

Diseño y desarrollo de un prototipo para la integración y visualización de recursos web semánticos mediante grafos conceptuales

Design and development of a prototype for integration and visualization of semantic web resources through conceptual graphs

ANDRÉS FELIPE POVEDA

Ingeniero de Sistemas, Investigador de la Universidad Distrital Francisco José de Caldas. Bogotá, Colombia. Contacto: felipe07@gmail.com

JOSÉ ANTONIO MOREIRO GONZÁLEZ

Licenciado en Filosofía y Letras, Doctor en Historia. Docente de la Universidad Carlos III de Madrid. Madrid, España. Contacto: jamore@bib.uc3m.es

JULIO BARÓN VELANDIA

Ingeniero de Sistemas, Especialista en Ingeniería de Software. Docente de la Universidad Distrital Francisco José de Caldas. Bogotá, Colombia. Contacto: jbaron@udistrital.edu.co

OSWALDO ALBERTO ROMERO VILLALOBOS

Ingeniero de Sistemas, Especialista en Ingeniería de Software. Docente de la Universidad Distrital Francisco José de Caldas. Bogotá, Colombia. Contacto: oromero@udistrital.edu.co

Fecha de recepción: 6 de septiembre de 2011

Clasificación del artículo: Investigación

Fecha de aceptación: 16 de octubre de 2012

Financiamiento: Universidad Distrital Francisco José de Caldas

Palabras clave: grafo, recurso web, Web semántica.

Key words: graph, web resource, semantic web.

RESUMEN

En este artículo se describe el diseño de un prototipo web que permite realizar consultas a repositorios de triplas semánticas. Dichas triplas al ser obtenidas pueden ser vinculadas en un documen-

to Resource Description Framework (RDF) generado por la aplicación. Este documento puede ser representado como un grafo conceptual dirigido en el navegador web, representación que puede ser ampliada con resultados de consultas posteriores realizadas a través de la aplicación.

El principal objetivo del prototipo es servir como gestor de búsqueda y agrupamiento de artículos académicos relacionados con el área de las ciencias de la computación y la informática. Este prototipo está conformado por tres componentes que cumplen las funciones de consulta, integración y visualización de recursos web extraídos de repositorios semánticos mediante el lenguaje estándar Protocol and RDF Query Language (SPARQL).

Tras la culminación del proyecto se obtuvieron los siguientes resultados: se realizó el proceso de consulta de repositorios de recursos web semánticos mediante el diseño y desarrollo de un mecanismo basado en el uso de consultas SPARQL, el lanzamiento de peticiones asincrónicas y la técnica cross-site request. El proceso de integración de recursos web fue resuelto mediante el uso del framework basado en Java para desarrollo de aplicaciones semánticas Jena. La visualización de recursos web semánticos se logró por medio de la librería RAP (RDF API for PHP), la cual genera grafos conceptuales a partir de un documento Resource Description Framework / Extensible Markup Language (RDF/XML) correctamente estructurado. La metodología relacionada con la construcción del prototipo siguió el modelo de proceso para desarrollo de software OpenUP.

ABSTRACT

This paper describes the design of a web-based prototype that allows execution of queries on se-

mantic triple repositories. Once obtained, triple repositories may be linked to an RDF document generated by the application. This document can be represented as a direct conceptual graph on the web browser. Such representation can be extended with further query results performed by means of the application.

The main objective of the prototype is to serve as a manager for searching and grouping academic papers connected with the fields of Computer Science and Information Technology. This prototype consists of three components that deal with query, integration and visualization functions, respectively. Web resources are extracted from semantic repositories using standard language SPARQL.

The main results can be summarized as follows: the query process of semantic web repositories was completed through the design and development of a mechanism based on the use of SPARQL queries, the launch of asynchronous request and a technique called cross-site request. The integration process of web resources was solved using Jena, the java framework for building semantic web applications. Visualization of web resources was achieved using library RAP (RDF API for PHP), which generates conceptual graphs from a well formed RDF/XML document. The methodology related to the construction of the prototype followed OpenUP as the software development process model.

* * *

1. INTRODUCCIÓN

La publicación del artículo “Design Issues: Linked Data” [1] ha generado un interés creciente por el estudio de propuestas para publicación y enlace de datos estructurados en la Web. Este interés se debe principalmente a la posibilidad que

brinda la técnica descrita por Berners-Lee para compartir información en la World Wide Web y establecer relaciones que unan diferentes recursos de tal modo que su exploración haga explícita la información y sus conexiones. Para Berners-Lee la ganancia fundamental obtenida tras publicar datos mediante esta técnica puede resumirse en

que el valor y utilidad de los datos se incrementa en la medida en que están enlazados con otros [1].

Centros de investigación y organizaciones gubernamentales de todo el mundo, tomando la iniciativa, han implementado repositorios disponibles públicamente con resultados tales como el proyecto Linking Open Data [2]. Estos esfuerzos han sido orientados hacia el mejoramiento de las técnicas y tecnologías usadas para exponer datos enlazados en la Web, así como al fomento de su uso en el ámbito científico y gubernamental. Entre los resultados más representativos que se han obtenido se encuentra la nube de repositorios del proyecto Linking Open Data [3], que contiene alrededor de 121 repositorios de datos, albergando en su interior desde 121 triplas hasta la considerable cantidad de 6400 000 000 triplas. El total de triplas de los 121 repositorios contabilizados hasta el 4 de mayo de 2010 es de 13 112 409 691.

Esta enorme y variada cantidad de triplas representa información sobre un amplio conjunto de aspectos del mundo, el conocimiento, la ciencia, la economía e incluso las personas, por consiguiente acceder a esta información supone una ventaja y un beneficio para cualquier organización o individuo.

