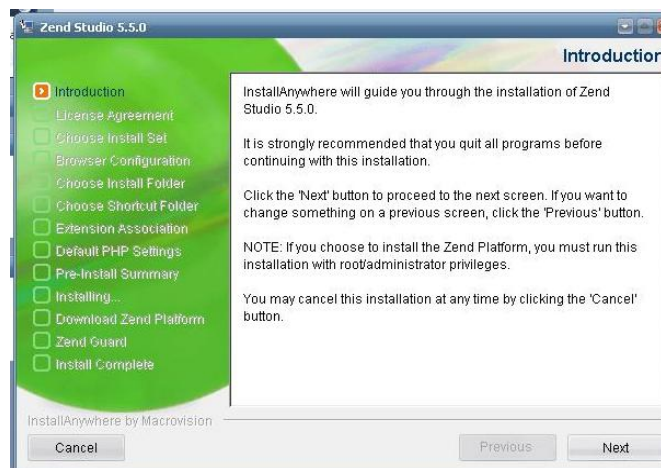
**Instalación.-** para este proyecto se ha utilizado la versión 5.5.0

---

<sup>61</sup> Wikipedia Zend Studio, recuperado el 11 de septiembre del 2008 de [http://es.wikipedia.org/wiki/Zend\\_Studio](http://es.wikipedia.org/wiki/Zend_Studio)



1. Descargar el paquete de Zend Studio de la siguiente dirección <http://www.zend.com/en/downloads/>



2. Dar clic sobre el instalador y saldrá la siguiente pantalla, que muestra una descripción del programa, luego hacer clic en Next.

Figura 33

3. Después aparece una pantalla donde aceptamos el contrato y clic en Next. Posteriormente se muestra una pantalla donde se escoge las características a instalar y dar clic en Next.

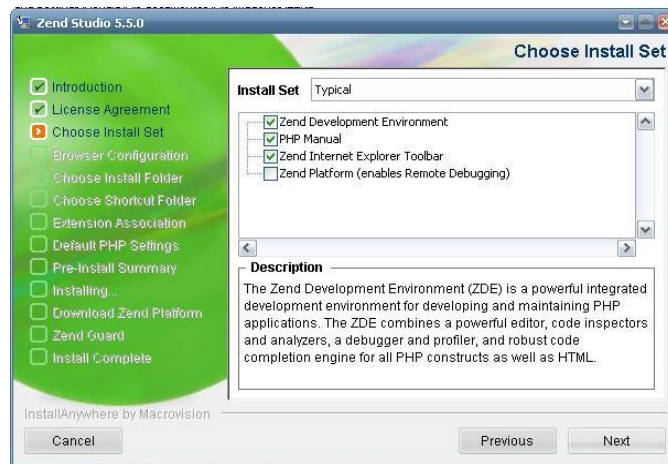
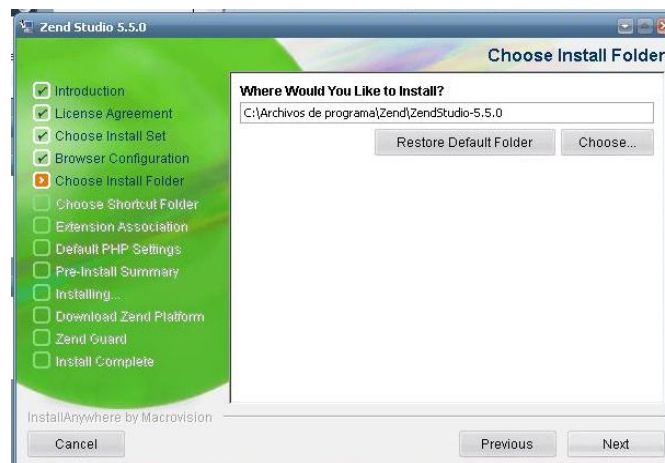


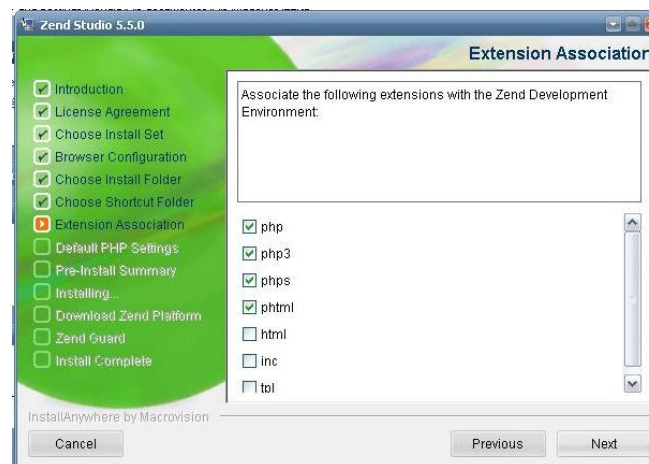
Figura 34



4. Luego se muestra una pantalla donde se escoge el lugar donde se instalará el paquete completo de Zend Studio, por defecto seleccionamos el lugar que sugiere el instalador, después clic en Next.



Figura 35



5. En la siguiente pantalla dar clic en Next, luego aparecerá una pantalla similar a ésta, donde escogemos los tipos de extensiones de archivos que asociaremos para que Zend Studio los abra por defecto, luego clic en Next.

Figura 36

6. Luego escogemos la versión del lenguaje de programación PHP (que será 5.x) y clic en Next. Después aparece una pantalla con un pequeño resumen de las características que hemos seleccionado anteriormente y damos clic en Install.

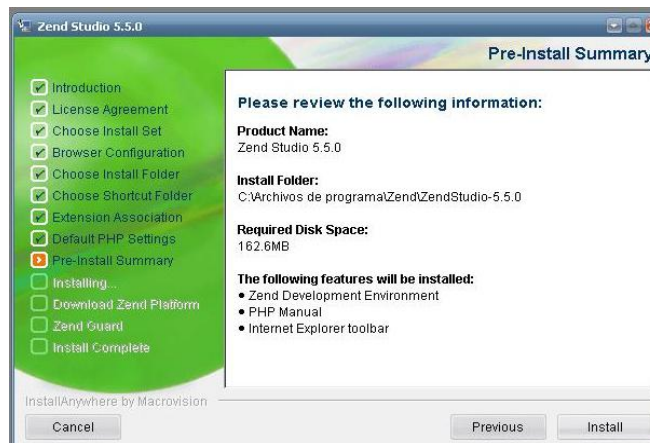


Figura 37

7. Por último aparecerá una pantalla que nos indica que sea instalado correctamente el programa, luego dar clic en Done. Si deseamos en ese momento podemos acceder al entorno dejando activada la casilla **Launch Zend**



**Development Environment.**

Figura 38

### 3.3. Razonador RDF API

#### 3.3.1. Conversión de OWL a RDF



Para utilizar este razonador de Jena exclusivamente para PHP, primero se debe tener la ontología en formato de archivo RDF. Si no lo está aún, se debe seguir el siguiente procedimiento:

1. Una vez que se ha terminado de modelar la ontología en Protégé, se debe convertir el archivo OWL a archivo RDF.

Para esto hacemos clic en **File >> Convert Project to Format.**

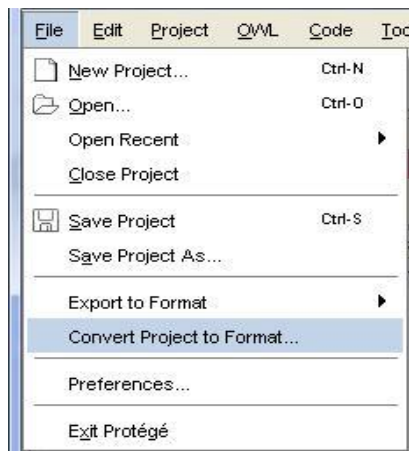


Figura 39

2. Luego se escoge en **Select Format a RDF files** y después clic en OK.



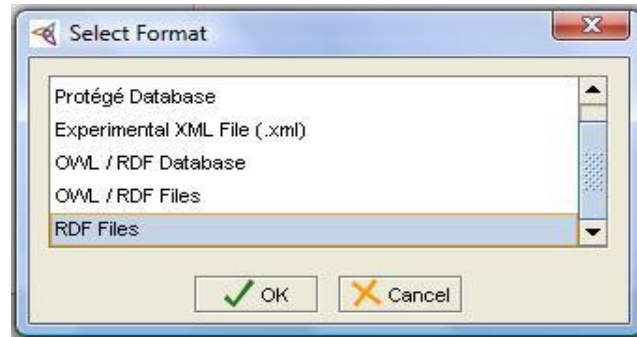


Figura 40

3. Después aparece una pantalla donde se debe escoger el nombre del proyecto que va a ser convertido al formato RDF (Project), el nombre que va llevar el archivo RDF (Classes File Name) y el nombre del archivo RDF con la extensión RDFS (Instances File Name). El campo Namespace se lo puede dejar en blanco. Luego



hacer clic en el botón OK.

