UNIVERSIDAD POLITÉCNICA DE PUEBLA

Maestría en Ingeniería en Sistemas y Cómputo Inteligente

"Creación de colecciones de mapas conceptuales en Greenstone"

Tesis de Maestría

Eduardo Hernández Ronquillo

Eduardo Hernández Ronquillo, Puebla. Noviembre 2012

UNIVERSIDAD POLITÉCNICA DE PUEBLA

Maestría en Ingeniería en Sistemas y Cómputo Inteligente

"Creación de colecciones de mapas conceptuales en Greenstone"

Eduardo Hernández Ronquillo

Tesis de maestría

Comité Evaluador

Dra. María Auxilio Medina Nieto
Asesor

Dr. Jorge de la Calleja Mora
Sinodal

Dr. Antonio Benítez Ruiz
Sinodal

Eduardo Hernández Ronquillo, Puebla. Noviembre 2012

Capítulo 1

Planteamiento del problema de investigación

1.1. Introducción

El concepto de biblioteca digital no debe entenderse como la digitalización de documentos, sino como un entorno que integra colecciones, servicios y personal para favorecer el ciclo de creación, difusión, uso y preservación de datos que se transforman en información y conocimiento. Entre las tareas de una biblioteca digital, se consideran las relacionadas con la preservación y también aquellas que involucran el manejo de estructuras de organización y servicios que agregan valor a los datos. Si bien en las bibliotecas digitales se utilizan metadatos y vocabularios controlados, dado el volumen de datos que se administra, son necesarios mecanismos de acceso y procesamiento que interpreten los datos de forma no ambigua, para ello se recomienda el empleo de estándares de metadatos y la incorporación de elementos semánticos.

Las herramientas de la web semántica promueven un cambio en la arquitectura de las bibliotecas digitales tradicionales, su propósito es contribuir al mejoramiento en la recuperación y la organización de la información. Una herramienta que pueda ayudar a construir y distribuir colecciones digitales es Greenstone, software de distribución libre

que provee mecanismos denominados clasificadores para organizar y publicar información en la web. La finalidad de Greenstone es permitir que los usuarios, en particular universidades, bibliotecas y otras instituciones públicas, puedan crear, mantener y distribuir sus propias colecciones.

El contenido de las colecciones no siempre es texto, en ocasiones se representa en forma gráfica como en los mapas conceptuales. Los mapas conceptuales son una herramienta para organizar y representar el conocimiento; fueron desarrollados con el objetivo de incrementar el aprendizaje en los estudiantes, sin embargo, su uso actualmente es más extenso. Existen herramientas de software que ayudan a generar estos mapas como Cmap Tools, que es un software de distribución libre [6].

Este trabajo de tesis describe un estudio de la interoperabilidad entre Cmap Tools y Greenstone, con el propósito de construir colecciones de mapas conceptuales para dominios específicos con mecanismos de recuperación de información sobre el contenido y descripción basada en estándares de metadatos. A manera de prueba, se recolectarán mapas conceptuales de dominios de interés general y de computación.

1.2. Objetivo general

Construir colecciones de mapas conceptuales a partir del análisis de los niveles de interoperabilidad entre GreenStone y Cmap Tools.

1.3. Objetivos específicos

- Identificar los niveles de interoperabilidad entre GreenStone y Cmap Tools.
- Construir una colección de mapas conceptuales de diferentes dominios de interés.

- Utilizar estándares de metadatos para describir a los elementos de la colección.
- Implementar mecanismos de consulta y exploración accesibles vía web.

1.4. Justificación

Cmap Tools es una herramienta multiplataforma para construir y transmitir información representada en forma de mapas conceptuales, permite incorporar notas, imágenes y estilos. Cmap Tools cuenta con la capacidad para exportar los mapas conceptuales a páginas web, archivos JPG y otros formatos. Por otro lado, Greenstone gestiona documentos también en diferentes formatos convirtiendólos en una colección de uso práctico. Las colecciones de Greenstone, entendidas como organizaciones jerárquicas de documentos, pueden ser accesibles a través de cualquier navegador web, pueden ejecutarse en servidores Linux o Windows y distribuirse a través de CD-ROM. Dado que Cmap Tools permite trabajar con archivos con extensión txt, doc, pdf, jpg, html, xml, odf, xls, postscript, rtf, o latex y que Greenstone puede crear colecciones con estos tipos de archivos, se considera factible su interoperación con CMapTools. La justificación de este trabajo de tesis, es la de proveer colecciones indizadas que integren la información de los mapas conceptuales en colecciones de Greenstone. En específico, se considera un grupo de mapas conceptuales desarrollados en diferentes instituciones que representan un conjunto de datos de prueba para formar colecciones. La colecciones contendrán mecanismos de consulta. Desde el punto de vista de los usuarios principales, a saber, profesores y estudiantes, la colecciones permiten mantener un conjunto de recursos educativos con representación gráfica sencilla que ayuda a reforzar conocimientos básicos. Desde un enfoque técnico, en el proyecto se explora la extracción automática de metadatos provenientes de CMapTools y su mapeo en Greenstone. La colección de

mapas conceptuales estará accesible vía web.

1.5. Recursos de hardware y software

Para el desarrollo de este proyecto, se requieren los recursos siguientes:

1.5.1. Recursos de hardware

- Memoria RAM 4 Gb.
- Procesador Intel Core i5.
- Disco Duro 320 Gb.

1.5.2. Recursos de software

- Software para bibliotecas digitales: Greenstone versión 2.85 [1]
- Cmap Tools versión 5.04 como generador de mapas conceptuales [2]

1.6. Alcances y limitaciones

- Las colecciones de mapas conceptuales estarán disponibles vía web, de manera que podrán consultarse en el momento que se requiera.
- La colecciones abordarán temas relacionados con el área de Computación.
- El número de mapas conceptuales que tendrán la colecciones serán mayor a 20. Este número se determina de forma experimental.

[1] http://www.greenstone.org/download

[2] http://cmap.ihmc.us/

- Los mecanismos de consulta de los mapas conceptuales serán por contenido y por metadatos, a saber, palabras clave, autor, tema y título.

Capítulo 2

Marco teórico

En este capítulo se hace una breve descripción de los mapas conceptuales así como de las herramientas de software que son utilizadas para crearlos. También se describe qué es la web semántica, los elementos que utiliza y de los lenguajes más comunes que existen para representarla. Además, se hace mención de qué tratan las bibliotecas digitales y de las principales herramientas que existen para crearlas.

2.1. Mapas conceptuales

Los mapas conceptuales son diagramas para representar ideas, tareas u otros conceptos que se encuentran relacionados con una palabra clave o idea central y que se ubican radicalmente a su alrededor. Se visualizan en foma de grafo, donde los nodos representan los conceptos y los enlaces las relaciones entre los conceptos. Tienen como objetivo lo siguiente [9]:

- Generar ideas
- Comunicar ideas complejas
- Fomentar el aprendizaje significativo para mejorar el éxito de los estudiantes

- Medir la comprensión de conceptos
- Explorar el conocimiento previo y los errores de concepto

2.1.1. Cmap Tools

Cmap Tools se desarrolló en el IHMC (Institute for Human and Machine Cognition), es un software multiplataforma para crear mapas conceptuales. Permite tanto el trabajo local e individual como en red, lo que facilita el trabajo en grupo o colaborativo. Algunas de las características de Cmap Tools son las siguientes [6]:

- Facilidad de uso
- Es gratuito
- Posee herramientas de edición
- Se da formato automáticamente

Es importante mencionar que Cmap Tools dispone de un acceso vía Internet a una gran colección de trabajos que pueden servir como guía para el proyecto, o simplemente como base para empezar a diseñar un mapa conceptual.

Se pueden insertar recursos tales como archivos de texto, imágenes y otros formatos de archivos a la colección; además convertir los esquemas directamente en formato web es otra de las aportaciones de esta herramienta. Sin embargo, el formato en el que se enfoca este trabajo de tesis para exportar los mapas conceptuales es el tipo CXL[1] , un lenguaje basado en XML para describir el contenido de los mapas conceptuales.

[1] Las siglas CXL se utilizan para hacer referencia al formato Concept Mapping Extensible Language, forma parte de la arquitectura KEA (Knowledge Exchange Arquitecture)

En la siguiente sección se explica qué es la web semántica, los metadatos y las ontologías; que son la base para esta web y que además ambos son utilizados para crear bibliotecas digitales con características semánticas.

2.2. La web semántica

La web semántica es una web extendida dotada de significados. Identifica un conjunto de tecnologías, herramientas y estándares que forman los bloques básicos de una infraestructura que da soporte a la visión de la web asociada con el significado de los datos [4].

Se pueden obtener soluciones a problemas habituales en la búsqueda de información debido a la utilización de una infraestructura común mediante la cual es posible compartir, procesar y transferir información de forma sencilla. La web semántica se basa en la idea de añadir metadatos y ontologías a la World Wide Web. Esta información adicional describe el contenido, el significado y la relación de los datos que se deben proporcionar de manera formal para que sea posible evaluarlas automáticamente por máquinas de procesamiento, con interpretación relativamente sencilla para los usuarios.

La arquitectura de la web semántica se compone de una serie de estándares organizados en una cierta estructura que es una expresión de sus interrelaciones. Esta arquitectura es a menudo representada usando el diagrama propuesto por primera vez por Tim Berners-Lee [10]. Este diagrama se muestra en la figura 2.1.

La web semántica tiene dos elementos: los metadatos y las ontologías. A partir de estos elementos y mediante el uso de lenguajes de representación de ontologías, se construyen las aplicaciones para el usuario. Las secciones 2.2.1 y 2.2.2 describen estos elementos clave.

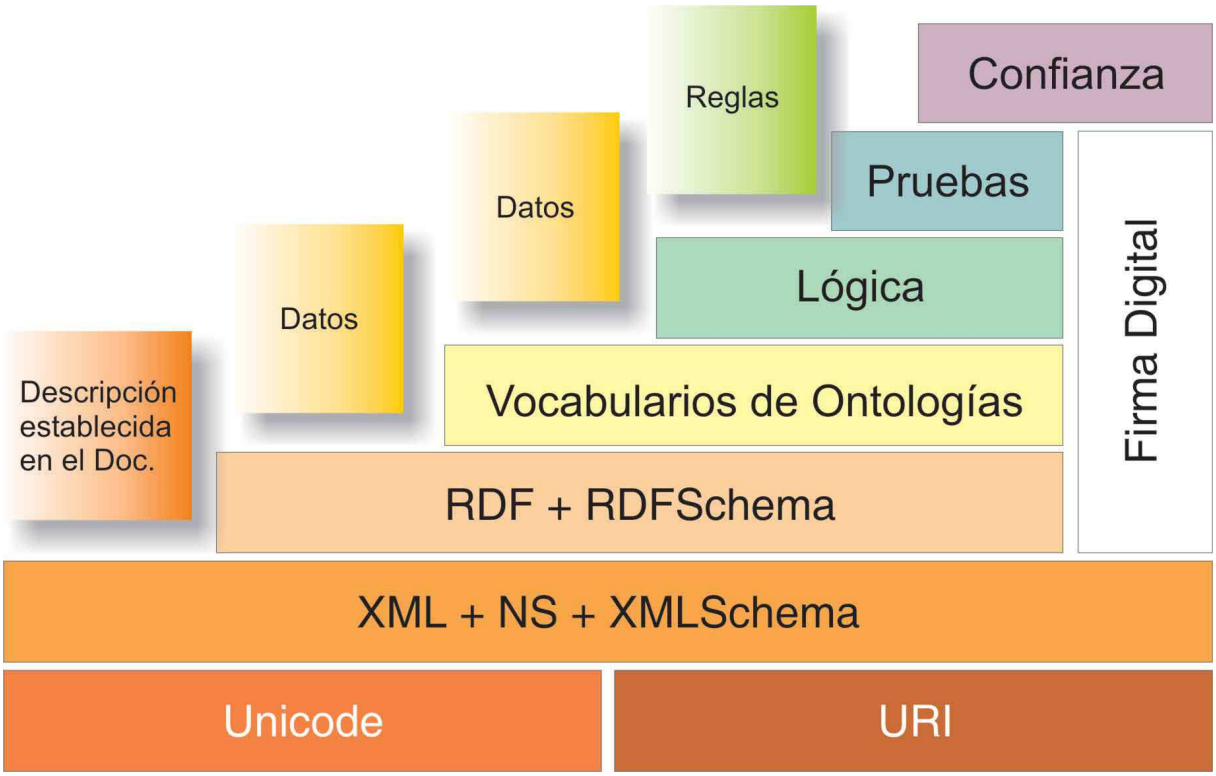

Figura 2.1: Diagrama de capas para la web semántica propuesta por Tim Berners-Lee[10].