En el marco de los aspectos descritos anteriormente nació el presente proyecto mediante el cual se definió un conjunto de lineamientos y pautas que permitieron concretar un plan de trabajo para la construcción de un prototipo con el cual se logró sacar provecho de las ventajas inherentes de acceder a recursos semánticos y efectuar operaciones de integración y exploración de los mismos.

2. FUNDAMENTOS

2.1 Web semántica

La necesidad actual de optimizar y utilizar eficazmente la Web para mejorar los sistemas de búsqueda, recolección de datos y velocidad en los

resultados, ha generado una nueva perspectiva: la Web semántica. El objetivo general de la Web semántica es enriquecer con conceptos, funciones y axiomas lógicos la Web actual, mediante metadatos que puedan ser entendibles y procesables por dispositivos o componentes computacionales.

De este modo, según [4] la Web semántica será el escenario para la confluencia de las técnicas de inteligencia artificial, proporcionando mecanismos de declaración de axiomas junto con los métodos de razonamiento basados en estos, así como las tecnologías Web que permitan dotar a los recursos disponibles en Web de la semántica requerida para que las máquinas realicen los procesos y actividades operativas que actualmente deben efectuar los seres humanos.

Esta definición enfatiza la importancia de brindar significado a los datos publicados en la Web mediante el uso de metadatos semánticos y ontológicos que permitan a los agentes inteligentes realizar el procesamiento y ejecución de determinadas labores repetitivas y extenuantes que los seres humanos realizan actualmente. De este modo, los enlaces entre los documentos existentes en la Web pasarían de ser hipervínculos especificados en lenguaje HTML (HyperText Markup Language) a relaciones establecidas mediante lenguajes semánticos en los cuales cada enlace indicaría claramente el significado de la relación a través de etiquetas procesables por las máquinas.

La figura 1 ilustra la diferencia en cuanto al modo de relacionar documentos en la Web actual, utilizando etiquetas HREF del lenguaje HTML y en la Web semántica, utilizando etiquetas semánticas.

Aunque uno de los principales objetivos que busca alcanzar la Web semántica es la capacidad de operar sobre los datos mediante procesos de inteligencia artificial, existen otros temas centrales que hacen parte de la propuesta establecida inicialmente por Berners-Lee, cuya implementación

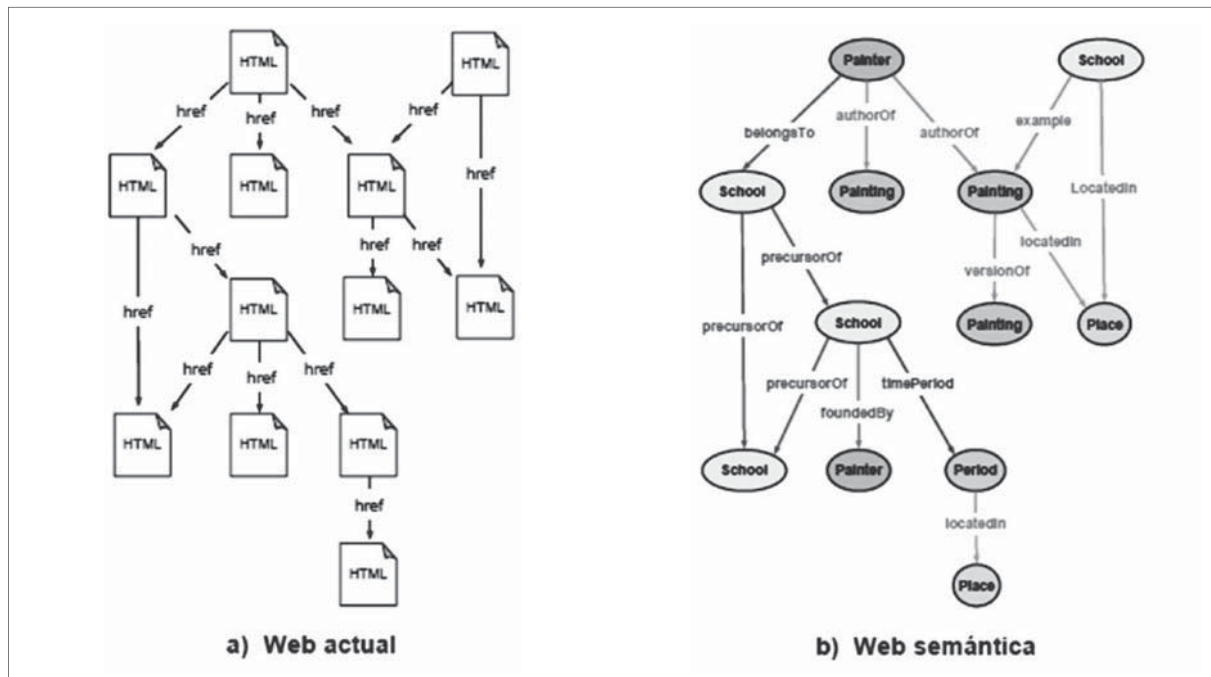


Figura 1. Web actual vs. Web semántica
Fuente: tomada de [4].

impactaría profundamente el funcionamiento y desempeño de la red mundial de información. Estos puntos son:

- La publicación libre por parte de organizaciones e individuos de datos usando un estándar abierto.
- El uso por parte de las compañías en sus procesos de negocio de los datos disponibles publicados de acuerdo al punto anterior.

Lograr concretar los puntos anteriores implicaría entender a la Web semántica como una base de datos global, disponible para organizaciones e individuos, facilitando el acceso libre a información que antes se encontraba encerrada en bases de datos separadas, heterogéneas y carentes de interoperabilidad a nivel semántico.