Figura 41



Como se puede observar, en el campo **Classes file name**, se coloca el nombre del archivo OWL, ya no con la extensión OWL, sino con el nuevo tipo de archivo, que es RDF. Por ejemplo si la ontología se llama pepito.**OWL**, ahora se llamará pepito.**RDF**

El tercer campo **Instances file name** contendrá el mismo nombre de la ontología en el formato OWL, pero ahora con el tipo de formato RDFS.

Esto se hace, ya que el razonador utiliza dos tipos de archivos al momento de inferir conocimiento de la ontología, que son RDF y RDFS.

4. Se espera a que termine de completar la transformación de la ontología al formato RDF, para luego ir a la carpeta donde se almacena los archivos RDF y RDFS, en este caso nos dirigimos a C:\wamp\www\ontologias, que es el lugar donde están las distintas ontologías con las distintas extensiones (como OWL, RDF, RDFS, PPRJ).



Name	Date modified	Type	Size	Tags
ejemplo_ontologia.rdf	05/05/2008 16:47	RDF File	2 KB	
Onto_biblioteca_bv.rdf	08/07/2008 17:53	RDF File	26 KB	
ontologia_biblioteca.rdf	05/05/2008 17:42	RDF File	37 KB	
Ontologia_BV.rdf	19/08/2008 11:03	RDF File	5 KB	
ontologia_utpl.rdf	03/08/2008 23:24	RDF File	33 KB	
ontoprueba.rdf	28/07/2008 17:19	RDF File	6 KB	
prueba.rdf	24/07/2008 11:09	RDF File	431 KB	
biblioteca.rdfs	21/08/2008 9:28	RDFS File	9 KB	
biblioteca2.rdfs	06/09/2008 12:22	RDFS File	21 KB	
copia_biblioteca.rdfs	24/07/2008 16:36	RDFS File	8 KB	
ejemplo_ontologia.rdfs	05/05/2008 16:47	RDFS File	1 KB	
Onto_biblioteca_bv.rdfs	08/07/2008 17:53	RDFS File	7 KB	
ontologia_biblioteca.r...	05/05/2008 17:42	RDFS File	4 KB	
Ontologia_BV.rdfs	19/08/2008 11:03	RDFS File	5 KB	
ontologia_utpl.rdfs	03/08/2008 23:24	RDFS File	6 KB	
ontoprueba.rdfs	28/07/2008 17:19	RDFS File	3 KB	
prueba.rdfs	24/07/2008 11:09	RDFS File	17 KB	
biblioteca.repository	26/08/2008 19:17	REPOSITORY File	1 KB	
Onto_biblioteca_bv.re...	08/07/2008 17:53	REPOSITORY File	1 KB	
ontologia_biblioteca.re...	05/05/2008 16:18	REPOSITORY File	1 KB	

Figura 42

5. Una vez completados los 3 primeros pasos, ya se ha convertido una ontología OWL a un archivo RDF, que puede ser interpretado por el Razonador RDF API.

### 3.3.2. Configuración del razonador RDF API

Para que pueda funcionar el razonador conjuntamente en el lenguaje de programación PHP, se debe seguir los siguientes pasos:

1. Descargar la última versión del paquete. Se lo puede hacer desde la siguiente página: <http://sourceforge.net/projects/rdfapi-php/>
2. Descomprimir el archivo .zip en la carpeta WWW del servidor Wamp Server, es decir en C:\wamp\www



Name	Artists	Type
fondos		File Folder
mediawiki-1.12.0		File Folder
ontologias		File Folder
rdfapi-php		File Folder
Onto_biblioteca_bv		XML Document
codigo consultas_3 al ...		Text Document
New Text Document		Text Document
biblioteca rdfc		RDF File

Figura 43

Como se puede observar (donde indica la flecha), dentro de la carpeta WWW se encuentra ya descomprimido el razonador RDF API

3. Para poder trabajar con los archivos RDF y RDFS, estos deben ser colocados en la carpeta API del razonador RDF API, es decir en C:\wamp\www\rdfapi-php\api. Para un mejor entendimiento se muestra el siguiente grafico:

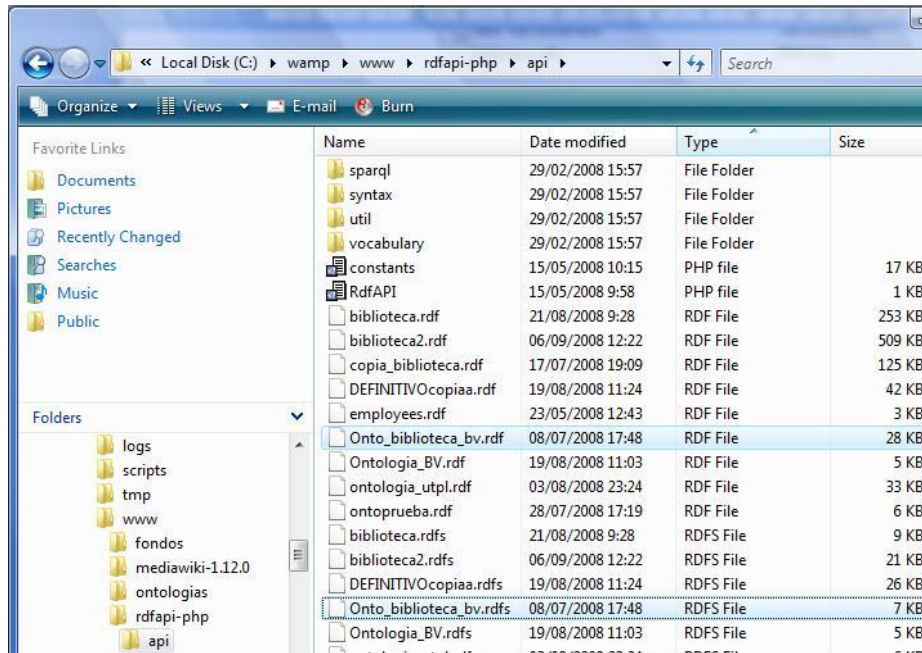


Figura 44

Incluir el RDF API en el script, de esta forma:

```
define("RDFAPI_INCLUDE_DIR",  
"C:/Apache/htdocs/rdf_api/api");  
include(RDFAPI_INCLUDE_DIR . "RDFAPI.php");
```

Como previamente se debe tener instalado y configurado adecuadamente el Zend Studio (que es el que permite la interoperabilidad entre estas 2 herramientas) se procede a cargar en un archivo .php el razonador. Por ejemplo en la siguiente figura se puede ver su implementación:



```
<?php  
  
define ("RDFAPI_INCLUDE_DIR", "C:/wamp/www/rdfapi-php/api/");  
include (RDFAPI_INCLUDE_DIR . "RdfAPI.php");  
  
$employees = ModelFactory::getDefaultModel();  
$employees->load(RDFAPI_INCLUDE_DIR ."ontologia.rdf");
```

Figura 45

Donde, en la primera variable \$employees se obtiene un modelo de la ontología y en la segunda variable \$employees se carga el archivo RDF