2.2.1. Metadatos

Los metadatos son datos que describen otros datos que ayudan a identificar, describir y localizar recursos. Estos recursos pueden ser de tipo texto, material visual, iconografía, etc. Pueden describir una colección en general, un recurso en particular o un solo elemento. Además, proporcionan información del recurso tal como su contenido, contexto y estructura [12]. Estos elementos se describen a continuación:

- *Contenido:* se refiere al asunto o materia sobre lo que trata el documento.
- *Contexto:* integra todos los aspectos relacionados con la creación del objeto de información tales como quien, qué, por qué, dónde y cómo.
- *Estructura:* hace referencia al conjunto formal de relaciones entre objetos de in-

formación.

Se puede realizar una clasificación de los metadatos por su tipo en [12]:

- *Metadatos descriptivos:* descripción e identificación de los recursos para permitir la búsqueda y recuperación de una determinada categoría de documentos o imágenes.
- *Metadatos estructurales:* facilitan la navegación y presentación de los recursos electrónicos proporcionando información sobre su estructura interna tal como página, sección, capítulo, etc.
- *Metadatos administrativos:* facilitan la gestión y procesamiento de las colecciones digitales.

Dublin Core(DC) es un modelo de metadatos elaborado por la DCMI (Dublin Core Metadata Initiative). Es un conjunto de 15 elementos de metadatos que pretenden transmitir un significado a los datos (información semántica de los datos). Este conjunto de elementos fue diseñado para proporcionar un vocabulario de características base, capaces de proporcionar información descriptiva básica de cualquier recurso, sin que importe el formato de origen, el área de especialización o el origen cultural.

Los elementos de DC se pueden clasificar en tres grupos que indican la clase o el ámbito de la información que se guarda en ellos [7]:

Grupo 1: Elementos relacionados con el contenido del recurso.

- **Título:** el nombre dado a un recurso, habitualmente lo asigna el autor. Etiqueta: `DC.Title`
- **Temas:** los tópicos del recurso. Típicamente, Subject expresa las claves o frases que describen el título o el contenido del recurso. Etiqueta: `DC.Subject`

- **Descripción:** una descripción textual del recurso. Etiqueta: `DC.Description`
- **Fuente:** secuencia de caracteres usados para identificar unívocamente un trabajo a partir del cual proviene el recurso actual. Etiqueta: `DC.Source`
- **Lengua:** lengua del contenido intelectual del recurso. Etiqueta: `DC.Language`
- **Relación:** es un identificador de un segundo recurso y su relación con el recurso actual. Este elemento permite enlazar los recursos relacionados y las descripciones de los recursos. Etiqueta: `DC.Relation`
- **Cobertura:** es la característica de cobertura espacial y/o temporal del contenido intelectual del recurso. Etiqueta: `DC.Coverage`

Grupo 2: Elementos relacionados con el recurso visto como propiedad intelectual.

- **Autor o creador:** la persona u organización responsable de la creación del contenido intelectual del recurso. Por ejemplo, los autores de un libro. Etiqueta: `DC.Creator`
- **Editor:** la entidad responsable de hacer que el recurso se encuentre disponible en la red en su formato actual. Etiqueta: `DC.Publisher`
- **Otros colaboradores:** una persona u organización que haya tenido una contribución intelectual significativa. Etiqueta: `DC.Contributor`
- **Derechos:** son una referencia (por ejemplo, una URL) para una nota sobre derechos de autor. Etiqueta: `DC.Rights`

Grupo 3: Elementos relacionados con la instanciación del recurso.

- **Fecha:** una fecha en la cual el recurso se puso a disposición del usuario en su forma actual. Etiqueta: `DC.Date`

- **Tipo del recurso:** la categoría del recurso. Por ejemplo, página personal, romance, poema, diccionario, libro, revista, etc. Etiqueta: `DC.Type`

- **Formato:** es el formato de datos de un recurso, se usa para identificar el software y, posiblemente, el hardware que se necesitaría para mostrar el recurso. Etiqueta: `DC.Format`

- **Identificador del recurso:** secuencia de caracteres utilizados para identificar unívocamente un recurso. Ejemplos para recursos en línea pueden ser URIs [2] y URNs [3] . Para otros recursos pueden ser usados otros formatos de identificadores por ejemplo ISBN. Etiqueta: `DC.Identifier`

2.2.2. Ontologías

Una ontología es una especificación explícita de una conceptualización [11]. Una ontología es un sistema de representación del conocimiento que resulta de seleccionar un dominio o ámbito del conocimiento y aplicar sobre él un método con el fin de obtener una representación formal de los conceptos que contiene y de las relaciones que existen entre dichos conceptos [1].

La web semántica requiere interoperabilidad con los metadatos asociados y la información web, por consiguiente, el papel de las ontologías es proveer vocabularios para describir los metadatos con una semántica comprensible por los humanos y las máquinas [3].

Algunas características de las ontologías son las siguientes [4]:

- Usan un vocabulario en común

[2] Por sus siglas en Inglés: Uniform Resource Identifier
[3] Por sus siglas en Inglés: Uniform Resource Name

- Tienen una estructura que sirve como esquema conceptual para la descripción e intercambio de información
- Representan reglas semánticas en un tipo de lenguaje basado en lógica descriptiva
- Eliminan ambigüedades al definir conceptos de manera única y precisa
- Aparecen como una taxonomía en un árbol conceptual

Es importante mencionar que una taxonomía es una forma de clasificar o categorizar un conjunto de cosas en forma de jerarquía, también se puede definir como una ontología a la que se le agrega el adjetivo light, ligera. Una jerarquía es una estructura de árbol, tiene una raíz y ramas. Cada nodo de la taxonomía incluyendo la raíz, es una entidad de información que representa una entidad del mundo real y cada enlace entre nodos representa una relación especial llamada: *clasificación de* (si la flecha del enlace está apuntando hacia el nodo padre) o *superclasificación de* (si la flecha del enlace está apuntando hacia abajo, en el nodo hijo). Cuando uno se dirige a la raíz de la taxonomía, las entidades llegan a ser más generales y cuando uno se dirige hacia las hojas, las entidades llegan a ser más especializadas [4].

Algunos de los lenguajes de etiquetado utilizados frecuentemente para representar información semántica y ontologías se describen a continuación.

2.2.3. Lenguajes de la web semántica

La web semántica emplea lenguajes de etiquetado como XML, XML Schema, RDF y OWL para representar metadatos y ontologías. A continuación se describen las características principales de estos lenguajes:

- **XML** es un metalenguaje, un conjunto de reglas de sintaxis para crear lenguajes de marcado semántico en un dominio particular [4]. Cualquier lenguaje creado a

través de las reglas XML se conoce como aplicación de XML. El uso principal de XML es el intercambio de información, sin embargo, otros beneficios son [4]:

- Crear documentos y datos independientes de la aplicación
- Tener una sintaxis estándar para metadatos
- Ofrecer una estructura estándar para documentos y datos

Un documento en XML está formado de elementos y atributos. Un elemento XML es un contenedor que consiste de una etiqueta inicial, contenido (que pueden ser datos, subelementos o ambos) y una etiqueta final, excepto para elementos vacíos que usan sólo una etiqueta denotando el inicio y el fin del elemento. Un ejemplo de un elemento es el siguiente:

<book>

<author> Michael C. Daconta </author>, <title> Java 1</title>

</book>

En el ejemplo anterior se presenta un elemento llamado `book`, formado por los subelementos: `author` y `title`.

Etiquetar el contenido es dividir el documento en partes semánticas. La creación de diversas partes de una entidad completa permite clasificar o agrupar partes y por lo tanto, tratarlas de forma diferente basándose en la pertenencia en un grupo. En XML, tal clasificación empieza restringiendo un documento válido de aquel que está compuesto de un solo elemento llamado `root`. A su vez, este elemento puede contener otros elementos o contenido.

- **XML Schema** es un lenguaje utilizado para describir la estructura y las restricciones de los documentos XML de forma precisa. XML Schema es similar a un esquema de una base de datos, en donde se definen los nombres de las columnas y los tipos de datos en las tablas. Usa sintaxis XML para declarar un conjunto simple o complejo de tipos de declaraciones. Un tipo es una plantilla con nombre que puede conservar uno o más valores, los tipos simples conservan sólo un valor. Así, un tipo tiene dos características principales: un nombre y un conjunto de valores.

- **RDF** (Resource Description Framework) es un modelo de datos que representa conocimiento en forma de enunciados. Para ser procesable por la máquina, se asocia con XML. Mientras XML liga metadatos de un documento, RDF crea metadatos del documento como una entidad única, es decir RDF captura metadatos del documento como es el autor, la fecha de creación y el tipo. El modelo RDF se conoce también como modelo de 3 partes. En la comunidad de representación de conocimiento, estas partes se describen en términos de gramática como partes de una sentencia: sujeto, predicado y objeto.

  La tabla 2.1 muestra un ejemplo de elemento `libro`, su relación con un objeto `Autor` y valores en el modelo RDF.

Tabla 2.1: Modelo RDF

Sujeto	Predicado	Objeto
Java1	es un	Libro
Libro	tiene	Autor
Michael C. Daconta	es Autor de	Java1

- **OWL** (Ontology Web Languaje) es un lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW. OWL extiende las propiedades

de modelado construido sobre RDF y codificado en XML. OWL ofrece mayor capacidad de representación que RDF al definir un vocabulario que incorpora clases, relaciones y restricciones.

2.2.4. Bibliotecas digitales con características semánticas

Los documentos descritos mediante metadatos y ontologías pueden formar colecciones, que a su vez se consideran componentes de bibliotecas digitales semánticas que son sistemas de bibliotecas digitales que aplican tecnologías semánticas para llevar a cabo sus objetivos. Con la aplicación de ontologías, se pueden combinar consultas estructuradas de metadatos con búsquedas en texto completo de los recursos, así también lograr una interoperabilidad entre sistemas.

Algunas tareas de aplicación de las ontologías en bibliotecas digitales semánticas son [10]:

Ontologías bibliográficas.- Usualmente una biblioteca digital usa un cierto formato de metadatos para organizar las descripciones bibliográficas. Una de las ontologías bibliográficas es MarcOnt, utilizada en JeromeDL, su objetivo es combinar diferentes estándares de metadatos que pueden describir varios conceptos en diferentes niveles de granularidad.

Ontologías para estructuras de contenido.- Las bibliotecas digitales no sólo almacenan metadatos bibliográficos sino también una representación electrónica del contenido. Típicamente, el contenido sigue alguna estructura que puede utilizarse para proveer una capa de metadatos y acceder a secciones específicas. Esta capa permite extender la descripción de los recursos con nuevos conceptos, sin cambiar el esquema fundamental de la base de datos o violar la integridad de la información existente.

Ontologías de comunidad.- Normalmente, una biblioteca es una institución que sirve

a ciertos grupos de usuarios o comunidades. El usuario individual no sólo requiere incrementar su conocimiento, sino poder compartir su experiencia con ciertos recursos.

En una biblioteca digital semántica, además de almacenar contenido y metadatos, se desea incorporar el conocimiento, la opinión y la experiencia de los usuarios. La siguiente sección contiene una breve descripción de herramientas de software utilizadas para construir bibliotecas digitales semánticas.