Lograr una Web semántica en la cual los datos puedan ser publicados utilizando un estándar

abierto, permitiendo que agentes inteligentes desempeñen procesos de manera autónoma, requiere entender la forma como la Web semántica almacena los datos y sus diferencias con los paradigmas de almacenamiento usados por la mayoría de gestores de bases de datos en la actualidad.

2.2 Modelo de almacenamiento usado en la Web semántica

El modelo de almacenamiento utilizado en la Web semántica puede ser descrito como una base de datos orientada a grafos en la cual las entidades son representadas como nodos y las relaciones entre nodos como aristas. En este modelo el concepto de jerarquía pierde relevancia, siendo más importantes las relaciones establecidas entre los diferentes nodos; por tanto, al realizar consultas sobre el grafo, lo que en realidad se está efectuando es un recorrido sobre los diferentes recursos enlazados y sus asociaciones.

A continuación se presenta un ejemplo de cómo puede ser representado un conjunto de datos mediante grafos. Inicialmente se define un conjunto de sentencias acerca de un dominio particular, por ejemplo:

- *María está casada.*
- *Juan está casado.*
- *Juan y María son pareja.*

Una vez se han establecido las sentencias que describen cada entidad y las relaciones existentes entre estas, es posible codificar dichas oraciones mediante el uso de un lenguaje formal. RDF (Resource Description Framework) [5] cumple este objetivo, puesto que permite definir entidades con sus atributos y las relaciones que las acompañan. La figura 2 muestra la forma como RDF describe cada recurso, sus atributos y sus relaciones. En el gráfico se pueden apreciar dos entidades (entidad 1 y entidad 2), cada una de las cuales tiene dos propiedades (estadoCivil y nombre), mas una propiedad que las relaciona conjuntamente (casadoCon). Formalmente, cada relación representada por la línea punteada se denomina predicado, de este modo mediante RDF es posible definir triplas del tipo sujeto, predicado, objeto, como por ejemplo “Juan está casado” o “Bogotá es la capital de Colombia”.

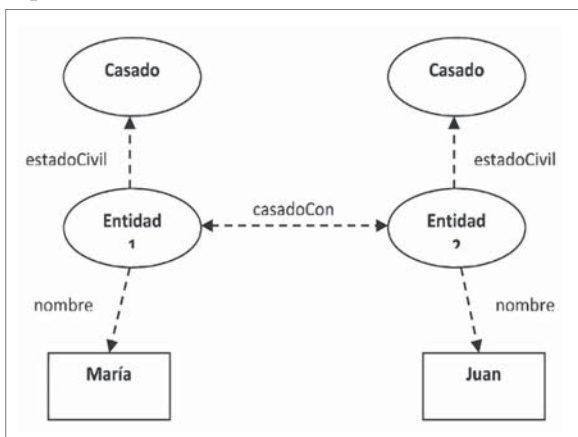


Figura 2. Triplas SPO
Fuente: elaboración propia.

2.3 Definición de triplas mediante RDF (Resource Description Framework)

Cada sentencia representada mediante RDF debe ser dividida en tres partes constituyentes esenciales: el sujeto, el predicado y el objeto, obteniéndose de este modo una tripla SPO (sujeto, objeto, predicado). Una tripla SPO puede ser definida como se muestra a continuación:

- Alemania es parte de la Unión Europea.

En donde:

- El sujeto es: Alemania
- El predicado (propiedad) es: es parte de.
- El objeto es: Unión Europea.

Visualmente esta tripla se representaría como lo muestra la figura 3:



Figura 3. Representación gráfica de una tripla SPO
Fuente: elaboración propia.

De este modo, mediante RDF/XML (la forma XML de RDF)[6] la tripla anterior puede ser representada como se muestra a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-
  rdf-syntax-ns#"

```

```
  xmlns:feature="http://www.geonames.com/
  countries-features#">
```

```
<rdf:Descriptionrdf:about="http://www.
  geonames.com/countries#Germany">
```

```
<feature:partOfrdf:resource="http://www.
  geonames.com/communities#UE"/>
```

```
</rdf:Description>
</rdf:RDF>
```

Las partes que constituyen la tripla representada mediante RDF/XML son:

- Sujeto: <http://www.geonames.com/countries#Germany>.
- Predicado: part of.
- Objeto: <http://www.geonames.com/communities#UE>.

Es importante tener en cuenta que aunque RDF permite definir las estructuras de datos para la Web semántica, en sí mismo no permite describir la semántica o el significado de los datos. Para lograrlo se debe hacer uso de lenguajes apropiados como RDFS (RDF Schema) [7] u OWL (Web Ontology Language) [8].

2.4 Consulta de datos semánticos

Una vez se han declarado las estructuras de datos mediante RDF y se han adicionado anotaciones semánticas a los mismos usando RDFS u OWL, es necesario disponer de un lenguaje que permita representar las preguntas acerca de los datos y sus relaciones. Para cumplir este propósito, el grupo de trabajo DAWG (RDF Data Access Working Group)[9], perteneciente al consorcio de la World Wide Web (W3C), estandarizó el lenguaje de consultas sobre RDF llamado SPARQL (Protocol and RDF Query Language) [10], el cual ofrece las siguientes posibilidades:

- Permite la escritura de consultas no ambiguas globales, puesto que toma ventaja del uso de las URIs como identificadores únicos de cualquier recurso. Adicionalmente, gracias al uso de las ontologías, permite que cualquier búsqueda converja hacia el repositorio de datos más conveniente, reflejando la

visión de la Web semántica de tratar a la red mundial de información como una enorme base de datos.

- Permite explorar datos consultando relaciones desconocidas.
- Permite establecer consultas complejas integrando repositorios diferentes en una sola consulta.
- Permite transformar datos RDF de un vocabulario a otro.

Debido a que las estructuras de datos que componen la Web semántica son representadas mediante triplas RDF, el patrón que SPARQL utiliza para representar las consultas está a su vez basado en la definición de triplas sujeto, objeto, predicado.

Los elementos constituyentes de una consulta escrita en SPARQL son los siguientes:

- *PREFIX (Prefijos de nombres de espacio)*

Ejemplo: PREFIX acm: <<http://www.acm.org/class/1998/ontology#>>

- *SELECT (Conjunto de resultados)*

Ejemplo: SELECT ?title

- *FROM (Conjunto de datos)*

Ejemplo: FROM <http://www.acm.org/class/1998/papers.rdf>

- *WHERE (Patrón de consulta basado en triplas)*

Ejemplo: WHERE {?papers rdf:typeacm:Article-Reference}

- *ORDER BY, DISTINCT, etc (Modificadores)*

Ejemplo: ORDER BY ?title

SPARQL permite expresar de manera formal las preguntas que surjan acerca de un dominio de conocimiento determinado. La siguiente consulta SPARQL define de manera formal la pregunta “¿cuáles son las capitales de los departamentos de Colombia?”:

```
PREFIX onto: <http://ejemplo.com/ontologiaE-
ejemplo#>
SELECT ?capital ?departamento
WHERE{
    ?aonto:nombreCiudad ?capital ;
onto:esCapitalDe ?b .
    ?bonto:nombreDepto ?departamento ;
onto:deptoDeonto:Colombia.
}
```

La consulta anterior retornaría los valores para las variables precedidas por el signo “?” o por el signo “\$”.

2.5 Linked Data

El término Linked Data fue utilizado por Tim Berners-Lee en su artículo “Design Issues: Linked Data” [1] para describir una técnica mediante la cual pudieran ser expuestos, compartidos y conectados datos en la Web semántica haciendo uso de URIs. Berners-Lee definió 4 principios fundamentales para la publicación de datos mediante Linked Data. En la figura 4, se muestra la nube de repositorios semánticos del proyecto Linked Data. Estos principios son:

- El uso de URIs para identificar cualquier tipo de recurso.

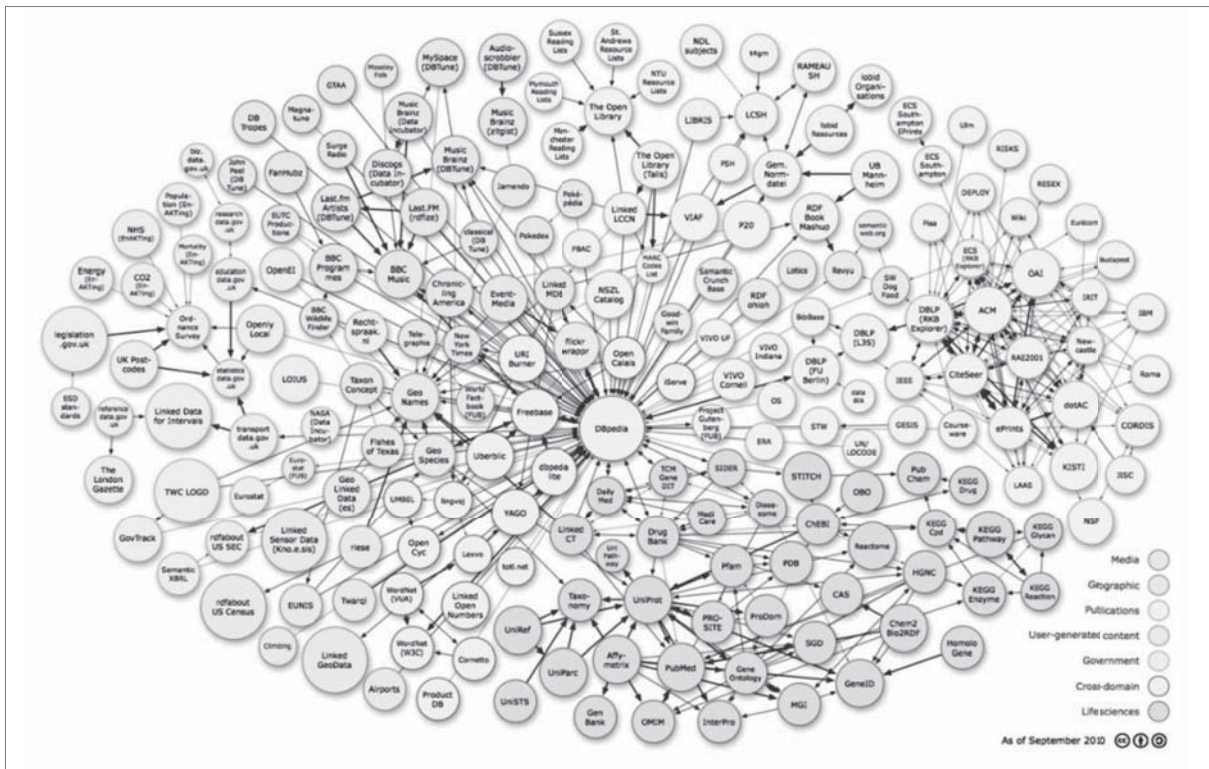


Figura 4. Nube de repositorios semánticos del proyecto LinkedData
Fuente: tomada de [3].

- El uso de URIs HTTP de modo que puedan ser referidas y posteriormente consultadas por personas y agentes inteligentes.
- Proveer información útil acerca de los recursos identificados con las URIs usando formatos estándar como RDF/XML.
- Incluir enlaces que relacionen las URIs de modo tal que se mejore el descubrimiento de información relacionada en la Web.
- Por consiguiente la premisa fundamental detrás del mecanismo de Linked Data es que el valor y utilidad de los datos se incrementa en la medida en que están enlazados con otros datos.

3. DISEÑO Y DESARROLLO DEL PROTOTIPO

El diseño del prototipo fue pensado con el objetivo de priorizar dos aspectos primordiales: por un lado la naturaleza asincrónica de la comunicación

entre la aplicación y los repositorios semánticos y por el otro, el aspecto sencillo e intuitivo requerido por la interfaz gráfica. Para cumplir con los aspectos anteriores se utilizó una arquitectura cliente-servidor que permite repartir la carga de procesamiento entre estas dos partes, teniendo en cuenta que la parte cliente de la aplicación sería la encargada de enviar las consultas a los repositorios, razón por la cual el diseño y desarrollo del componente para consultas debía maximizar la velocidad de transmisión y procesamiento de los datos. La figura 5 muestra el diagrama de despliegue del prototipo.