## ANEXO II: CÓDIGO RECOMENDADOR.php

```
<html>
<head>
  <title>Recomendaciones Academicas</title>
  <!--<LINK
href="/media/hdd2/www/eva19/course/Recommendations/estilos_otrosb.css"
rel="stylesheet" type="text/css">-->
<body background-image="blue">

<style type="text/css">
<!--
a:link { font-family: Tahoma, Verdana, Arial; font-size=11; color: #00008B; text-
decoration: none}
a:visited { font-family: Tahoma, Verdana, Arial; font-size=11; color: #8B0000; text-
decoration: none}
a:hover { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996600}
a:active { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996666; text-
decoration: none}

a.estilos:link { font-family: Tahoma, Verdana, Arial; font-size=11; color: #0000FF;
text-decoration: none}
a.estilos:visited { font-family: Tahoma, Verdana, Arial; font-size=11; color:
#996600; text-decoration: none}
a.estilos:hover { font-family: Tahoma, Verdana, Arial; font-size=11; color: #8B6914}
a.estilos:active { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996666;
text-decoration: none}

body {background-color: #E6E6FA ; background-repeat: x ; background-position:
left bottom}
.centrado {text-align:center}
-->
</style>
<style>
  .thdrccell {
    background:#F3F0E7;
    font-family:arial;
    font-size:12px;
    font-weight:bold;
    padding:5px;
    border-bottom:1px solid #C8BA92;
  }

  .tdatccell {
    font-family:arial;
    font-size:12px;
    padding:5px;
    background:#FFFFFF
  }
}
```



```
.dvhdr1 {
  background:#F3F0E7;
  font-family:arial;
  font-size:12px;
  font-weight:bold;
  border:1px solid #C8BA92;
  padding:5px;
  width:150px;
}

.dvbdy1 {
  background:#FFFFFF;
  font-family:arial;
  font-size:12px;
  border-left:1px solid #C8BA92;
  border-right:1px solid #C8BA92;
  border-bottom:1px solid #C8BA92;
  padding:5px;
  width:150px;
}

p {
  margin-top:20px;
}

h1 {
  font-size:13px;
}

.dogvdvhdr {
  width:300;
  background:#C4D5E3;
  border:1px solid #C4D5E3;
  font-weight:bold;
  padding:10px;
}

.dogdvbdy {
  width:300;
  background:#FFFFFF;
  border-left:1px solid #C4D5E3;
  border-right:1px solid #C4D5E3;
  border-bottom:1px solid #C4D5E3;
  padding:10px;
}

.pgdiv {
  width:320;
  height:250;
  background:#E9EFF4;
```





```
border:1px solid #C4D5E3;
padding:10px;
margin-bottom:20;
font-family:arial;
font-size:12px;
}
</style>
</head>

<body style="font-family:arial;font-size:12px">
<?php
?>

<?php
if (file_exists("/media/hdd2/www/eva19/course/Recomendations/rdfapi-
php/api/RdfAPI.php"))
{
    define ("RDFAPI_INCLUDE_DIR",
"/media/hdd2/www/eva19/course/Recomendations/rdfapi-php/api/");
    include (RDFAPI_INCLUDE_DIR . "RdfAPI.php");
    $employees = ModelFactory::getDefaultModel();
    $employees->load(RDFAPI_INCLUDE_DIR . "biblioteca2_002.rdf");

//SABER EL NOMBRE DE LA MATERIA Y LA CARRERA A LA QUE PERTENECE
}
else {
}

$idpas=$_GET["idpas"];
$id_user=$_GET["id_user"];
$sql = "SELECT firstname FROM mdl_user WHERE id=$id_user";
$result = mysql_query($sql);
while ($row = mysql_fetch_array($result))
{
$row["lastname"], "</font>","<br>";
echo $row["firstname"],"<br>";
}
//SABER EL NOMBRE DE LA MATERIA Y LA CARRERA A LA QUE PERTENECE
////Materia
$query ='
SELECT ?Materias
WHERE (?Carreras, <kb:imparte>, ?Materias)
AND ?Materias ~~ "/'. $idpas. '/'
USING kb FOR <http://protege.stanford.edu/kb#>';
$rdqllter = $employees->rdqlQueryasIterator($query);
$result_labels=$rdqllter->getResultLabels();
//echo "Materias".$result_labels->Materias;
```



```
while ($rdqIter->hasNext())
{
    $current_result=$rdqIter->next();

    for ($j=0; $j<count($result_labels); $j++)
    {
        $cadena[$j]=$current_result[$result_labels[$j]];
        $nom_materia3[$j]=$current_result[$result_labels[$j]];
        $extraer[$j]=substr("$cadena[$j]",49);
        $extraer2[$j]=strrev("$extraer[$j]");
        $extraer3[$j]=substr("$extraer2[$j]",2);
        $extraer4[$j]=strrev("$extraer3[$j]");
        //$nom_materia3=$extraer4[$j];
    }
};

foreach ($nom_materia3 as $nom_materia1) {
    //echo $cadena=substr("$nom_materia[$j]",49);
    foreach ($nom_materia1 as $nom_materia2)
    {
        //print_r("BBB".$nom_materia2."<br>");
        $nom_materia4=$nom_materia2;
    }
}
//echo "VAR materia=".$nom_materia=substr($nom_materia4,40),"<br>";
$nom_materia=substr($nom_materia4,40);
$query1 ='
SELECT ?Carreras
WHERE (?Carreras, <kb:imparte>, ?Materias)
AND ?Materias ~~ "/.sidpas.'"
USING kb FOR <http://protege.stanford.edu/kb#>';
$rdqIter = $employees->rdqQueryasIterator($query1);
$result_labels=$rdqIter->getResultLabels();
//echo "Carr=".$result_labels->Carreras;

while ($rdqIter->hasNext())
{
    $current_result=$rdqIter->next();
    for ($j=0; $j<count($result_labels); $j++)
    {
        $cadena[$j]=$current_result[$result_labels[$j]];
        $nom_carrera3[$j]=$current_result[$result_labels[$j]];
        $extraer[$j]=substr("$cadena[$j]",41);
        $extraer2[$j]=strrev("$extraer[$j]");
        $extraer3[$j]=substr("$extraer2[$j]",2);
        $extraer4[$j]=strrev("$extraer3[$j]");
        //$extraer5[$j]=str_replace("_", " ",$extraer4[$j]);
        //$nom_carrera=$extraer4[$j];
    }
}
```



```
}
};
//Fin Carrera

foreach ($nom_carrera3 as $nom_carrera1)
{
    foreach ($nom_carrera1 as $nom_carrera2)
    {
        $nom_carrera4=$nom_carrera2;
    }
}
$nom_carrera=substr($nom_carrera4,31);
/////FIN SABER EL NOMBRE DE LA MATERIA Y LA CARRERA A LA QUE
PERTENECE
//
$presen_materia=str_replace("_","",$nom_materia);
$presen_carrera=str_replace("_","",$nom_carrera);

echo "<font size=2 face=Eurostile>";
echo "Las sugerencias que a continuacion te presentamos se las realiza en base a
la asignatura ",strtoupper($presen_materia), " que cursa. Ademas se presentan
otras sugerencias como bases de datos, bibliotecas virtuales, revistas,... de
acuerdo a su carrera ", strtoupper($presen_carrera)," en la que esta matriculado.";
//echo "Las sugerencias que a continuacion te presentamos se las realiza en base
a la asignatura ",strtoupper($nom_materia), " que cursa. Ademas se presentan
otras sugerencias como bases de datos, bibliotecas virtuales, revistas,... de
acuerdo a su carrera ", strtoupper($presen_carrera)," en la que esta matriculado.";
echo "</font>","<hr>";

?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3c.org/TR/1999/REC-html401-19991224/loose.dtd">
<!--<HTML><HEAD><TITLE>Tab Pane (WebFX)</TITLE>-->
<META http-equiv=Content-Type content="text/html; charset=utf-8">
<SCRIPT src="Tab_Pane_(WebFX)_files/webfxlayout.js"
type=text/javascript></SCRIPT>

<SCRIPT src="Tab_Pane_(WebFX)_files/webfxapi.js"
type=text/javascript></SCRIPT>

<SCRIPT src="Tab_Pane_(WebFX)_files/tabpane.js"
type=text/javascript></SCRIPT>
<LINK href="Tab_Pane_(WebFX)_files/tab.webfx.css" type=text/css
rel=StyleSheet>
<STYLE type=text/css>.dynamic-tab-pane-control .tab-page {
    OVERFLOW: auto; WIDTH: 475px; HEIGHT: 300px
}
.dynamic-tab-pane-control .tab-page .dynamic-tab-pane-control .tab-page {
    HEIGHT: 100px
}
}
```



```
FORM {
    PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM: 0px;
    MARGIN: 0px; PADDING-TOP: 0px
}
.tab-pane.dynamic-tab-pane-control {
    MARGIN: 10px; WIDTH: 580px
}
.dynamic-tab-pane-control H2 {
    WIDTH: auto; TEXT-ALIGN: center
}
.dynamic-tab-pane-control H2 A {
    DISPLAY: inline; WIDTH: auto
}
.dynamic-tab-pane-control A:hover {
    BACKGROUND: none transparent scroll repeat 0% 0%
}
.dynamic-tab-pane-control .tab-page H2 {
    TEXT-ALIGN: left
}
</STYLE>

<SCRIPT type=text/javascript>
//

var tabPane;

function showArticleTab( sName ) {
    if (typeof tabPane != "undefined" ) {

        switch ( sName ) {

            case "main":
                tabPane.setSelectedIndex( 0 );
                break;

            case "usage":