2.2.5. Como construir bibliotecas digitales semánticas

Para la construcción de bibliotecas digitales semánticas, existen diferentes programas que realizan este propósito como JeromeDL [8], DSpace[5] y Greenstone por mencionar algunos de los más conocidos y que son de distribución libre.

- *JeromeDL.-* Es una biblioteca digital semántica, permite a las instituciones publicar documentos en la web. Soporta una variedad de formatos de documentos, permite guardar y consultar una descripción bibliográfica de cada documento. Los usuarios pueden buscar documentos por palabras clave.

- *DSpace.-* Es un software de código abierto que provee herramientas para la administración de colecciones digitales. Soporta una gran variedad de datos incluyendo libros, tesis, fotografías y otras formas de contenido. Los datos son organizados como *items* que pertenecen a una colección y cada colección pertenecen a una comunidad.

- *Greenstone.-* Es un conjunto de programas de software diseñado para crear y distribuir colecciones digitales, proporcionando así una nueva forma de organizar y publicar la información a través de la web o en forma de CD-ROM. Surgió en el Proyecto Biblioteca Digital de Nueva Zelanda.

En esta tesis se emplea Greenstone ya que ha sido desarrollado y distribuido en colaboración con la UNESCO[4] y la ONG[5] . Greenstone es un conjunto de programas y aplicaciones de software especialmente diseñados para la creación y difusión de colecciones de documentos digitales, ofrece formas para organizar la información y publicarla en la web.

En la mayoría de las colecciones de Greenstone existen varias maneras de encontrar información. Por ejemplo, se pueden buscar palabras específicas que aparecen en un texto en una sección de un documento, se pueden consultar documentos por título, por tema, por autor, entre otros metadatos, además de que indexa el contenido. El indexado y la interfaz de usuario puede configurarse para diferentes idiomas como inglés, español, francés o portugués.

Greenstone crea índices a partir del texto del documento, es decir, índices que permiten buscar palabras en el contenido del documento. Se pueden buscar en los índices determinadas palabras, combinaciones de palabras o frases.

En la mayoría de las colecciones, los datos descriptivos como el autor, el título, la fecha, las palabras clave, etc., se asocian a cada documento. Estos metadatos se emplean en las consultas. Los índices se suministran explícitamente o pueden derivarse automáticamente de los propios documentos durante un proceso de creación que hace uso de los metadatos.

Los documentos de origen se presentan en diversos formatos y se convierten a un formato normalizado XML para la indexación mendiante plugins. Los plugins distribuidos con Greenstone procesan archivos con formato html, word, pdf, por citar algunos de los más comunes. Pueden escribirse nuevos plugins para otro tipo de documentos, según las necesidades de cada contexto de aplicación. En este proyecto, es de interés

[4] UNESCO: United Nations Educational, Scientific and Cultural Organization
[5] Organización No Gubernamental

especial el formato CXL, el cual se describe en el siguiente capítulo.

Capítulo 3

Metodología

En este capítulo se menciona la arquitectura KEA [2] y su relación con Cmap Tools. Además se trata de los requerimientos funcionales y no funcionales del sistema desarrollado en esta tesis, se presenta un diagrama de casos de uso para el mismo.

3.1. KEA

El uso de CmapTools está experimentando un crecimiento rápido que necesita de nuevas formas de manipular los Cmaps (mapas conceptuales construidos con Cmap Tools). Lo mencionado anteriormente condujo a una nueva arquitectura abierta llamada KEA[1] que provee los datos y los procedimientos para permitir a los desarrolladores diseñar programas que interactúen con Cmap Tools. KEA consiste de un formato de archivo portable estándar para el intercambio de información (CXL) y un conjunto de protocolos para recuperar, almacenar y manipular la información del Cmap. Se enfatiza en la siguiente sección sobre el formato CXL.

[1] Por sus siglas en inglés: Knowledge Exchange Architecture

3.1.1. CXL

CXL[2] es un lenguaje basado en XML para describir el contenido de Cmaps. La implementación de esta representación es el primer paso para establecer a CXL como un estándar internacional para guardar los mapas conceptuales así como el mecanismo para el intercambio de mapas entre aplicaciones.

Existen cuatro secciones básicas en un archivo CXL. La primera sección define el tipo de documento XML y el espacio de nombres. La segunda especifica un elemento `'res-meta'`, que contiene la información del recurso de Cmap tal como el título, descripción y autor. La tercera sección es la parte de datos de Cmap que contiene la lista de conceptos y frases que se ligan entre sí. La sección final define la información gráfica tal como la localización, tamaño, colores y fuentes definidas de la sección datos.

En la Figura 3.1 se muestra la representación de un mapa conceptual que representa un sólo enunciado. Los rectángulos con esquinas redondeadas se utilizan como contenedores de conceptos. La leyenda en la línea permite relacionar los conceptos.

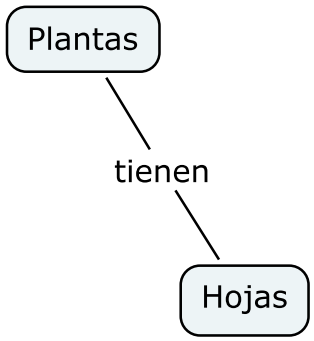

Figura 3.1: Mapa conceptual generado con Cmap Tools

La Figura 3.2 muestra el archivo CXL correspondiente a la Figura 3.1:

El formato CXL está definido por el esquema XML cmap.xsd[3] . Su estructura puede

[2] Por sus siglas en inglés: standard file format for data exchange
[3] Su definición se encuentra en http://cmap.ihmc.us/xml/cmap.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<cmap xmlns="http://cmap.ihmc.us/xml/cmap.dtd"
xmlns:dc="http://purl.org/dc/elements/1.1/">

<res-meta>
       <dc:title>Plantas</dc:title>
       <dc:description>¿Como crecen las plantas?</dc:description>
</res-meta>

<map>
       <concept-list>
              <concept id="1" label="Plantas"/>
              <concept id="2" label="Hojas"/>
       </concept-list>

       <linking-phrase-list>
              <linking-phrase id="3" label="tienen"/>
       </linking-phrase-list>

       <connection-list>
              <connection from-id="1" to-id="3"/>
              <connection from-id="3" to-id="2"/>
       </connection-list>

       <concept-appearance-list>
              <concept-appearance id="1" x="73" y="56"/>
              <concept-appearance id="2" x="136" y="158"/>
       </concept-appearance-list>