Para lograr que la parte cliente del prototipo pueda comunicarse de manera eficiente con los repositorios semánticos y asimismo procesar su respuesta, se utilizaron varios mecanismos y tecnologías orientadas a comunicaciones asincrónicas, las cuales se describen a continuación:

- Ajax: el nombre de esta tecnología proviene del acrónimo Asynchronous JavaScript And XML [12] y permite comunicar de ma-

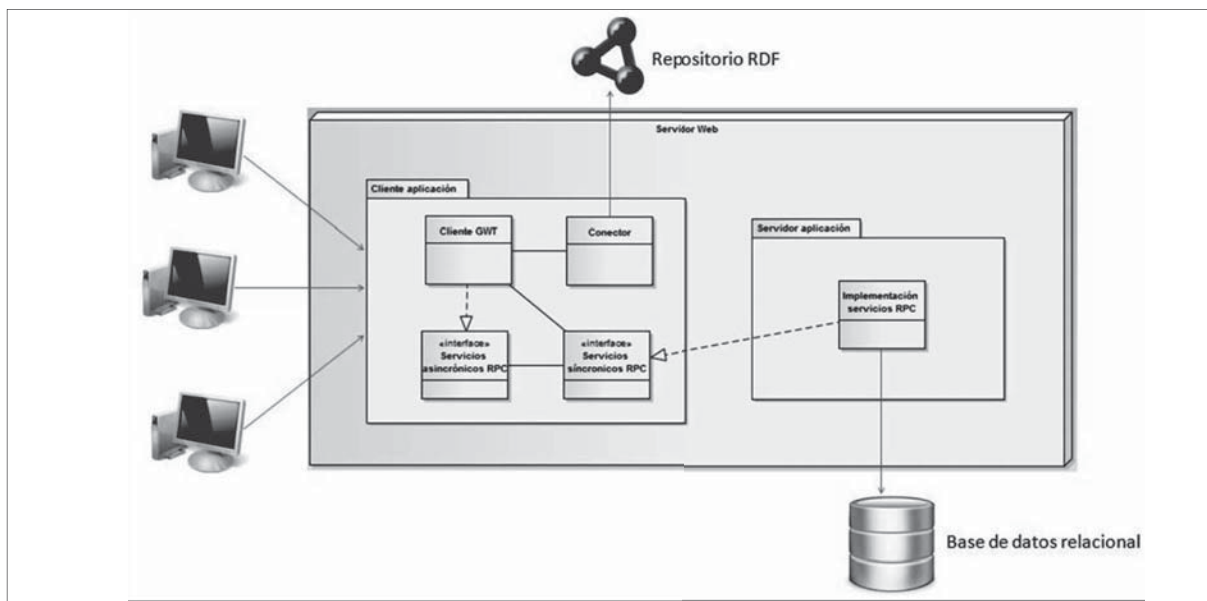


Figura 5. Diagrama de despliegue del prototipo

Fuente: tomada de [11].

nera asíncrona al cliente de una aplicación web con el servidor. Su funcionamiento se basa en el envío de peticiones asíncronas cuya respuesta es cargada en la página web en segundo plano, permitiendo manipular y actualizar el contenido de un sitio sin necesidad de recargarlo completamente. El objeto javascript XML Http Request es quien se encarga de realizar las peticiones HTTP o HTTPS a los servidores Web, en este sentido funciona como una interfaz para la comunicación que puede generar tantas instancias de sí misma como se necesiten. El formato de los datos transmitidos desde el servidor hacia el cliente no tiene que ser necesariamente XML, ya que existen diversos formatos entre los cuales se encuentra JSON (JavaScript ObjectNotation) [13], el cual es usado por la aplicación. Ajax aumenta significativamente la interactividad, velocidad y usabilidad de las aplicaciones, razón por la cual fue adoptada como herramienta base para el desarrollo del proyecto.

- Call tracking (seguimiento de llamada): para aumentar la velocidad de respuesta de la aplicación se definió un modelo de lanzamiento de consultas asíncrono en el cual una consulta compleja es dividida en varias sencillas, cuyos tiempos de ejecución en conjunto son menores al de la consulta original. Sin embargo, surge un inconveniente al utilizar este modelo, ya que resulta imposible conocer el estado de cada llamada realizada puesto que las consultas se vuelven indistinguibles para la aplicación, generando inconsistencias al momento de procesar las respuestas a las peticiones realizadas. La solución a este problema consiste en agregar un identificador a cada llamada realizada con el fin de poder rastrearla y así conocer su estado. Este identificador aumenta a medida que se realicen más llamadas en paralelo.