                tabPane.setSelectedIndex( 1 );
                break;

            case "implementation":
                tabPane.setSelectedIndex( 3 );
                break;

            case "looknfeel":
                tabPane.setSelectedIndex( 4 );
                break;

        }
    }
}</pre></div><div data-bbox="827 918 867 936" data-label="Page-Footer"><p>180</p></div>
```



```
// help tips
htDom = "Document Object Model 1 is a standard developed by the W3C.<br />" +
"<a href='\"http://www.w3.org/DOM/\"' target='\"_blank\"'>http://www.w3.org/DOM/</a>";

//]]>
</SCRIPT>
```

```
<META content="MSHTML 6.00.6001.18148" name=GENERATOR></HEAD>
<BODY><!-- WebFX Layout Include -->
<SCRIPT type=text/javascript>
<!--var articleMenu= new WebFXMenu;-->
<!--articleMenu.left = 384;
articleMenu.top = 86;
articleMenu.width = 140;
articleMenu.add(new WebFXMenuItem("Tab Pane", "javascript:showArticleTab(
\"main\" )"));
articleMenu.add(new WebFXMenuItem("Usage", "javascript:showArticleTab(
\"usage\" )"));
articleMenu.add(new WebFXMenuItem("API", "javascript:showArticleTab( \"api\"
)"));
articleMenu.add(new WebFXMenuItem("Implementation",
"javascript:showArticleTab( \"implementation\" )"));
articleMenu.add(new WebFXMenuItem("Look & Feel",
"javascript:showArticleTab( \"looknfeel\" )"));
articleMenu.add(new WebFXMenuItem("Demo", "demo.html"));
articleMenu.add(new WebFXMenuSeparator);
articleMenu.add(new WebFXMenuItem("Download",
"http://webfx.eae.net/download/tabpane102.zip"));
webfxMenuBar.add(new WebFXMenuButton("Article Menu", null, null,
articleMenu));-->
```

```
<!--webfxLayout.writeTitle("Tab Pane");
webfxLayout.writeMenu();
webfxLayout.writeDesignedByEdger();-->
</SCRIPT>
```

```
<DIV class=webfx-main-body><!-- end WebFX Layout Includes -->
```

```
<!-- begin tab pane -->
<DIV class=tab-pane id=article-tab>
<SCRIPT type=text/javascript>
tabPane = new WebFXTabPage( document.getElementById( "article-tab" ), true );
</SCRIPT>
```

```
<!-- begin intro page -->
<DIV class=tab-page id=intro-page>
<H2 class=tab>Por Contenidos</H2>
<SCRIPT type=text/javascript>
tabPane.addTabPage( document.getElementById( "intro-page" ) );
```



```
</SCRIPT>

<?php
//CONTENIDOS
//echo "nombre materia=".$nom_materia;
echo "<div id='menu_izq'>";
$query2 = '
SELECT ?Nombre
WHERE (?Contenido, <kb:se_ve_en>, ?Asignatura)
      (?Asignatura, <kb:se_estudia_en>, ?Carrera)
      (?PW, <kb:tiene_informacion_de_pw>, ?Contenido)
      (?PW, <rdfs:comment>, ?Nombre)
AND ?Contenido ~~ "/.$nom_materia.'" || ?Asignatura ~~ "/.$nom_materia.'"
USING kb FOR <http://protege.stanford.edu/kb#> rdfs FOR
<http://www.w3.org/2000/01/rdf-schema#>;

$rdqIter = $employees->rdqQueryasIterator($query2);
$result_labels=$rdqIter->getResultLabels();

echo "<table width='95%' height='10%'>";
while ($rdqIter->hasNext())
{
    $current_result=$rdqIter->next();
    for ($j=0; $j <count($result_labels); $j++)
    {
        //echo $cadena[$j]=$current_result[$result_labels[$j]];
        echo $cadena[$j]=$current_result[$result_labels[$j]];
        //echo $contenidos3[$j]=substr($current_result[$result_labels[$j]],30);
        //substr($nom_carrera4,31)
        //echo "QQ";
        // $extraer[$j]=substr("$cadena[$j]",9);
        // $extraer2[$j]=strrev("$extraer[$j]");
        // $extraer3[$j]=substr("$extraer2[$j]",2);
        //echo "<td>",$extraer4[$j]=strrev("$extraer3[$j]"),"</td>";
        //echo "<tr>";
    }
};

echo "</p>";
echo "</table>";
echo "</div>";
//FIN CONTENIDOS
?>
</DIV><!-- end intro page -->

<!-- begin usage page -->
<DIV class=tab-page id=usage-page>
<H2 class=tab>Servicios de Bases de Datos</H2>
<SCRIPT type=text/javascript>
```



```
tabPane.addTabPage( document.getElementById( "usage-page" ) );
</SCRIPT>

<?php
$query3 =
SELECT ?accesos, ?idioma, ?propietario, ?Enlace
WHERE (?SBD, <kb:tiene_informacion_de_sbd>, ?Carrera)
      (?SBD, <rdfs:comment>, ?Enlace)
      (?SBD, <kb:acceso>, ?accesos)
      (?SBD, <kb:idioma>, ?idioma)
      (?SBD, <kb:propietario>, ?propietario)
      (?SBD, <kb:disciplina>, "Especifica")
AND ?Carrera ~~ "/'.$nom_carrera.'"
USING kb FOR <http://protege.stanford.edu/kb#> rdfs FOR
<http://www.w3.org/2000/01/rdf-schema#>;

$rdqIter = $employees->rdqQueryasIterator($query3);
$result_labels=$rdqIter->getResultLabels();
echo "<p class='bdg'>","<b>","<font size=2 color=green face=Eurostile>","BASES DE
DATOS","</font>","</b>";
echo "<table width='95%'>";
while ($rdqIter->hasNext())
{
    $curren_result=$rdqIter->next();
    for ($j=0; $j <count($result_labels); $j++)
    {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
    }
};
echo "</table>";
echo "</p>";
?>
</DIV><!-- end usage tab -->

<!-- begin implementation tab -->
<DIV class=tab-page id=implementation-page>
<H2 class=tab>Revistas</H2>
<SCRIPT type=text/javascript>
tabPane.addTabPage( document.getElementById( "implementation-page" ) );
</SCRIPT>
<?php
$query4 =
SELECT ?Materias
WHERE (?Enlaces, <kb:tiene_informacion_de_revista>, ?'.$nom_carrera.')
      (?Enlaces, <rdfs:comment>, ?Materias)
AND ?'.$nom_carrera.' ~~ "/'.$nom_carrera.'"
USING kb FOR <http://protege.stanford.edu/kb#> rdfs FOR
<http://www.w3.org/2000/01/rdf-schema#>;

$rdqIter = $employees->rdqQueryasIterator($query4);
```



```
$result_labels=$rdqIter->getResultLabels();
echo      "<p      class='bdg'>","<b>","<font      size=2      color=green
face=Eurostile>","REVISTAS","</font>","</b>";
echo "<table width='95%'>";
while ($rdqIter->hasNext())
{
    $curren_result=$rdqIter->next();
    for ($j=0; $j <count($result_labels); $j++)
    {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
    }
};
echo "</table>";
echo "</p>";
?>
</DIV><!-- end implementation tab -->

<!-- begin look and feel tab -->
<DIV class=tab-page id=look-page>
<H2 class=tab>Portales</H2>
<SCRIPT type=text/javascript>
tabPane.addTabPage( document.getElementById( "look-page" ) );
</SCRIPT>

<?php
$query5 ='
SELECT ?accesos, ?idioma, ?descripcion, ?Materias
WHERE (?Enlaces, <kb:tiene_informacion_de_portal>, ?'.$nom_carrera.')
    (?Enlaces, <rdfs:comment>, ?Materias)
    (?Enlaces, <kb:acceso>, ?accesos)
    (?Enlaces, <kb:idioma>, ?idioma)
    (?Enlaces, <kb:descripcion>, ?descripcion)
    (?Enlaces, <kb:disciplina>, "Especifica")
AND ?'.$nom_carrera.' ~~ "/'.$nom_carrera.'"
USING kb FOR <http://protege.stanford.edu/kb#> rdfs FOR
<http://www.w3.org/2000/01/rdf-schema#>;
$rdqIter = $employees->rdqQueryasIterator($query5);
$result_labels=$rdqIter->getResultLabels();
echo      "<p      class='bdg'>","<b>","<font      size=2      color=green
face=Eurostile>","PORTALES","</font>","</b>";
echo "<table width='95%'>";
while ($rdqIter->hasNext())
{
    $curren_result=$rdqIter->next();
    for ($j=0; $j <count($result_labels); $j++)
    {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
        ",$extraer4[1]=strrev("$extraer3[1]"),"</font>","<br>";
        face=Eurostile>",$extraer4[2]=strrev("$extraer3[2]"),"</font>";
    }
}
```





```
};
echo "</table>";
echo "</p>";
//FIN RECOMENDACIONES
?>
</DIV><!-- end look and feel tab -->

</DIV><!-- end tab pane -->
<?php
echo "<form action='buscador_definitivo.php' method='POST'>";
echo "<input type='text' id='term' name='nTerm' size='60' value=''>";
echo "<input type='submit' value='Buscar' />","<br />";
echo "</form>";

$nTerm=$_POST['nTerm'];
//echo " " <body
onload="\ventanaRecomendaciones('Recommendations/buscador_definitivo.php?nT
erm=".$idpas."')\"></body>";
?>

<!--<form action='buscador_definitivo.php' method='POST'>
<input type='text' id='term' name='nTerm' size='60' value=''>
<input type='submit' value='Buscar' /><br />
</form>

<body
onload="\ventanaRecomendaciones('Recommendations/buscador_definitivo.php?nT
erm=".$id."')\"></body>-->
</BODY></HTML>
<hr>
</html>
```