       <linking-phrase-appearance-list>
              <linking-phrase-appearance id="3" x="104" y="107"/>
       </linking-phrase-appearance-list>
</map>
</cmap>
```

Figura 3.2: Mapa conceptual en formato CXL.

ser consultada en http://cmap.ihmc.us/xml/CXL.html.

Los elementos de importancia en el formato CXL para esta tesis son: los elementos **dublin core** y sus valores, así como también los valores de las etiquetas **label**. En la Figura 3.2, existen dos elementos D.C: *dc:title* y *dc:description* y los valores: *Plantas*, *Hojas* y *Tienen*.

3.1.2. Representación de Cmaps en otros formatos

Cmaps pueden ser exportado a otros formatos como pdf (formato de documento portable), imagen o página web, sin embargo, para intercambio de información con

otros programas, no solamente es CXL, a continuación se mencionan los otros dos más conocidos [2]:

- *XCM (XML Concept Maps).-* Este formato fue diseñado específicamente para los mapas conceptuales construidos con Cmap Tools, es un formato abierto para los desarrolladores. Cuando se exporta en este formato, se crea un archivo con la extensión XML y una carpeta con el mismo nombre que almacena los recursos (imágenes, URL's, otros Cmaps, etc.) que fueron referenciados en el Cmap.

- *XTM (XML Concept Maps).-* Es un formato estándar para el intercambio de mapas de tópicos y por lo tanto compatible con algunas otras aplicaciones. Los mapas de tópicos son similares a los mapas conceptuales. XTM incluye los enlaces y recursos anidados a los conceptos, no los estilos ni las imágenes. Una carpeta se crea con el mismo nombre del archivo XTM, que guarda los recursos anidados en el Cmap.

3.1.3. Procesamiento de colecciones CXL en Greenstone

Como se mencionó en el capítulo anterior, en la mayoría de las colecciones los datos descriptivos (llamados metadatos) como el autor, el título, la fecha, las palabras clave, etc., se asocian a cada documento y son la materia prima para la consulta. Los metadatos se pueden suministrar explícitamente o deben poder derivarse automáticamente de los propios documentos. En esta tesis se extraen de forma automática de un Cmap.

Para extraer los metadatos de forma automática, primero se convierten los documentos que integran una colección a formato CXL; una vez realizado lo anterior se analizan estos archivos por un *parseador* con el objetivo de extraer los metadatos (el parseador es un sistema que se describe en la siguiente sección). Después se asocian

estos metadatos mediante archivos *metadata.xml* a los documentos en el proceso de construcción de la colección, este proceso se realiza por Greenstone. Sin embargo, para que Greenstone pueda procesar archivos CXL se crea un plugin; la configuración se hace desde la interfaz bibliotecario Greenstone.

3.1.4. Metadata.xml

La forma de asignar metadatos automáticamente a las colecciones es a través de los archivos metadata.xml. Estos archivos se encuentran en la carpeta *import* de la colección en Greenstone.

La definición de tipo de documento (DTD) XML para estos archivos se muestra a continuación:

<¡DOCTYPE GreenstoneDirectoryMetadata[
<¡ELEMENT DirectoryMetadata (FileSet*)>
<¡ELEMENT FileSet (FileName+,Description)>
<¡ELEMENT FileName (#PCDATA)>
<¡ELEMENT Description (Metadata*)>
<¡ELEMENT Metadata (#PCDATA)>
<ATTLIST Metadata name CDATA #REQUIRED>
<ATTLIST Metadata mode (accumulate — override) override>
]>

3.1.5. Requerimientos funcionales y no funcionales del sistema

El parser es un sistema que extrae los metadatos de los archivos CXL y obtiene las *palabras clave* que se utilizan también para consulta. Las palabras clave son los valores de las etiquetas de nombre **label** descritas en la sección anterior.

A continuación se describen los requerimientos funcionales y no funcionales del sistema.

Requerimientos funcionales

Número de Requisito	R1
Nombre del Requisito	Leer colección

Introducción: El usuario selecciona la carpeta donde se encuentran los archivos CXL.
Salida: Si se ha escogido una carpeta regresará un valor verdadero, en caso contrario el valor será falso.

Número de Requisito	R2
Nombre del Requisito	Procesar colección

Introducción: El sistema procesará la colección (carpeta) que se seleccionó en R1. La colección puede tener uno o más archivos incluyendo subcarpetas. La lectura de los archivos se hace de forma recursiva.