- Cross-siterequest: esta técnica permite evitar o pasar por alto la restricción SOP (Same Origin Police) implementada en la mayoría de los navegadores web actuales. SOP es un concepto de seguridad que consiste en permitir la ejecución de scripts en sitios web pertenecientes al mismo dominio pero evitando que se ejecuten scripts o se tenga acceso a atributos o métodos entre sitios web de diferente dominio. Esta restricción hacía imposible la ejecución de consultas en repositorios semánticos desde una aplicación web, ya que el dominio al que pertenecen ambos sitios puede ser diferente. La técnica *cross-siterequest* consiste en crear dinámicamente un elemento `<script>` en la página web de la aplicación que va a lanzar la petición al sitio remoto. Tras la ejecución de la consulta, los datos son enviados de vuelta al método del aplicativo que hizo el llamado para posteriormente realizar el procesamiento de los datos.

Como se mencionó anteriormente, uno de los aspectos priorizados fue la definición y construcción de una interfaz gráfica sencilla e intuitiva para realizar las consultas a los repositorios semánticos. Para lograrlo el aplicativo debe contar con un mecanismo que, o bien permite al usuario digitar libremente el criterio de búsqueda a usar, o mostrará un conjunto de criterios de búsqueda de posible uso. Se escogió la segunda opción por dos razones: primero, contar con un catálogo de términos estándar muy amplio y de uso común garantiza que cualquier búsqueda arroja como mínimo un resultado, ahorrando tiempo al usuario del aplicativo en largas y estériles búsquedas. La segunda razón tiene que ver con motivos prácticos y se resume en el hecho de que los datos al interior de la mayoría de repositorios semánticos de ciencias de la computación e informática se encuentran catalogados mediante el sistema de clasificación ACM [14], lo cual implica que si se usa un conjunto de criterios de búsqueda que no

se ajuste a los definidos en el sistema, se requiere el desarrollo de un procedimiento que traduzca los términos originalmente ingresados a los que el repositorio identifica y utiliza en su estructura interna.

A nivel visual se decidió estructurar el conjunto de criterios de búsqueda como un árbol, que puede contener términos de búsqueda que a su vez

son ramas que contienen otros términos de búsqueda. Estos términos son cargados de la base de datos bajo demanda, es decir, a medida que son requeridos, aumentando el tiempo de cargado de la aplicación y evitando saturar la memoria con datos sin uso.