### ANEXO III: CÓDIGO BUSCADOR.php

```
<html>
<title>Busqueda de Recomendaciones</title>
<LINK
href="C:\wamp\www\moodle_tesis\course\Recomendations\estilos_buscadador.css
" rel="stylesheet" type="text/css">
<style type="text/css">
<!--
a:link { font-family: Tahoma, Verdana, Arial; font-size=11; color: #00008B; text-
decoration: none}
a:visited { font-family: Tahoma, Verdana, Arial; font-size=11; color: #8B0000; text-
decoration: none}
a:hover { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996600}
a:active { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996666; text-
decoration: none}

a.estilos:link { font-family: Tahoma, Verdana, Arial; font-size=11; color: #0000FF;
text-decoration: none}
a.estilos:visited { font-family: Tahoma, Verdana, Arial; font-size=11; color:
#996600; text-decoration: none}
a.estilos:hover { font-family: Tahoma, Verdana, Arial; font-size=11; color: #8B6914}
a.estilos:active { font-family: Tahoma, Verdana, Arial; font-size=11; color: #996666;
text-decoration: none}

body {background-color: #E6E6FA ; background-repeat: x ; background-position:
left bottom}

.centrado {text-align:center}

-->
</style>

</html>

<?php
{

if (file_exists("/media/hdd2/www/eva19/course/Recomendations/rdfapi-
php/api/RdfAPI.php"))
{
define ("RDFAPI_INCLUDE_DIR",
"/media/hdd2/www/eva19/course/Recomendations/rdfapi-php/api/");
include (RDFAPI_INCLUDE_DIR . "RdfAPI.php");
$employees = ModelFactory::getDefaultModel();
$employees->load(RDFAPI_INCLUDE_DIR ."biblioteca2_002.rdf");
}
else {
}
```



```
$posibilidades= Array("bases","base","informacion");
$posibilidades2= Array("bibliotecas","biblioteca","informacion");
$posibilidades3= Array("revistas","revista","informacion");
$posibilidades4= Array("paginas","pagina","informacion");
$presen=array();

$nTerm = $_POST['nTerm'];
$split = split(" ", $nTerm);
$cantidad=count($split);

echo "<b>,<font size=3 color=green face=Eurostile>,"RECOMENDACIONES",</font>,</b>,<hr>,<p>";
echo "<font size=2 color=black face=Arial>,"Su criterio de busqueda es:
",</font>,<b>,</b>,</p>";

$z=0;
// if principal
if (in_array($split[$z],$posibilidades))
{
    //if recorrer Array de posibilidades
    if ($split[$z]=="bases" || $split[$z]=="base" || $split[$z]=="informacion")
    {
        echo "<font size=4 color=green face=Eurostile>,"BASES DE DATOS.-
",</font>,<br>";

        $rdql_query='
SELECT ?Acceso, ?Descripcion, ?Nombre
WHERE (?Carrera, <kb:se_encuentra_en_sbd>, ?SBD)
(?SBD, <kb:disciplina>, "Especifica")
(?SBD, <kb:acceso>, ?Acceso)
(?SBD, <kb:descripcion>, ?Descripcion)
(?SBD, <rdfs:comment>, ?Nombre);
$rdql_query=$rdql_query.' AND ?Carrera ~ "/'.$split[0].'" || ?Carrera ~
"/'.$split[0].'"';
        if ($cantidad>1)
        {
            for ($i=1; $i<$cantidad; $i++)
            {
                if (strlen($split[$i])>4)
                {
                    $rdql_query=$rdql_query.' && ?Carrera ~ "/'.$split[$i].'" ||
?Carrera ~ "/'.$split[$i].'"';
                }
            }
            $rdql_query=$rdql_query.' , ?Carrera !~ "/Carreras/" USING kb FOR
<http://protege.stanford.edu/kb#> rdfs FOR <http://www.w3.org/2000/01/rdf-
schema#>';
        }
        $rdqlIter = $employees->rdqlQueryasIterator($rdql_query);
        $result_labels=$rdqlIter->getResultLabels();
    }
}
```



```
$cantidad_resultados=$rdqIter->countResults();

echo "<b>","Se han encontrado ",$cantidad_resultados," temas
relacionados","</b>","<br>";

while ($rdqIter->hasNext())
{
    $curren_result=$rdqIter->next();
    for ($j=0; $j<count($result_labels); $j++)
    {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
//          $extraer[$j]=substr("$cadena[$j]",9);
//          $extraer2[$j]=strrev("$extraer[$j]");
//          $extraer3[$j]=substr("$extraer2[$j]",2);
//          if ($j>1)
//          {
//              $extraer4[$j]=strrev("$extraer3[$j]");
//              echo "<td>","<hr>";
//              echo "    <font size=2 color=black
face=Eurostile>",$extraer4[0]=strrev("$extraer3[0]"),"</font>"," | ";
//              echo "    <font size=2 color=black
face=Eurostile>",$extraer4[1]=strrev("$extraer3[1]"),"</font>";
//              echo "    <font size=2 color=black
face=Eurostile>","", $extraer4[2]=strrev("$extraer3[2]"),"</font>";
//              echo "<tr>";
//          }
//      }
    };
} // IF array posibilidades
} // fin IF principal

$x=0;
if (in_array($split[$x],$posibilidades2))
{
    if ($split[$x]=="bibliotecas" || $split[$x]=="biblioteca" ||
$split[$x]=="informacion")
    {
        echo "<font size=4 color=green face=Eurostile>","PORTALES (BIBLIOTECAS
VIRTUALES)","</font>","<br>";
        $rdql_query1='
SELECT ?Acceso, ?Descripcion, ?Nombre
WHERE (?Carrera, <kb:se_encuentra_en_portal>, ?Portal)
      (?Portal, <kb:disciplina>, "Especifica")
      (?Portal, <kb:acceso>, ?Acceso)
      (?Portal, <kb:descripcion>, ?Descripcion)
      (?Portal, <rdfs:comment>, ?Nombre)';

        $rdql_query1=$rdql_query1.' AND ?Carrera ~~ "/'. $split.' /" ';
        if ($cantidad>1)
        {
```



```
for ($i=1; $i<$cantidad; $i++)