Salida: Por cada archivo CXL se crea un archivo llamado *auxmetadata.xml*. El contenido de este archivo son los metadatos extraidos de cada CXL.

Número de Requisito	R3
Nombre del Requisito	Filtrar palabras clave de consulta

Introducción: Una vez terminado R2, el sistema filtra las palabras clave de consulta. Primero elimina palabras que se repitan y después elimina las palabras vacías (stop words). En esta versión del sistema, no se considera algún peso para las palabras no vacías.

Salida: Un vector con las palabras clave filtradas.

Número de Requisito	R4
Nombre del Requisito	Copiar archivo auxiliar a metadata.xml

Introducción: Después de realizar R3, se hace una copia del archivo `auxmetadata.xml` al archivo `metadata.xml`.

Salida: `metadata.xml` con la estructura definida en la sección 3.1 y que se asocia a los documentos de la colección creada en Greenstone.

Requerimientos no funcionales

Para la instalación y ejecución del parser es necesario lo siguiente:

- Tener instalado JRE 6.27 o superior.
- Ejecutar el archivo Parseador.jar.

3.2. Diagrama de casos de uso

En la Figura 3.3 se muestran los casos de uso del sistema:

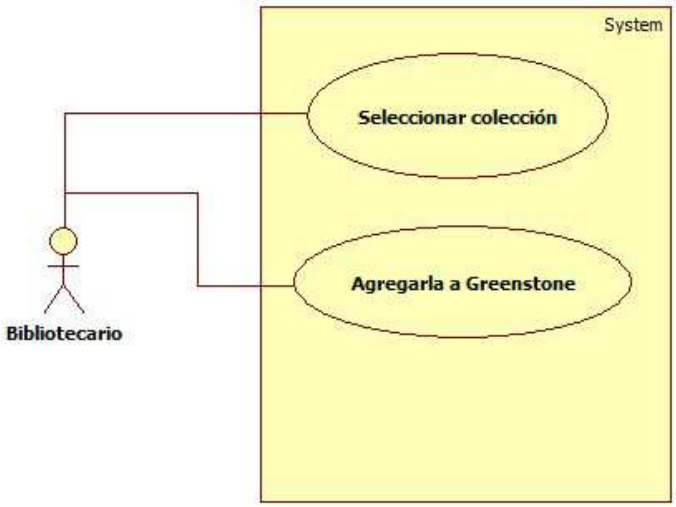

Figura 3.3: Diagrama de casos de uso.

Capítulo 4

Implementación

Este capítulo muestra el diagrama de clases del sistema así como la interfaz del parser y su funcionamiento con Greenstone.

4.0.1. Diagrama de clases

La Figura 4.1 muestra el diagrama de clases del sistema, el cual está compuesto de 6 clases principales. A continuación se describe el funcionamiento de cada una.

- *ParseadorGView:* Clase principal, muestra la interfaz gráfica. Sus principales métodos son:

  - *escogerColeccion:* Permite seleccionar la carpeta donde se encuentran los documentos de la colección.
  - *leerColeccion:* Lee los archivos de la colección de forma recursiva e identifica cuáles son directorios y cuáles son los archivos CXL.
  - *crearMetadata:* Este método obtiene los elementos Dublin Core del archivo CXL, así como los valores de las etiquetas **label**, que se conocen también como *clave*.

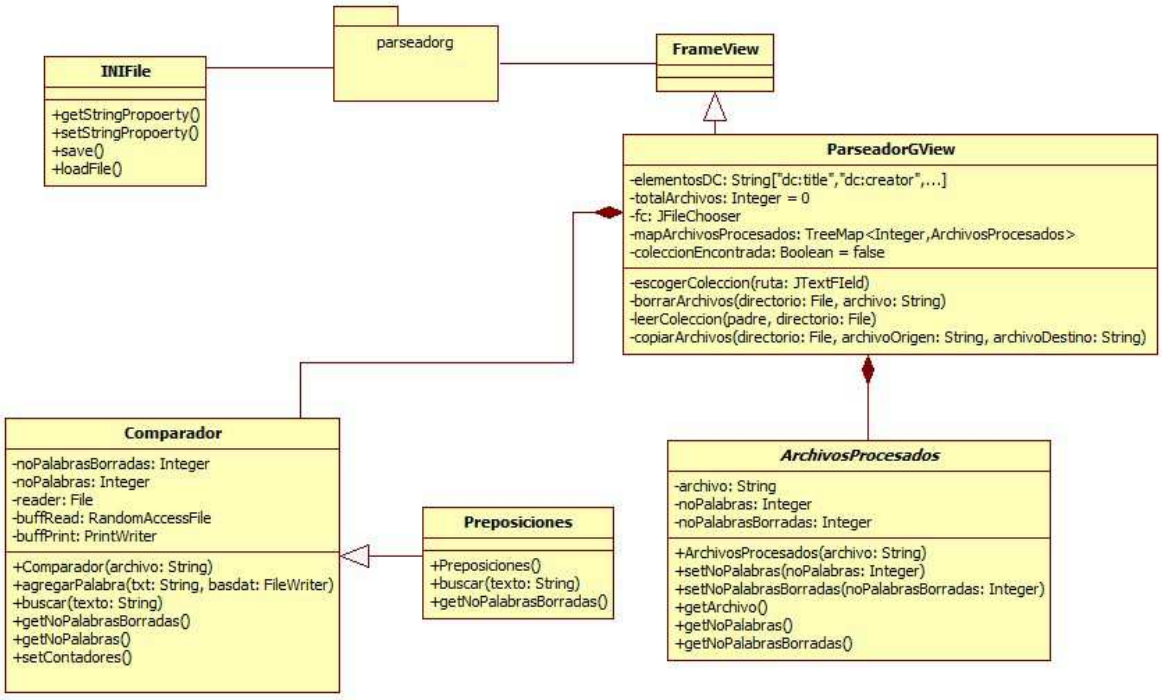

Figura 4.1: Diagrama de clases del parser

- *filtrarEtiquetas:* Elimina las palabras repetidas así como las palabras sin significado como artículos, pronombres, preposiciones, etc. a este tipo de palabras se les llama *palabras vacías* .

- *ArchivosProcesados:* Esta clase almacena los nombres de los archivos que fueron procesados, el número de palabras clave que se obtuvieron de cada uno de ellos y el total de palabras clave que fueron eliminadas por repetición.

- *Comparador:* es la clase que implementa los métodos de lectura y escritura de los archivos txt que contienen las palabras vacías.

- *INIFile:* Clase que tiene los métodos para abrir y obtener los valores de un archivo de configuracin.INI. El archivo de configuración, es un archivo de texto plano que almacena valores para ser recuperados por la clase principal.

4.0.2. Interfaz de usuario

Esta sección muestra el uso del sistema para el usuario o bibliotecario. En la Figura 4.2 se observa la pantalla principal:

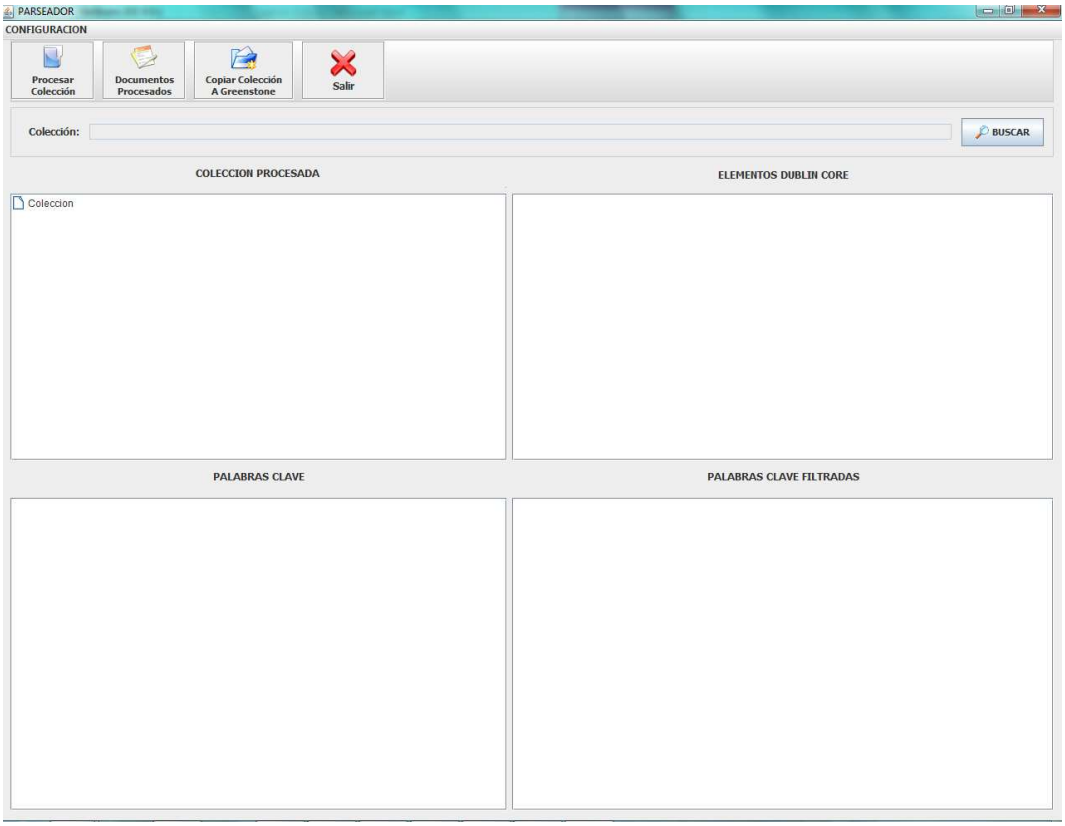

Figura 4.2: Interfaz de usuario

En la Figura 4.3 se muestra el menú *CONFIGURACIÓN:*

El paso 1 de la figura 4.3 se configura cada vez que se instale el sistema. Si se selecciona esta opción se abre un archivo de configuración llamado config.ini donde se especifican las rutas de las siguientes carpetas:

- *Dirtemporal:* Es la carpeta auxiliar que se utiliza en el proceso de la construcción de la colección. Almacena los documentos procesados con su respectivo archivo metadata.xml. Esta carpeta no guarda ningún archivo CXL. Por defecto es:

Figura 4.3: Menú de configuración

c:\temporal.

- *Dircoleccionbase:* Es la carpeta con los archivos básicos para la construcción de una colección en Greenstone. Las subcarpetas principales son las siguientes: `etc, images, import, log, macros, metadata, style y script`. Por defecto es: c:\Coleccion-base.

- *Dircollect:* Es la carpeta que utiliza Greenstone para alojar las colecciones que han sido o serán procesadas. Por defecto se encuentra ubicada en el directorio de instalación, el se encuentra ubicado en la carpeta del usuario en el sistema operativo Windows.