La figura 6 muestra el diagrama de secuencia para el proceso de consulta de un repositorio

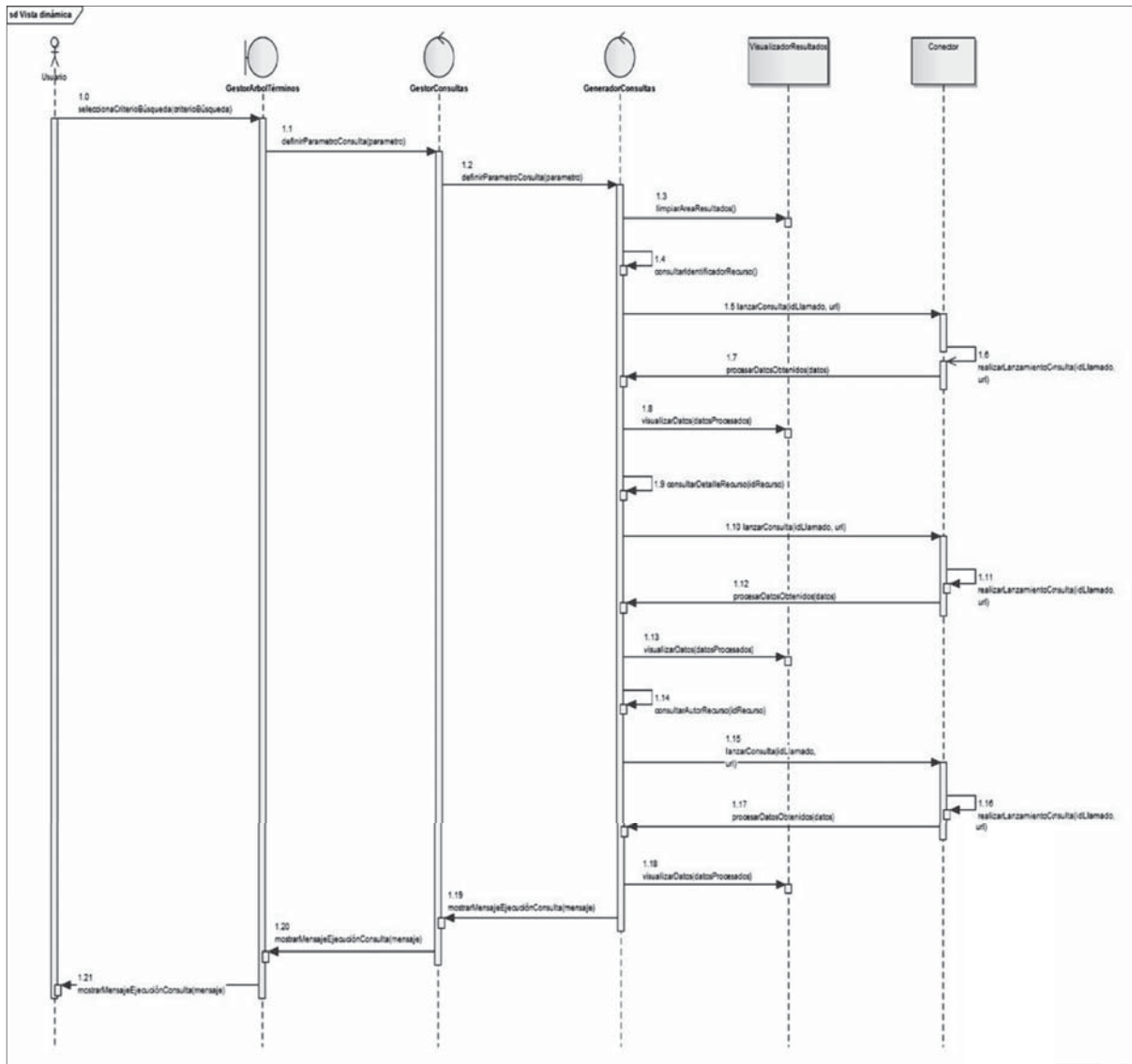


Figura 6. Diagrama de secuencia para la consulta de un repositorio semántico

Fuente: tomada de [11].

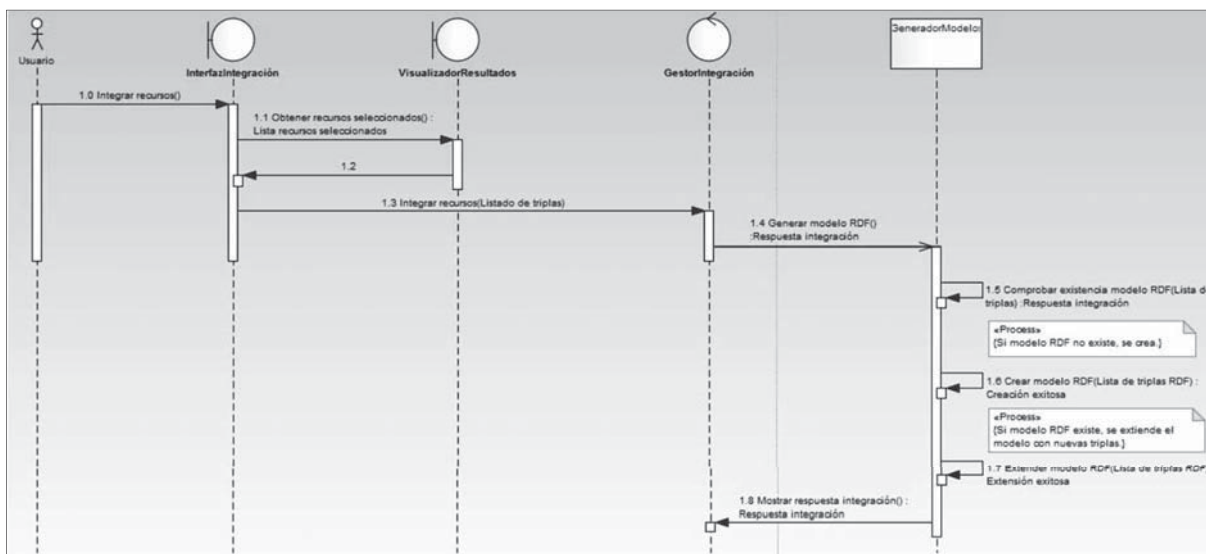


Figura 7. Diagrama de secuencia para el proceso de integración de recursos web
Fuente: tomada de [11].

semántico. El usuario inicia la ejecución de la consulta mediante la selección de un criterio de búsqueda del árbol de términos de la interfaz gráfica.

Otro aspecto central de la aplicación es la posibilidad que tiene un usuario de seleccionar cuales de los recursos web obtenidos (en este caso artículos) desea integrar y almacenar en un documento RDF, el cual podrá ser extendido con nuevos recursos obtenidos de consultas posteriores. Esta funcionalidad fue desarrollada utilizando el framework para desarrollo de aplicaciones semánticas Jena, el cual entre otras características permite la creación de documentos de triplas RDF. En el caso del prototipo, dichas triplas no son otra cosa que una representación de los recursos web obtenidos, las relaciones que existen entre los mismos y el tipo y nombre que dichas relaciones tienen.

La figura 7 representa el proceso de integración de recursos web, el cual se caracteriza por permitirle al usuario seleccionar individualmente los recursos a integrar o si lo desea, seleccionar

todos los recursos obtenidos. Este proceso no ocurre en el lado cliente del prototipo, por lo cual se beneficia de la velocidad de procesamiento de la máquina servidor, velocidad que en términos generales resulta ser considerablemente mayor a la alcanzada por la mayoría de máquinas cliente.

La visualización de los recursos web se realiza mediante la ejecución de una librería para la representación visual de documentos RDF. Este proceso toma como entrada el documento RDF generado en la integración de recursos web y lo transforma en su representación gráfica, en la cual los recursos web son dibujados como nodos y las relaciones entre los mismos como aristas. La figura 8 representa dicho proceso.

3.1 Diseño y desarrollo de las pantallas del prototipo

El diseño y desarrollo de la interfaz gráfica de usuario mantuvo la sencillez, haciendo énfasis en la usabilidad. La figura 9 muestra la interfaz antes de la realización de consultas.

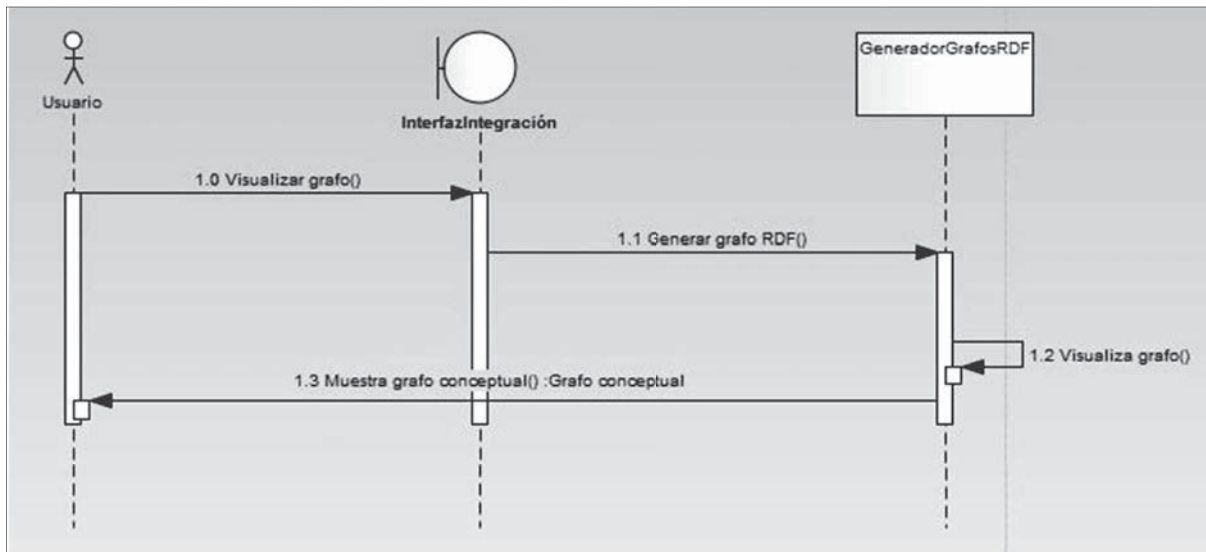