{
    if (strlen($split[$i])>4)
    {
        $rdql_query1=$rdql_query1.' && ?Carrera ~ ~ "/.$split[$i]./" || ?Carrera
~ ~ "/.$split[$i]./" ';
    }
}
$rdql_query1=$rdql_query1.' , ?Carrera !~ "/Carreras/" USING kb FOR
<http://protege.stanford.edu/kb#> rdfs FOR <http://www.w3.org/2000/01/rdf-
schema#>;
}
$rdqlIter = $employees->rdqlQueryasIterator($rdql_query1);
$result_labels=$rdqlIter->getResultLabels();
$cantidad_resultados=$rdqlIter->countResults();

echo "<b>";"Se han encontrado "$cantidad_resultados," temas
relacionados","</b>";" <br>";

while ($rdqlIter->hasNext())
{
    $current_result=$rdqlIter->next();
    for ($j=0; $j<count($result_labels); $j++)
    {
        echo $cadena[$j]=$current_result[$result_labels[$j]];
// $extraer[$j]=substr("$cadena[$j]",9);
// $extraer2[$j]=strrev("$extraer[$j]");
// $extraer3[$j]=substr("$extraer2[$j]",2);
// if ($j>1)
// {
// $extraer4[$j]=strrev("$extraer3[$j]");
// echo "<td>";" <hr>";
// echo "<font size=2 color=black
face=Eurostile>";" $extraer4[0]=strrev("$extraer3[0]");" </font>";" | ";
// echo "<font size=2 color=black
face=Eurostile>";" $extraer4[1]=strrev("$extraer3[1]");" </font>";"
// echo "<font size=2 color=black
face=Eurostile>";" "" $extraer4[2]=strrev("$extraer3[2]");" </font>";"
// //echo $extraer4[4]=strrev("$extraer3[3]");" </td>";" </p>";"
// echo "<tr>";
// }
// }
};
}
}

$y=0;
if (in_array($split[$y],$posibilidades3))
{
    if ($split[$y]=="revistas" || $split[$y]=="revista" || $split[$y]=="informacion")
```



```
{
echo "<font size=4 color=green face=Eurostile>","REVISTAS","</font>","<br>";
$rdql_query2='
SELECT ?Acceso, ?Descripcion, ?Nombre
WHERE (?Carrera, <kb:se_encuentra_en_revistas>, ?Revistas)
      (?Revistas, <kb:acceso>, ?Acceso)
      (?Revistas, <kb:descripcion>, ?Descripcion)
      (?Revistas, <rdfs:comment>, ?Nombre)';
$rdql_query2=$rdql_query2.' AND ?Carrera ~~ "/'.$split.'/"' ;
if ($cantidad>1)
{
    for ($i=1; $i<$cantidad; $i++)
    {
        if (strlen($split[$i])>4)
        {
$rdql_query2=$rdql_query2.' && ?Carrera ~~ "/'.$split[$i].'" || ?Carrera ~~
"/'.$split[$i].'"';
        }
    }
$rdql_query2=$rdql_query2.' , ?Carrera !~ "/Carreras/" USING kb FOR
<http://protege.stanford.edu/kb#> rdfs FOR <http://www.w3.org/2000/01/rdf-
schema#>';
}
$rdqlIter = $employees->rdqlQueryasIterator($rdql_query2);
$result_labels=$rdqlIter->getResultLabels();
$cantidad_resultados=$rdqlIter->countResults();
echo "<b>","Se han encontrado ",$cantidad_resultados," temas
relacionados","</b>","<br>";
while ($rdqlIter->hasNext())
{
    $curren_result=$rdqlIter->next();
    for ($j=0; $j<count($result_labels); $j++)
    {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
//
// $extraer[$j]=substr("$cadena[$j]",9);
// $extraer2[$j]=strrev("$extraer[$j]");
// $extraer3[$j]=substr("$extraer2[$j]",2);
// if ($j>1)
//
// {
// $extraer4[$j]=strrev("$extraer3[$j]");
// echo "<td>","<hr>";
// echo "<font size=2 color=black
face=Eurostile>",$extraer4[0]=strrev("$extraer3[0]"),"</font>"," | ";
// echo "<font size=2 color=black
face=Eurostile>",$extraer4[1]=strrev("$extraer3[1]"),"</font>";
// echo "<font size=2 color=black
face=Eurostile>","",$extraer4[2]=strrev("$extraer3[2]"),"</font>";
// echo "<tr>";
// }
}
}
}
```



```
};
}
}
if (in_array($split[$a],$posibilidades4))
{
  if ($split[$a]=="paginas" || $split[$a]=="pagina" || $split[$a]=="informacion")
  {
    echo "<font size=4 color=green face=Eurostile>","PAGINAS
WEB", "</font>","<br>";
    $rdql_query3='
SELECT ?Acceso, ?PW, ?Descripcion, ?Nombre
WHERE (?Contenido, <kb:se_ve_en>, ?Asignatura)
      (?Asignatura, <kb:se_estudia_en>, ?Carrera)
      (?PW, <kb:tiene_informacion_de_pw>, ?Contenido)
      (?PW, <kb:acceso>, ?Acceso)
      (?PW, <kb:descripcion>, ?Descripcion)
      (?PW, <rdfs:comment>, ?Nombre)';
    $rdql_query3=$rdql_query3.' AND ?Contenido ~ "/'. $split[0]. '/' ';
    if ($scantidad>1)
    {
      for ($i=1; $i<$scantidad; $i++)
      {
        if (strlen($split[$i])>4)
        {
          $rdql_query3=$rdql_query3.' && ?Carrera ~ "/'. $split[$i]. '/' && ?Asignatura
~ "/'. $split[$i]. '/' || ?PW ~ "/'. $split[$i]. '/' ';
        }
      }
      $rdql_query3=$rdql_query3.' , ?Carrera !~ "/Carreras/" USING kb FOR
<http://protege.stanford.edu/kb#> rdfs FOR <http://www.w3.org/2000/01/rdf-
schema#>';
    }
    $rdqlIter = $employees->rdqlQueryasIterator($rdql_query3);
    $result_labels=$rdqlIter->getResultLabels();
    $scantidad_resultados=$rdqlIter->countResults();
    echo "<b>","Se ha encontrado ",$scantidad_resultados," tema(s)
relacionado(s)", "</b>","<br>";
    while ($rdqlIter->hasNext())
    {
      $curren_result=$rdqlIter->next();
      for ($j=0; $j<count($result_labels); $j++)
      {
        echo $cadena[$j]=$curren_result[$result_labels[$j]];
// $extraer[$j]=substr("$cadena[$j]",9);
// $extraer2[$j]=strrev("$extraer[$j]");
// $extraer3[$j]=substr("$extraer2[$j]",2);
// $extraer4[$j]=strrev("$extraer3[$j]");
// echo $extraer5[$j]=str_replace("_", " ",$extraer4[$j]),"<br>";
      }
    }
  }; } }
?>
```