El paso 2 y 3 de la figura 4.3 abre un archivo de texto plano con extención txt, que contiene una lista de palabras vacías o sin significado para la búsqueda. Este listado de palabras está en Español y en Inglés, se les han atribuido el nombre de *stop words*. Pueden agregarse nuevas palabras a estos listados.

El paso 4 y 5 de la figura 4.3 requieren configurarse cada vez que se procese una colección, a diferencia de los anteriores pasos. El paso 4 se selecciona el idioma de los documentos de la colección: inglés o español. Por defecto se encuentra seleccionado

Idioma Español. En el paso 5 se escoge el formato de archivo en que se encuentran los documentos: JPG o PDF. Por defecto se encuentra seleccionado JPG.

Después de que se han realizado los pasos anteriores, se busca la colección que se va a procesar. Si se escoge la opción BUSCAR, se abre una caja de diálogo en donde se puede navegar por el sistema de archivos hasta encontrar la carpeta de la colección. Esta carpeta debe tener los archivos CXL así como su correspondiente formato JPG o PDF. En la Figura 4.4 se muestra esta caja de diálogo:

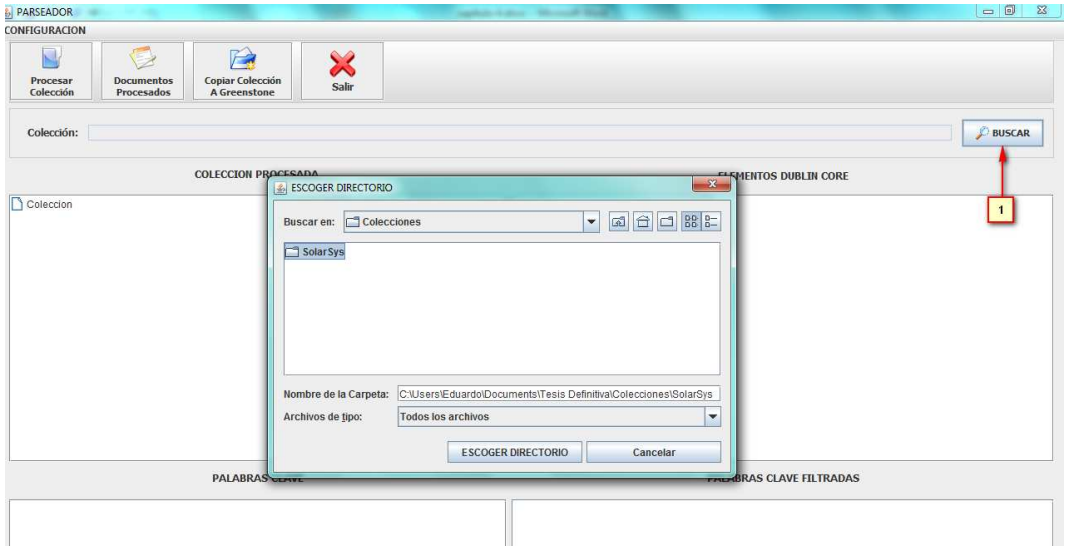

Figura 4.4: Caja de diálogo para seleccionar colección a procesar

Después de haber seleccionado la carpeta de la colección, se muestra su ubicación (Ver la Figura 4.4). Para iniciar el proceso de parseo de la colección seleccionada se emplea el botón: **Procesar Colección** como se muestra en el paso de 2 de la figura 4.5:

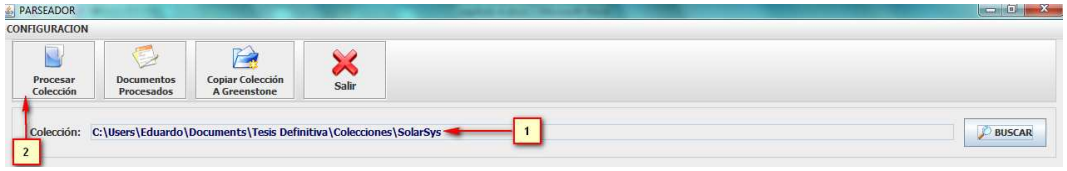

Figura 4.5: Colección seleccionada para iniciar proceso de parseo

Una vez que se ha terminado de parsear la colección se muestra un cuadro de diálogo informando que la colección ha sido procesada (Ver la Figura 4.6).

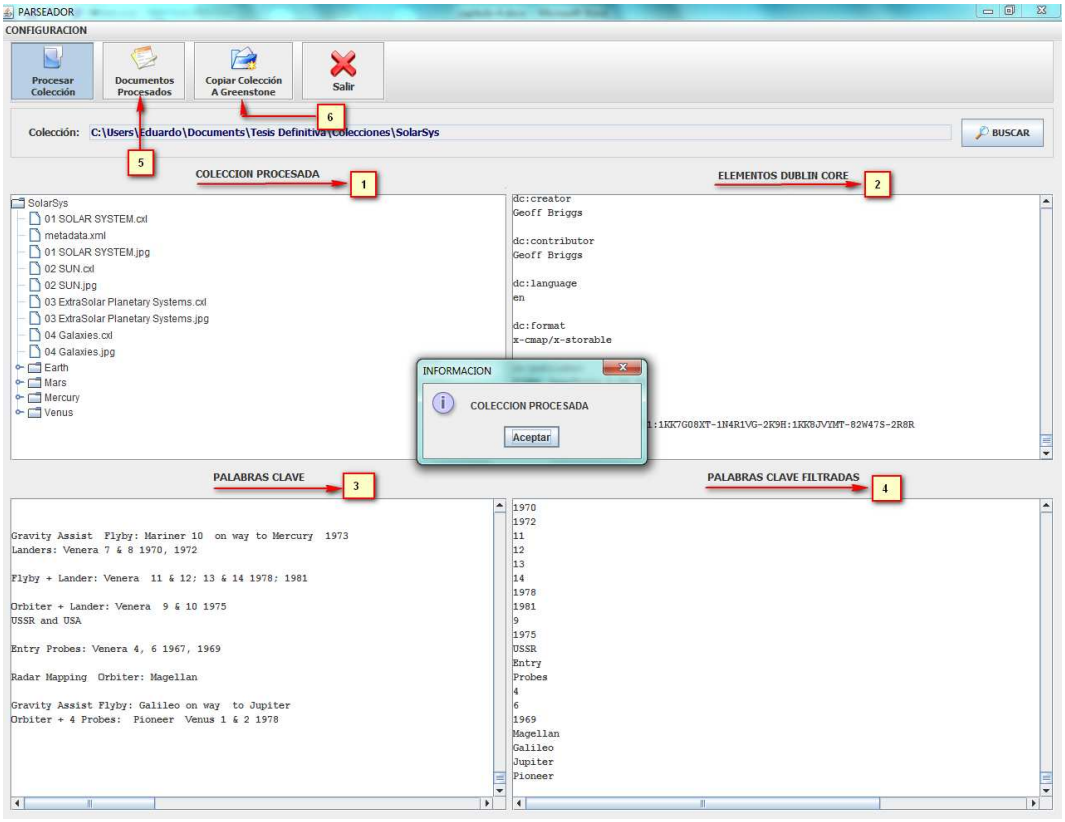

Figura 4.6: Colección procesada de forma correcta

En el Paso 1 de la Figura 4.6 se presenta los archivos de la colección que han sido procesados así como su respectivo archivo metadata.xml que ha sido agregado a cada carpeta de la colección. Si se requiere abrir un archivo se selecciona y con el botón derecho del ratón aparece un cuadro emergente con esta opción.

En el paso 2 se presentan los elementos Dublin Core con sus respectivos valores que han sido extraidos de los archivos CXL. En la parte superior se despliega el archivo que fue procesado y en la parte inferior los elementos extraidos.

En el paso 3 se presentan las palabras clave que fueron extraidas de los archivos CXL. Las palabras clave son los valores que se encuentran en las etiquetas **label** en los

archivos CXL.

En el paso 4 se encuentra un filtrado de las palabras clave que se obtuvieron en el paso 3. En este listado se han eliminado palabras repetidas y las palabras vacías. El objetivo de realizar un filtrado es quedarse con las palabras clave más significativas para una búsqueda. Las palabras que han sido filtradas son almacenadas en los archivos metadata.xml que fueron creados cuando se procesó la colección.

En el paso 5 se presenta un listado de los archivos procesados así como el número de palabras clave que fueron analizadas y el número de palabras que fueron eliminadas como consecuencia del proceso de filtrado de palabras.

En el paso 6 se copian los archivos de la colección procesada con las respectivas carpetas y archivos de configuración que necesita Greenstone para poder abrirla desde su interfaz. La copia se hace en la carpeta **Collect**.

Para abrir la colección que se procesó desde la interfaz de Greenstone, se escoge el menú Archivo y se muestra una ventana como la de la Figura 4.7:

El nombre de la colección es el mismo que el de la carpeta seleccionada en el paso 1. Una vez abierta la colección, se muestra la interfaz en la Figura 4.8:

Se pueden observar los metadatos asignados a los archivos en la pestaña Enriquecer como se muestra en la Figura 4.9. Se enfatiza en la imagen el elemento **Dublin Core: dc:claves**, el cual su valor son las palabras claves filtradas.

Para crear la colección, se escoge la pestaña **Crear**, como se muestra en la Figura 4.10 y después se selecciona el botón **crear colección**.

Una vez realizada la creación de la colección, se muestra una pantalla como la de la Figura 4.12. Además se despliega la información de los archivos procesados.

Una vez terminado el proceso, se escoge el botón **Vista previa** de la colección.