#### **ANEXO IV: CÓDIGO BLOCK\_SIMPLEHTML.php**

El siguiente código corresponde al bloque de recomendaciones académicas implementado en el entorno EVA, de igual manera los archivos `config_global.html` y `config_instance.html`, que nos sirven para el diseño del bloque.

```
<script language="JavaScript">
function ventanaRecomendaciones (URL)
{
    window.open(URL,"ventana1","width=523,      height=527,      scrollbars=yes,
menubar=no,  location=yes,  resizable=yes,  toolbar=no,  directories=yes,
top=100,left=100")
}

function ventanaBuscador (URL)
{
    window.open(URL,"ventana1","width=523,      height=527,      scrollbars=yes,
menubar=yes,  location=yes,  resizable=yes,  toolbar=yes,  directories=no,
top=100,left=100")
}
</script>

<?PHP

class block_simplehtml extends block_list
{

    function init()
    {
        $this->title = get_string('simplehtml', 'block_simplehtml');
        $this->version = 2004111200;
    }

    function get_content()
    {

        if ($this->content !== NULL) {
            return $this->content;
        }

        if (empty($this->instance)) {
            $this->content = "";
            return $this->content;
        }

        $this->content = new stdClass;
        $this->content->items = array();
        $this->content->icons = array();
    }
}
```





```
$this->content->footer = "";

if ($this->instance->pageid != SITEID ||
    $CFG->showsiteparticipantslist > 1 ||
    ($CFG->showsiteparticipantslist == 1 && isteacherinanycourse()) ||
    isteacher(SITEID)) {

    $id_curso=$this->instance->pageid;

    require_once('../config.php');
    $pasid=get_record("course_utpl","courseid",$id_curso);
    $idpas=$pasid->pasid;

    $this->content->items[] = '-----';
    $this->content->items[] = '-----' <a
href="javascript:ventanaRecomendaciones(\'Recomendations/recomendador_defin
itivo.php?idpas='.$idpas.'\')">Recomendaciones</a>';

    }

    return $this->content;
}

function instance_allow_config()
{
    return true;
}

function specialization()
{
    $this->title = $this->config->title;
}

function instance_allow_multiple()
{
    return true;
}

function has_config()
{
    return true;
}

function config_save($data)
{
    {
        if(isset($data['block_simplehtml_strict'])) {
            set_config('block_simplehtml_strict', '1');
        }
        else {
            set_config('block_simplehtml_strict', '0');
        }
    }
}
```



```
    }
    return true;
}

function instance_config_save($data)
{
    // Limpia los datos si tenemos que hacerlo
    global $CFG;
    if(!empty($CFG->block_simplehtml_strict)) {
        $data['text'] = strip_tags($data['text']);
    }

    // Y ahora remite a la implementación por defecto definida en la clase padre
    return parent::instance_config_save($data);
}
}
?>
```

### **config\_global.html**

```
<div style="text-align: center;">
<input type="hidden" name="block_simplehtml_strict" value="0" />
<input type="checkbox" name="block_simplehtml_strict" value="1"
<?php if(!empty($CFG->block_simplehtml_strict)) echo 'checked="checked"'; ?>
/>
<?php print_string('donotallowhtml', 'block_simplehtml'); ?>
<p><input type="submit" value="<?php print_string('savechanges'); ?>" /></p>
</div>
```

### **config\_instance.html**

```
<table cellpadding="9" cellspacing="0">
<tr valign="top">
    <td align="right">
        <?php print_string('configcontent', 'block_simplehtml'); ?>:
    </td>
    <td>
        <?php print_textarea(true, 10, 50, 0, 0, 'text', $this->config->text); ?>
    </td>
</tr>
<tr>
    <td colspan="2" align="center">
        <input type="submit" value="<?php print_string('savechanges') ?>" />
    </td>
</tr>
</table>
<?php use_html_editor(); ?>
<tr valign="top">
```

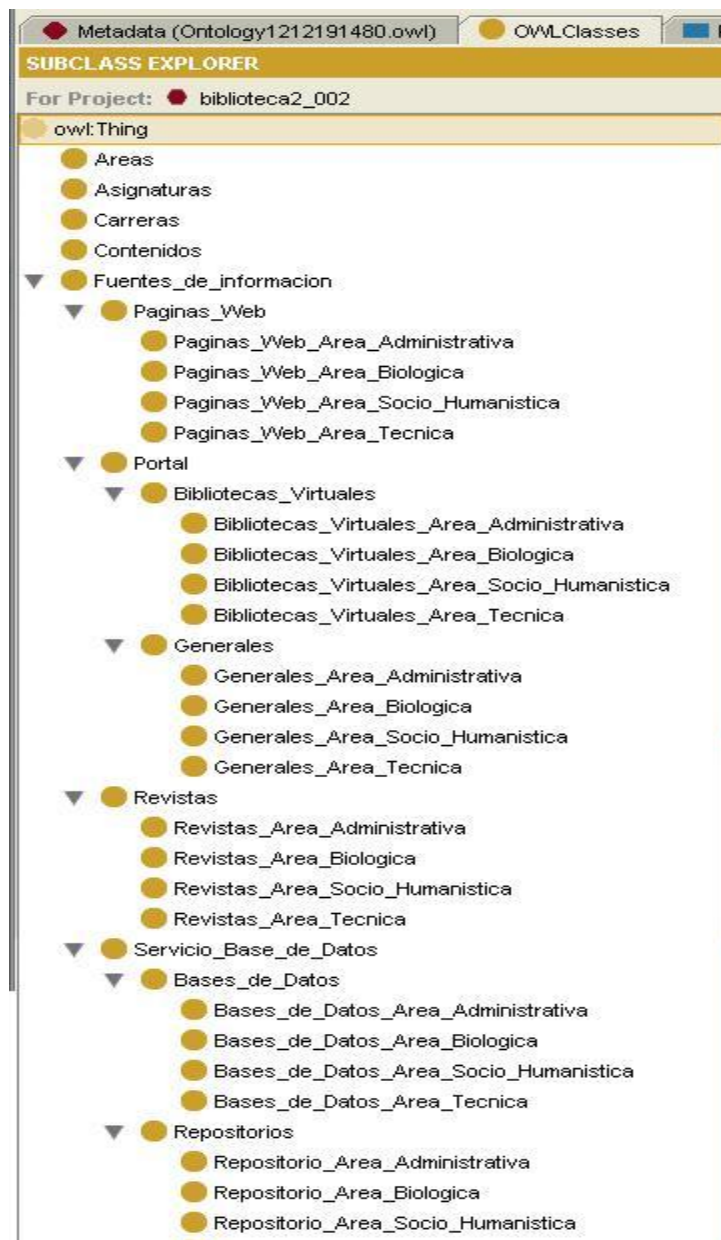


```
<td align="right"><p><?php print_string('configtitle', 'block_simplehtml');
?></td>
<td><input type="text" name="title" size="30" value="<?php echo $this->config-
>title; ?>" /></td>
</tr>
```



## ANEXO V: ONTOLOGÍA CON CLASES, PROPIEDADES OBJECT Y DATATYPE E INSTANCIAS

### CLASES





The image shows two side-by-side screenshots of a 'PROPERTY BROWSER' interface for the project 'biblioteca2\_002'. The left window is set to 'Datatype' and lists 20 properties such as 'idioma', 'busqueda', and 'nombre\_contenido'. The right window is set to 'Object' and lists 13 properties such as 'tiene\_informacion\_de\_sbd' and 'estudia', each with a bidirectional arrow icon.

Property Name
idioma
busqueda
oferta_academica
nombre_contenido
descripcion_contenido
acceso
fecha_publicacion
servicio
descripcion
nombre_asignatura
titulo
ISSN
nombre_area
Entidad_Editora
autor
disciplina
tipo
propietario
modalidad
volumen
nombre_carrera

Object Property Name
tiene_informacion_de_sbd ↔ se_encuentra_en_sbd
estudia ↔ se_ve_en
se_estudia_en ↔ imparte
tiene_informacion_de_portal ↔ se_encuentra_en_portal
imparte ↔ se_estudia_en
pertenece ↔ tiene
tiene ↔ pertenece
se_ve_en ↔ estudia
se_encuentra_en_pw ↔ tiene_informacion_de_pw
tiene_informacion_de_pw ↔ se_encuentra_en_pw
tiene_informacion_de_revistas ↔ se_encuentra_en_revistas
se_encuentra_en_portal ↔ tiene_informacion_de_portal
se_encuentra_en_sbd ↔ tiene_informacion_de_sbd
se_encuentra_en_revistas ↔ tiene_informacion_de_revistas

PROPIEDADES OBJECT Y DATATYPE



The screenshot displays the Protege software interface. On the left, the 'INSTANCE BROWSER' shows a list of 'Asserted Instances' for the class 'Carreras'. The 'economia' instance is highlighted. The main area is the 'INDIVIDUAL EDITOR' for the 'economia' instance. It features a table for 'rdfs:comment' and several property editors: 'nombre\_carrera' (value: 'Economia'), 'pertenece' (value: 'administrativa'), 'se\_encuentra\_en\_revis' (a list of values including 'RUCS\_A', 'RED\_A', 'E-Learning\_A', 'EJEL\_A', 'RIED\_A', 'E.FRM'), 'imparte' (a list of values including 'AS\_EC\_02\_microeconomia\_I\_Ec', 'AS\_EC\_03\_matematicas\_Ec', 'AS\_EC\_01\_macroeconomia\_I\_Ec'), 'se\_encuentra\_en\_porte' (a list of values including 'The\_Electronic\_Journals\_Library\_A', 'Voice\_of\_the\_Shuttle\_A', 'National\_Academy\_Press\_A', 'Virtual\_Library\_A', 'Internet\_Public\_Library\_A'), and 'se\_encuentra\_en\_sbd' (a list of values including 'Business\_Business\_Dateline', 'Interdisciplinary\_Proquest\_Childrens', 'Informe\_A', 'Regional\_Business\_News', 'General\_Media\_A').