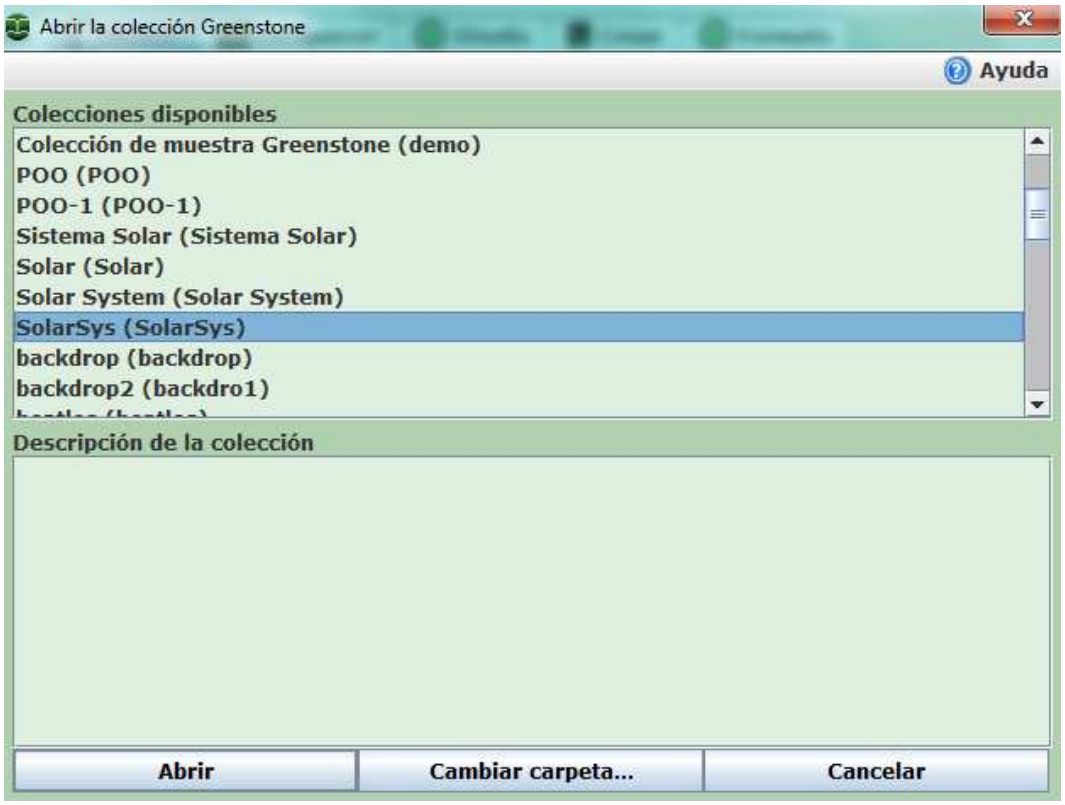

Figura 4.7: Abrir colección en Greenstone

Figura 4.8: Interfaz Greenstone, pestaña Reunir

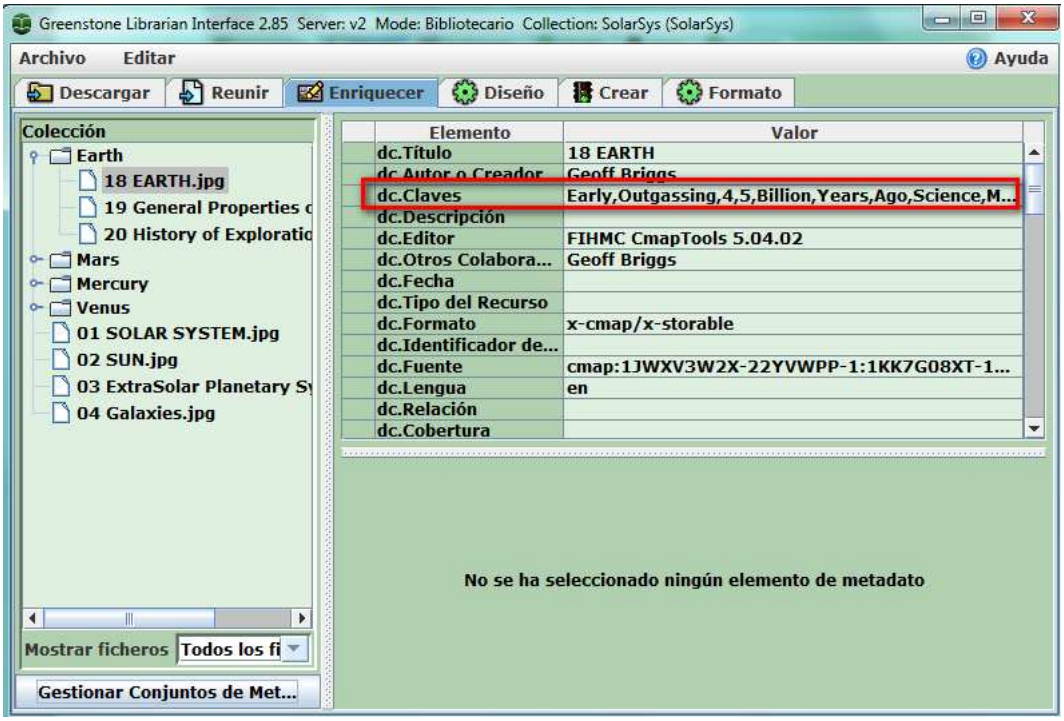

Figura 4.9: Metadatos obtenidos de los archivos metadata.xml

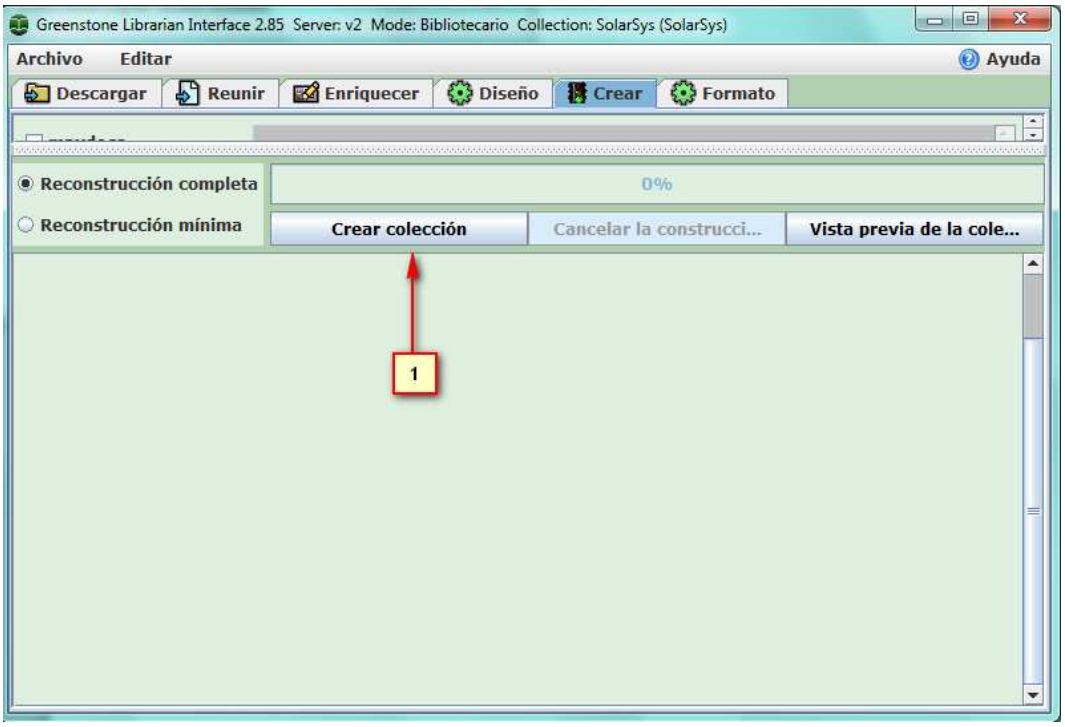

Figura 4.10: Crear colección, con la opción de reconstrucción completa

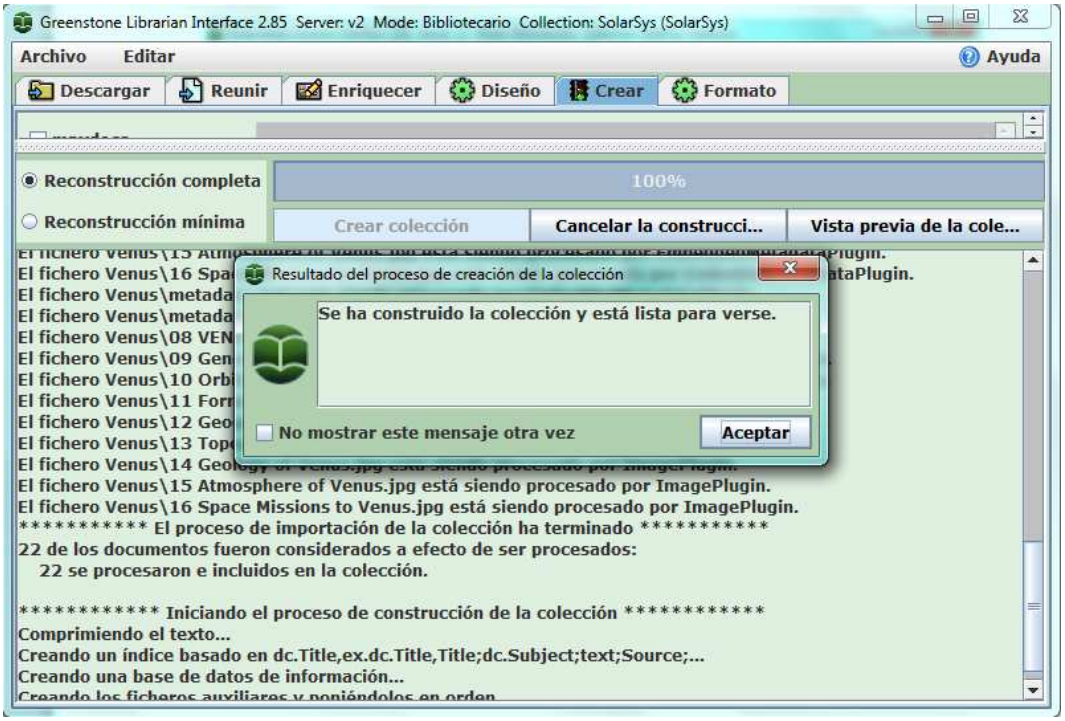

Figura 4.11: colección terminada de forma correcta

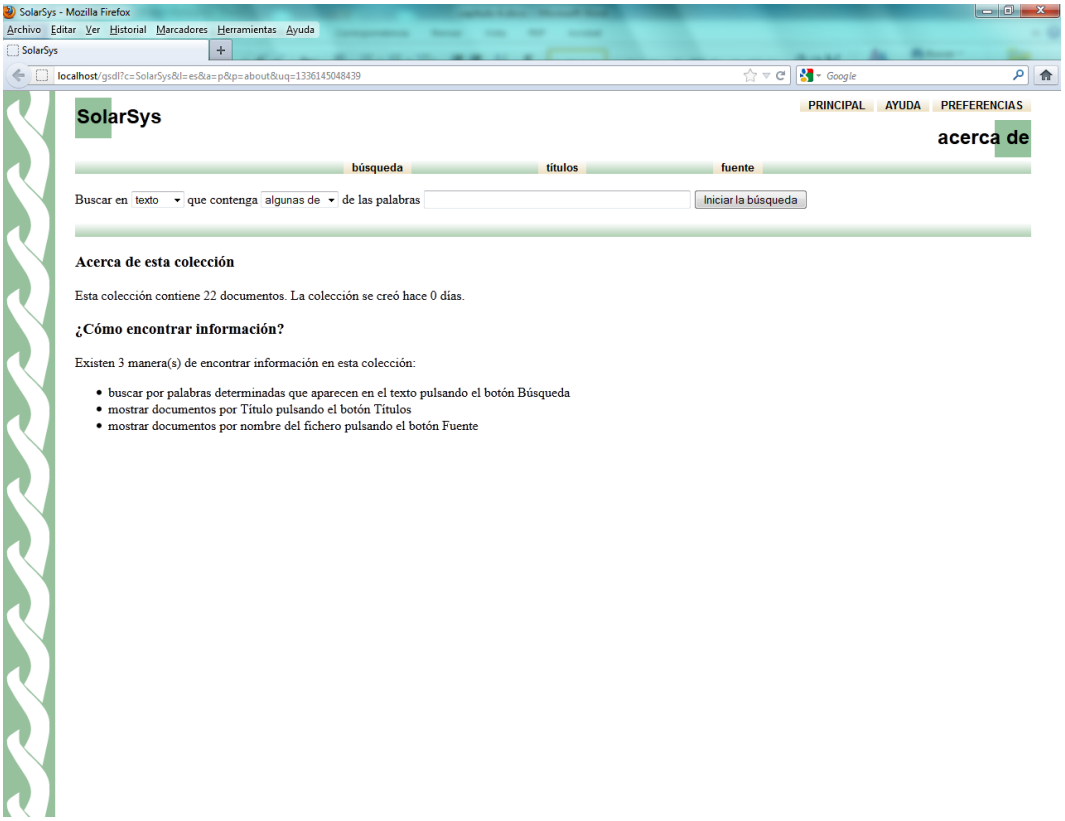

Figura 4.12: Interfaz web de la colección procesada por Greenstone

Capítulo 5

Resultados

Se descargaron diferentes colecciones de mapas conceptuales de diferentes autores e instituciones, que se utilizaron para hacer pruebas con el sistema para crear su colección correspondiente. A continuación se describen los tres servidores principales donde se extrajeron estos mapas:

IHMC MAST Project: Es un conjunto de mapas conceptuales que son usados para expresar conocimiento sobre el dominio de seguridad en las redes de cómputo. El proyecto MAST es patrocinado por la Fundación Nacional de Ciencia (NSF). Los mapas pueden ser descargados desde la siguiente url: mast.ihmc.us/cmap.php.

IHMC Space Exploration: Son varios grupos de colecciones conceptuales relacionados con la exploración del Universo, desarrollados por la NASA. Sólo se descargaron mapas relacionados con el sistema solar, debido al gran número de mapas que tiene este sitio. Puede ser consultada esta información desde la siguiente url: http:
cmapspaceexp.ihmc.us.

IHMC Public Cmaps: Es un sitio de dominio público, donde existe información de diferentes campos del conocimiento. En este sitio se descargaron mapas conceptuales relacionados con el tema programación orientada a objetos de distintos autores. Puede ser consultada la información desde la siguiente url: http:

cmapspublic.ihmc.us.

Se tomaron algunas palabras que tenían en común el conjunto de mapas conceptuales de las colecciones que se mencionaron anteriormente y que iban a ser utilizadas como palabras de búsqueda desde la biblioteca digital semántica. Por lo tanto la prueba que se realizó fue saber si el número de documentos que regresaba la biblioteca digital a través de una palabra de búsqueda correspondía al número de documentos que tenían esta palabra, debido a que primero se contabilizaron los documentos de forma manual.

En la Tabla 4.1 se muestra el número de documentos que se recuperaron de cada coleccion con respecto a la palabra de búsqueda que se utilizó en la biblioteca digital y que se encuentra del lado izquierdo del número.

Tabla 5.1: Resultados de busqueda de las palabras clave en las colecciones

MAST	Resultado	**Space**	Resultado	**POO**	Resultado
Hosts	7	**hydrogen**	7	**paquete**	2
internet	2	**water**	7	**clase**	7
spoofing	1	**satellite**	8	**herencia**	8
overflow	1	**diameter**	7	**encapsulamiento**	4
attack	8	**temperature**	8	**polimorfismo**	5
password	2	**moon**	9	**abstraccion**	6

La colección *MAST* tiene un total de 8 documentos, la colección *Space* tiene un total de 18 documentos y la colección *POO* tiene un total de 12.

Se comprobó que el número de documentos recuperados por cada palabra de búsqueda es igual al número de documentos que se tenía previsto ya que este proceso de búsqueda se hizo primero de forma manual.

Se procesó correctamente todos los archivos de cada una de las colecciones con el plugin *CXL*. A continuación se presentan los resultados obtenidos.

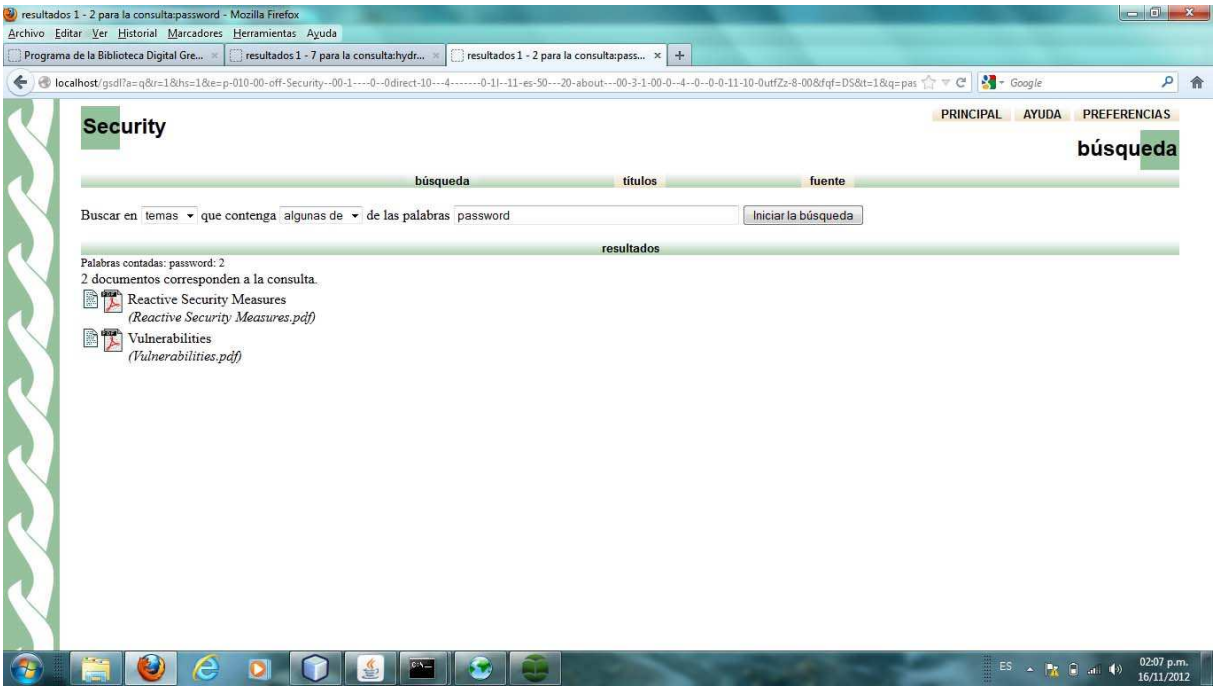

Figura 5.1: Resultado de la consulta con la palabra *password* de la colección *MAST Project*. Los documentos obtenidos se encuentran en formato PDF.

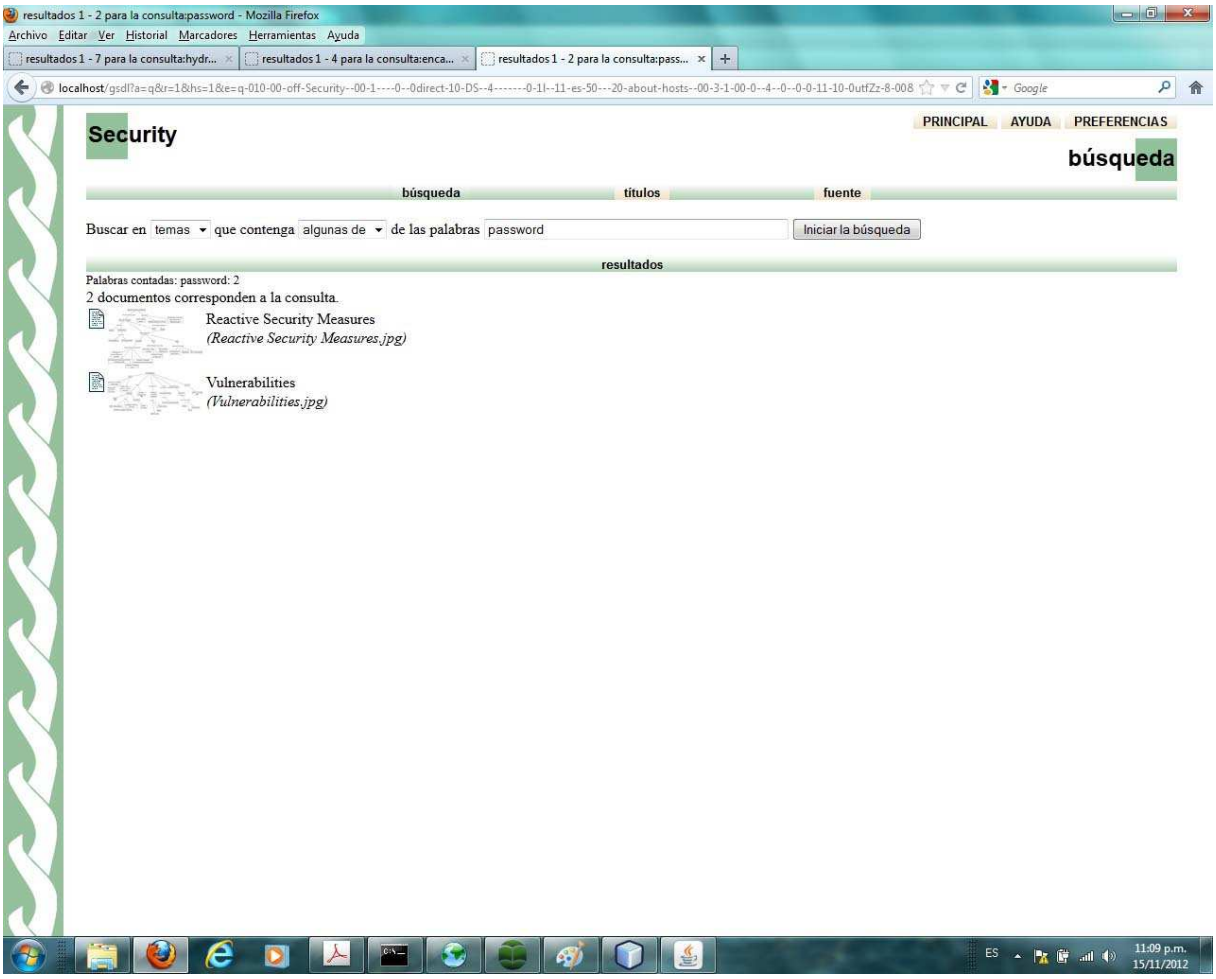

Figura 5.2: Resultado de la consulta con la palabra *password* de la colección *MAST Project*. Los documentos obtenidos se encuentran en formato JPG.

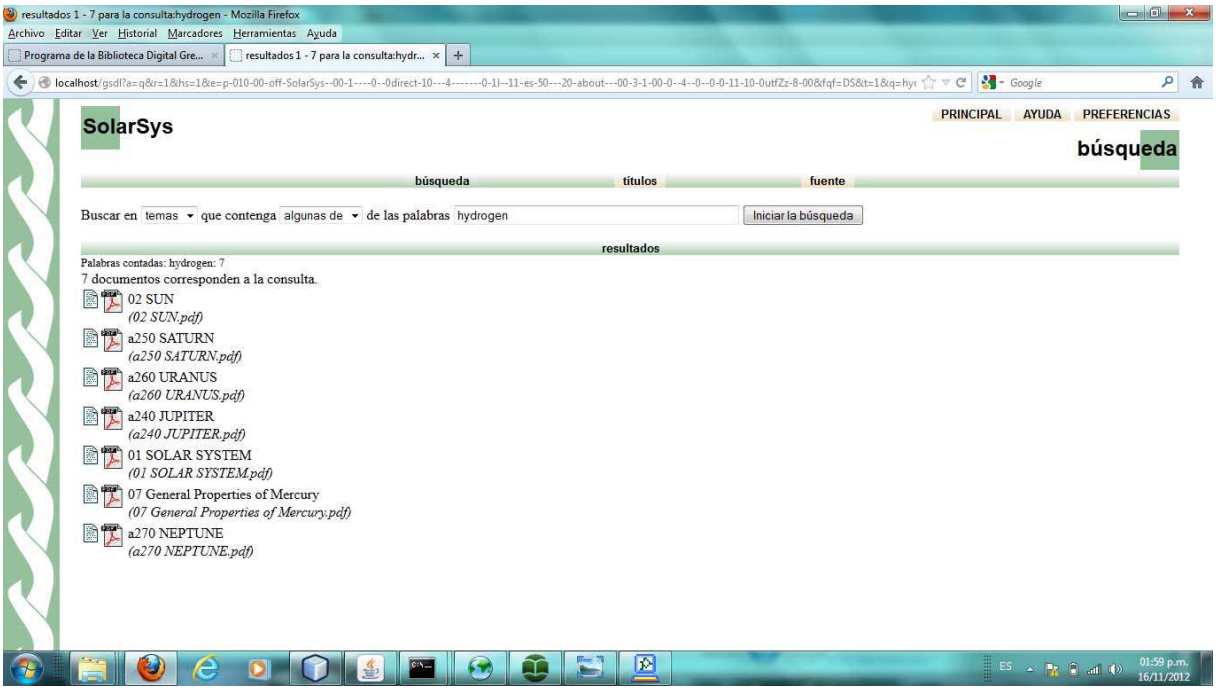

Figura 5.3: Resultado de la consulta con la palabra *hydrogen* de la colección *IHMC Space Exploration*. Los documentos obtenidos se encuentran en formato PDF.

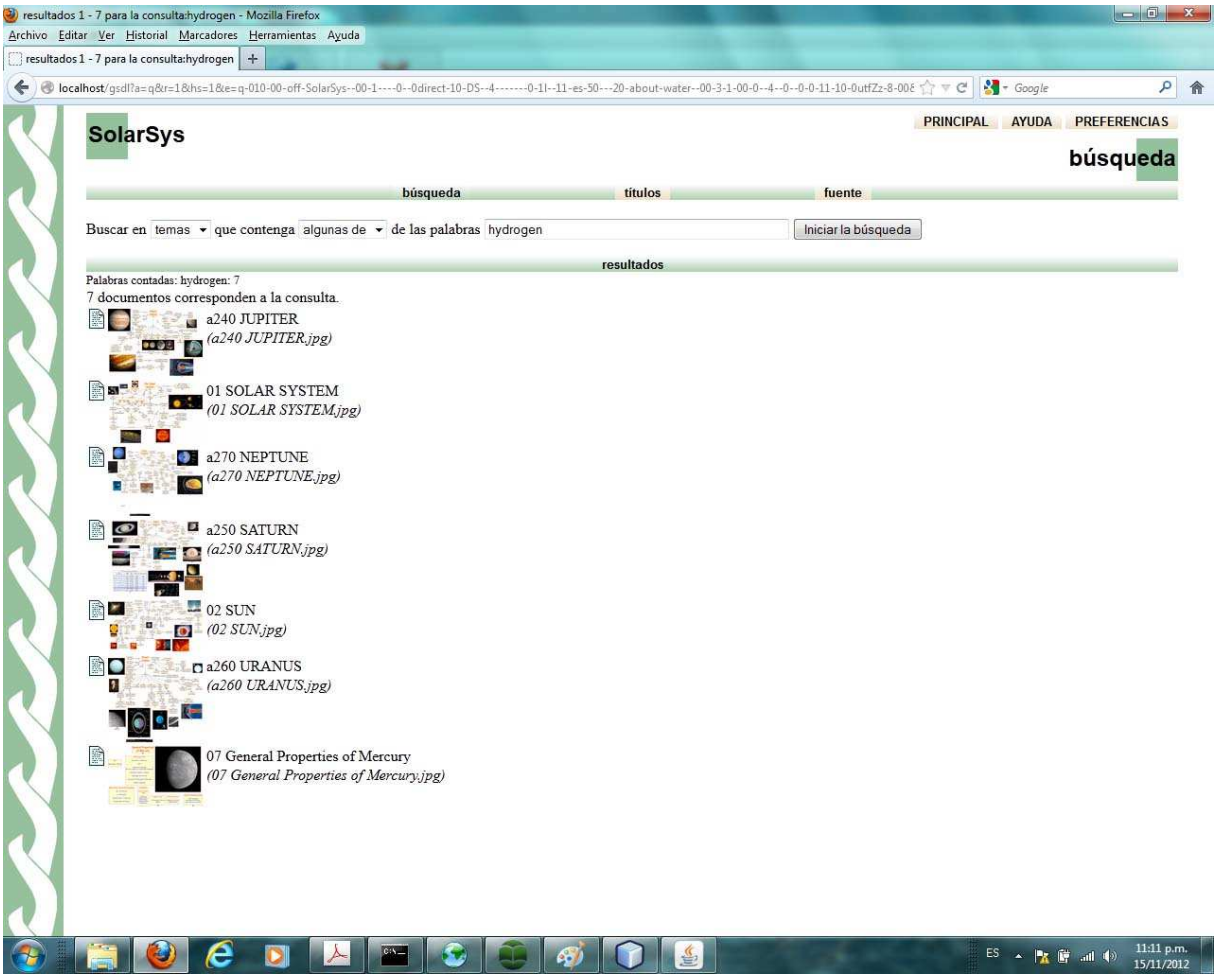

Figura 5.4: Resultado de la consulta con la palabra *hydrogen* de la colección *IHMC Space Exploration*. Los documentos obtenidos se encuentran en formato JPG.

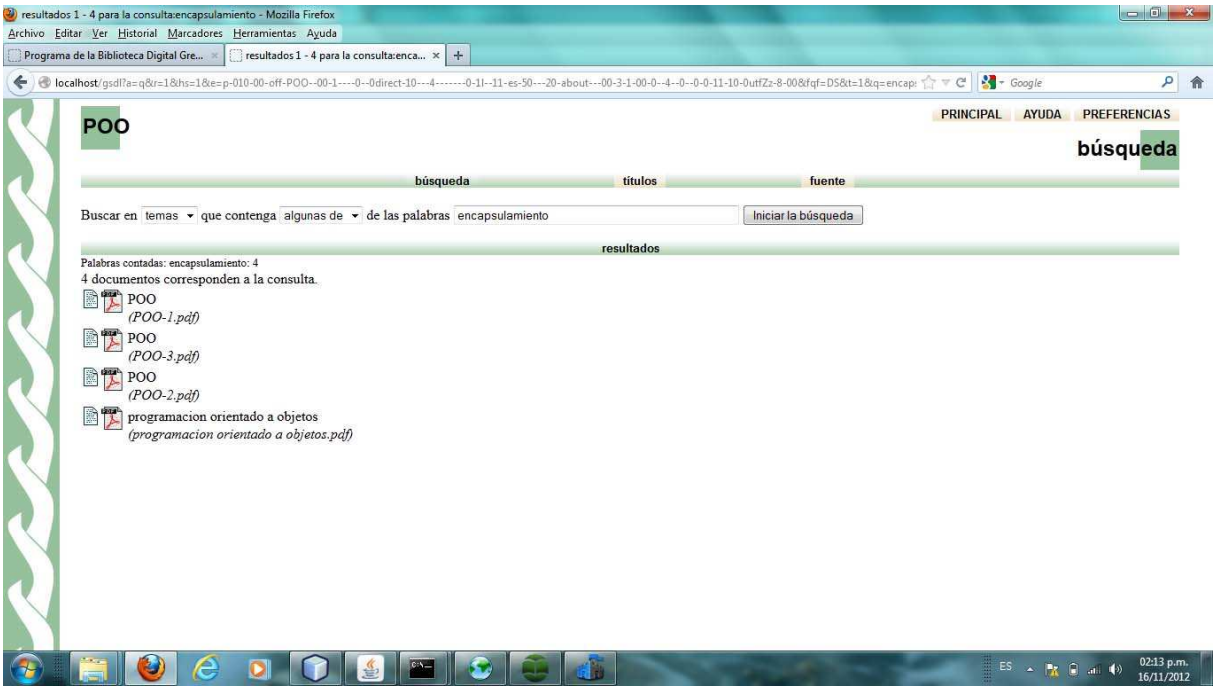