## INSTANCIAS



## ANEXO VI: TABLA DE GLOSARIO DE TÉRMINOS

<b>Nombre del Concepto</b>	<b>Sinónimo</b>	<b>Descripción</b>	<b>Tipo</b>
Asignatura	Materia Disciplina Curso	Materias que son dictadas en periodos académicos	Concepto
Área	Superficie Espacio	Clasificación académica de acuerdo a las carreras	Concepto
Carrera	Corrida Persecución Huida	Diferentes especialidades por las que se puede optar.	Concepto
Contenido	Incluido Implícito	Lo que se imparte en determinadas asignaturas	Concepto
Administrativa	Funcionaria	Agrupación de carreras bajo un dominio común, dentro del ámbito administrativo	Instancia
Biológica	Biológico Orgánica	Agrupación de carreras bajo un dominio común, dentro del ámbito biológico	Instancia
Socio_Humanística		Agrupación de carreras bajo un dominio común, dentro del ámbito humanístico	Instancia
Técnica	Habilidad Destreza	Agrupación de carreras bajo un dominio común.	Instancia
Derecho	Abogacia	En la cual el estudiante estudia para ser abogado	Instancia
Administración de Empresas		Permite adquirir conocimiento para administrar empresas	Instancia



Administración en Banca y Finanzas		Conocimiento relacionado con las finanzas y banca	Instancia
<b>Nombre del Concepto</b>	<b>Sinónimo</b>	<b>Descripción</b>	<b>Tipo</b>
Administración en Gestión Pública		Relaciones publicas y mas	Instancia
Estudia	Estudiar Aprende	Relación entre Asignaturas y su Contenido	Relación
Tiene	Agrupar	Relacionar las materias agrupadas dentro de un dominio común	Relación
Pertenece	Pertenecer	Cuando se dice que una carrera o materia pertenece a un área determinada	Relación
Imparte	Impartir	Mediante esto se indica que una carrera tiene asignaturas	Relación
Ciclo contable		Contenido que se imparte en la asignatura de Economía I	Instancia
Estudio de las cuentas del estado de resultados		Contenido que se imparte en la Asignatura Economía I	Instancia
La empresa y la Contabilidad		Contenido que se imparte en la asignatura Economía I	Instancia
Páginas web		Página sencilla que contiene información sobre un tema específico	Clase
Revistas		Publicaciones electrónicas mensuales o quincenales de acuerdo a la entidad editora	Clase
Repositorio	Almacén	Contenido de información organizada de acuerdo a una estructura jerárquica	Clase
Portal	Galería	Página web con contenido variado de acuerdo a su tipo de publicación,	Clase





		puede brindar servicios	
Bases de datos		Contiene información electrónica sobre diferentes temas	Clase
<b>Nombre del Concepto</b>	<b>Sinónimo</b>	<b>Descripción</b>	<b>Tipo</b>
Biblioteca Virtual		Contiene información variada sobre diferentes temas relacionados con varias áreas del conocimiento	Clase
Recurso Digital		Cualquier documento almacenado o disponible en formato digital	Clase
Biblioteca Dominico Virtual		Información sobre variados temas educativos: Historia,...	Instancia
Biblioteca Jurídica UTPL		Esta biblioteca pertenece a la UTPL, posee información relacionada con la constitución Política del Ecuador	Instancia
Business and Economics		Repositorio que contiene información sobre economía y negocios	Instancia
Business Business Dateline		Información relacionada acerca diferentes negocios	Instancia



**ANEXO VII: TABLA DEL DICCIONARIO DE CONCEPTOS**

<b>Nombre del concepto</b>	<b>Instancias</b>	<b>Atributos de la clase</b>	<b>Atributos de la instancia</b>	<b>Relaciones</b>
Área	Biológica Técnica Socio Humanística Administrativa	nombre_área		Tiene (carrera)
Asignatura	Zoología General Writing III Balance de Materia y Energía Iniciación a la Lecto- estructura Archivística Escultura V Bioquímica Administración en Salud	Nombre_asignatura oferta_académica ciclo modalidad tipo	Pregrado Postgrado Pregrado y Posgrado Clásica - Abierta Virtual Semipresencial Clásica y Abierta Troncal de Carrera Formación Básica o Comunes De Carrera Genéricas de Carrera Complementarias Libre	se_estudia_en (carreras) estudia (contenidos)



Nombre del concepto	Instancias	Atributos de la clase	Atributos de la instancia	Relaciones
Contenido	Ciclo Contable Estudio de las cuentas del estado de resultados La empresa y la contabilidad Plan de cuentas y descripción de las cuentas	nombre_contenido		se_encuentra_en_pw (Paginas_web) se_ve_en (asignaturas)
Carreras	Abogacía Administración de Empresas Economía Sistemas Civil Arquitectura Arte y Diseño	nombre_carrera		Pertenece (área) Imparte (asignatura) se_encuentra_en_portal (portal) se_encuentra_en_sbd (servicio_de_base_datos) se_encuentra_en_revistas (revistas)
Fuentes de Información		Descripción Acceso Idioma		
Páginas Web	Árbol binario Matemática aplicadas a la computación	Descripción Acceso Idioma	Titulo Autor Español Ingles	tiene_informacion_de_pw (contenidos)



	Teoría de conjuntos Tutorial orientado a objetos Teoría sobre oferta y demanda		Japonés Otros Libre Acceso Acceso Restringido	
Biblioteca Virtual	Biblioteca Dominico Virtual Internet Public Library Biblioteca Virtual de la Salud Biblioteca Jurídica UTPL	Descripción Acceso Idioma Búsqueda Propietario Disciplina Servicios	Español Ingles Japonés Otros Multidisciplina ria Especifica Búsqueda Básica Búsqueda Avanzada No posee búsqueda Foros Servicio de correo	
Portal		Descripción Acceso Idioma		tiene_informacion_de_portal (carrera)
Revistas	E-Learning Revista Iberoamericana de Educación a Distancia Revista de Educación a Distancia Electronic Journal of e-Learning Electronic Journal of Knowledge	Descripción Acceso Idioma	Fecha de Publicación Volumen Entidad Editora ISSN	tiene_informacion_de_revistas (carreras)



Nombre del concepto	Instancias	Atributos de la clase	Atributos de la instancia	Relaciones
Base de datos	Academic OneFile Evidence Based Medicine Health Economic Evaluations Database Academic OneFile Interdisciplinar y Dissertations y Theses Social Sciences Criminal Justice Periodicals Medical Sciences Nursing y Allied Health Source	Descripción Acceso Idioma Propietario Disciplina	Libre acceso Acceso restringido Español Ingles Japonés Otros Multidisciplina ria Especifica Búsqueda Básica Búsqueda Avanzada Búsqueda Publicación Búsqueda por Guía de Temas Búsqueda Visual	
Repositorios	Languages and Literatures Social Sciences Law and Political Science	Descripción Acceso Idioma Propietario Disciplina	Libre acceso Acceso restringido Español Ingles Japonés	



	Technology and Engineering Mathematics and Statistics Arts and Architecture		Otros Multidisciplina ria Especifica	
Servicio Base de Datos		Descripción Acceso Idioma Propietario Disciplina		tiene_informacion _de_sbd



### ANEXO VIII: TABLA DE ACEPTACIÓN DE USUARIOS

	<b>Excelente</b>	<b>Muy Bueno</b>	<b>Bueno</b>	<b>Regular</b>
Tiempo en espera de solicitud				
La información es confiable				
Adecuada interfaz de presentación al usuario				
Información útil en la interfaz de usuario				