Figura 5.5: Resultado de la consulta con la palabra *encapsulamiento* de la colección *Programación*. Los documentos obtenidos se encuentran en formato PDF.

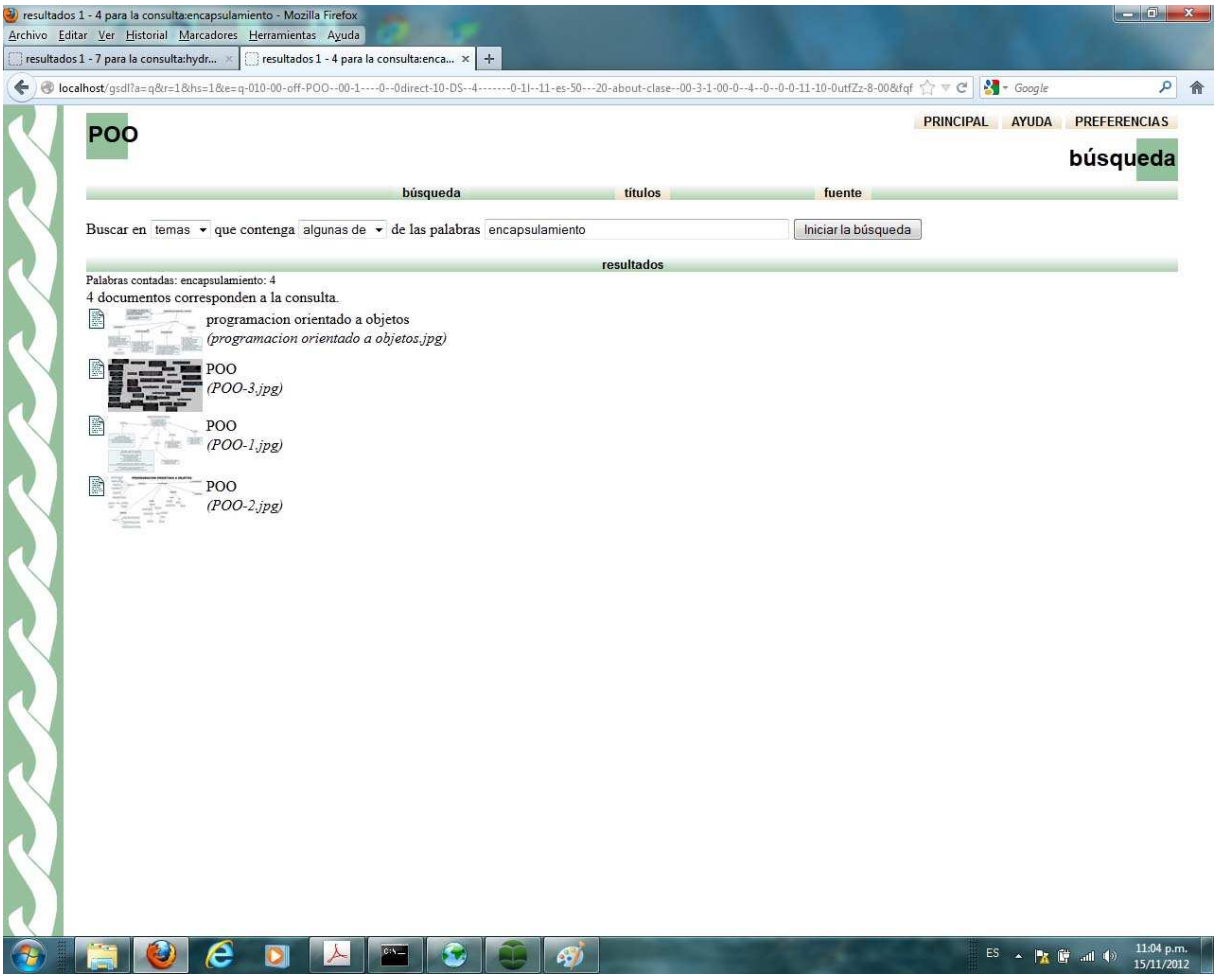

Figura 5.6: Resultado de la consulta con la palabra *encapsulamiento* de la colección *Programación*. Los documentos obtenidos se encuentran en formato JPG.

Capítulo 6

Conclusiones

En este proyecto de tesis se investigó la forma en cómo se comunicarían dos sistemas distintos creados por diferentes instituciones: Greenstone y Cmap Tools, ambos sistemas de distribución libre. El primero integra un conjunto de herramientas para usuarios finales y bibliotecarios mientras que el segundo utiliza una arquitectura de intercambio de conocimiento con la finalidad de que exista una forma de interoperar con otros programas y poder compartir así mapas conceptuales.

Lo anterior condujo a que se creara un medio que hiciera la función de interconexión entre ambos sistemas, es decir, se propuso el desarrollo de un parser para poder interoperar. Este parser tiene como entrada una colección de documentos y como salida una carpeta de directorios y archivos, preparada para ser procesada por Greenstone y construir una biblioteca digital.

Por lo tanto, se puede concluir que los objetivos de este proyecto fueron alcanzados y que se realizaron de forma satisfactoria. Además, cabe resaltar que a la fecha, éste ha sido el primer trabajo de interoperabilidad entre Cmap Tools con Greenstone. Esto tiene como potencial la construcción de bibliotecas digitales, en donde la información de salida sea mapas conceptuales en formato de imagen o pdf a partir de una búsqueda realizada en texto.

Esto conlleva a que en una biblioteca digital semántica creada por estas dos herramientas, se obtenga como salida mapas conceptuales en formato de imagen a partir de una búsqueda realizada en texto.

El uso de las colecciones de mapas conceptuales en bibliotecas digitales se espera sea mayor en ambientes principalmente educativos, debido a la naturaleza de los mapas conceptuales, sin embargo, pueden ser también de utlidad en otros ámbitos.

Como trabajo a futuro, se planea lo siguiente.

El proceso de convertir los mapas conceptuales a formato CXL es una tarea que consume tiempo, si es un gran número de mapas el tiempo de conversión será considerable, ya que se tiene que hacer archivo por archivo. Se podría eficientar este proceso descargando los documentos y realizar la conversión a formato CXL directamente de los servidores Cmaps a través de servicios web sin tener que descargarlos y covertirlos individualmente.

Otro punto a considerar es que el sistema Parser sólo asocia formatos de archivos PDF o JPG a los documentos devueltos por una consulta hecha a la biblioteca digital, se podrían asociar más formatos a los archivos de salida como por ejemplo documentos en html o word.

Bibliografía

[1] Chandrasekaran B. and Josephson J. R. What are ontologies, and why do we need them? *IOS Press, Amsterdam*, pages 21–23, January/February 1999.

[2] Eds. Caas A. J., Novak J. D. Kea: A knowledge exchange architecture based on web services, concept maps and cmaptools. *www.ihmc.us*, page 7, 2006.

[3] The World Wide Web Consortium. http://www.w3.org/standards/semanticweb/, 2012 Copyright 2012, W3C.

[4] Obrst L. J. Daconta M. C. and Smith K. T. *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management.* 2003.

[5] DSpace. http://www.dspace.org, November 2002. Copyright Creative Commons Corporation.

[6] ihmc. http://http://cmap.ihmc.us/, 2012. Copyright 1998-2006, Institute for Human and Machine Cognition.

[7] Dublin Core Metadata Initiative. http://dublincore.org/documents/dces/, October 2010. Copyrigth 1995-2012, DCMI.

[8] JeromeDL. http://www.jeromedl.org/, 2008. Copyright 2003-2008, National University of Ireland, Galway, Ireland.

[9] Gowin BOB D. Joseph D., Novak. *Aprendiendo a aprender*, volume 1. Ediciones Martínez Roca, 1998.

[10] Berners L., Hendler J., and Lassila O. The semantic web. *Scientific American*, 1:180–210, 2001.

[11] Gruber T. R. A translation approach to portable ontology specifications. *Technical Report KLS 92-71*, pages 1–4, April 1993.

[12] Bray T. Rdf and metadata. *http://www.xml.com/pub/a/2001/01/24/rdf.html*, 1998. Copyright 2010, O'Reilly Media, Inc.