Figura 8. Diagrama de secuencia para el proceso de visualización del grafo conceptual
Fuente: tomada de [11].

Una vez obtenidos los resultados, estos se muestran en paneles independientes que contienen los detalles de cada recurso, así como una casilla de marcado para seleccionar aquellos que se desean

integrar (figura 10). El prototipo también dispone de un botón que permite seleccionar todos los recursos.



Figura 9. Interfaz de consulta
Fuente: tomada de [11].

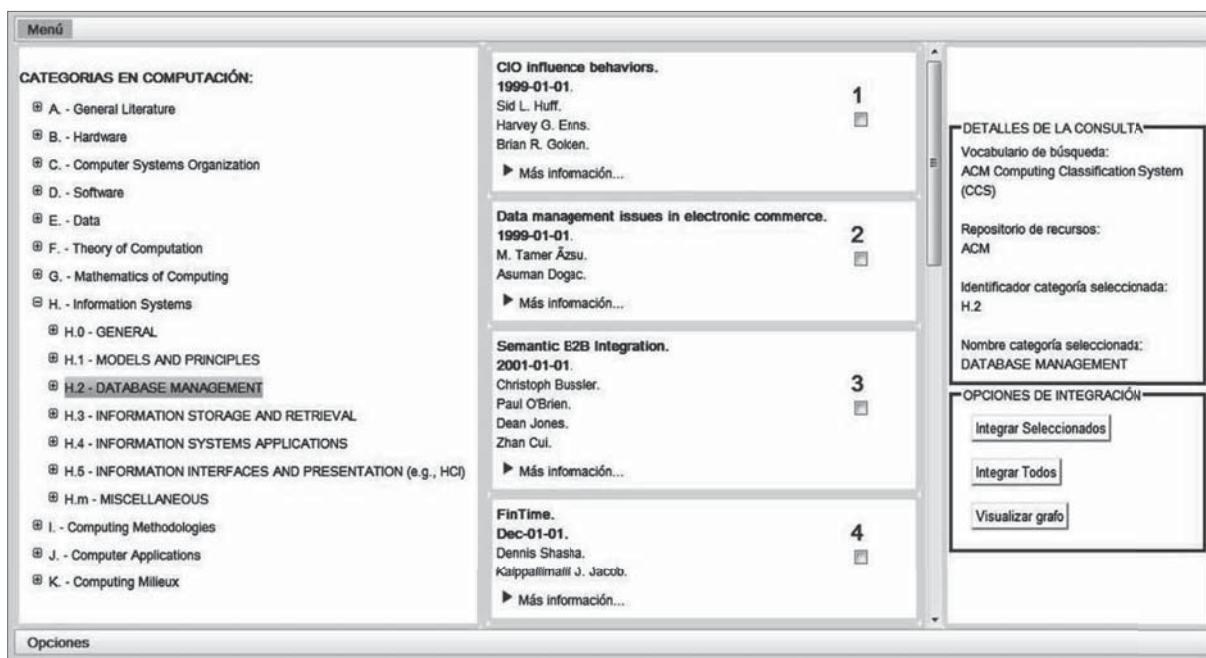


Figura 10. Resultados obtenidos tras la selección de un criterio de búsqueda

Fuente: tomada de [11].

Por último, la figura 11 muestra un segmento del grafo generado a partir del procesamiento de un documento de triplas RDF, mostrando que cada recurso web es representado por un nodo y cada relación entre nodos como una arista de este.

4. METODOLOGÍA

En este proyecto se tomó como base metodológica el modelo de proceso para el desarrollo de software OpenUP [15]. En la fase inicial del proyecto se efectuó la recopilación y estructuración de recursos bibliográficos, se estableció de manera progresiva el estado del arte [16] y se llevó a cabo una lectura cuidadosa de las fuentes de información provistas por el director del proyecto, acompañada por la exploración y lectura de la información perteneciente a los diversos recursos electrónicos existentes sobre el tema a tratar.

La redacción del estado del arte se realizó de manera paralela a la lectura y análisis de las

fuentes de información a partir de una revisión conjunta del material, de modo que se alcanzara una versión madura del documento.

En esta misma fase inicial se estableció la visión del proyecto, la cual define los objetivos estratégicos y funcionales que a largo plazo debe incorporar el prototipo. Se realizó una descripción de alto nivel del problema y su solución con base en un conjunto inicial de requerimientos.

También se estableció el glosario del sistema con el fin de definir los términos y conceptos importantes utilizados por los participantes del proyecto. Su objetivo es facilitar una colaboración efectiva reduciendo la ambigüedad e inexactitud en la comunicación. El plan general del proyecto fue definido en esta fase de inicio con el propósito de determinar los objetivos de alto nivel de cada una de las iteraciones así como sus correspondientes hitos. Se especifica-

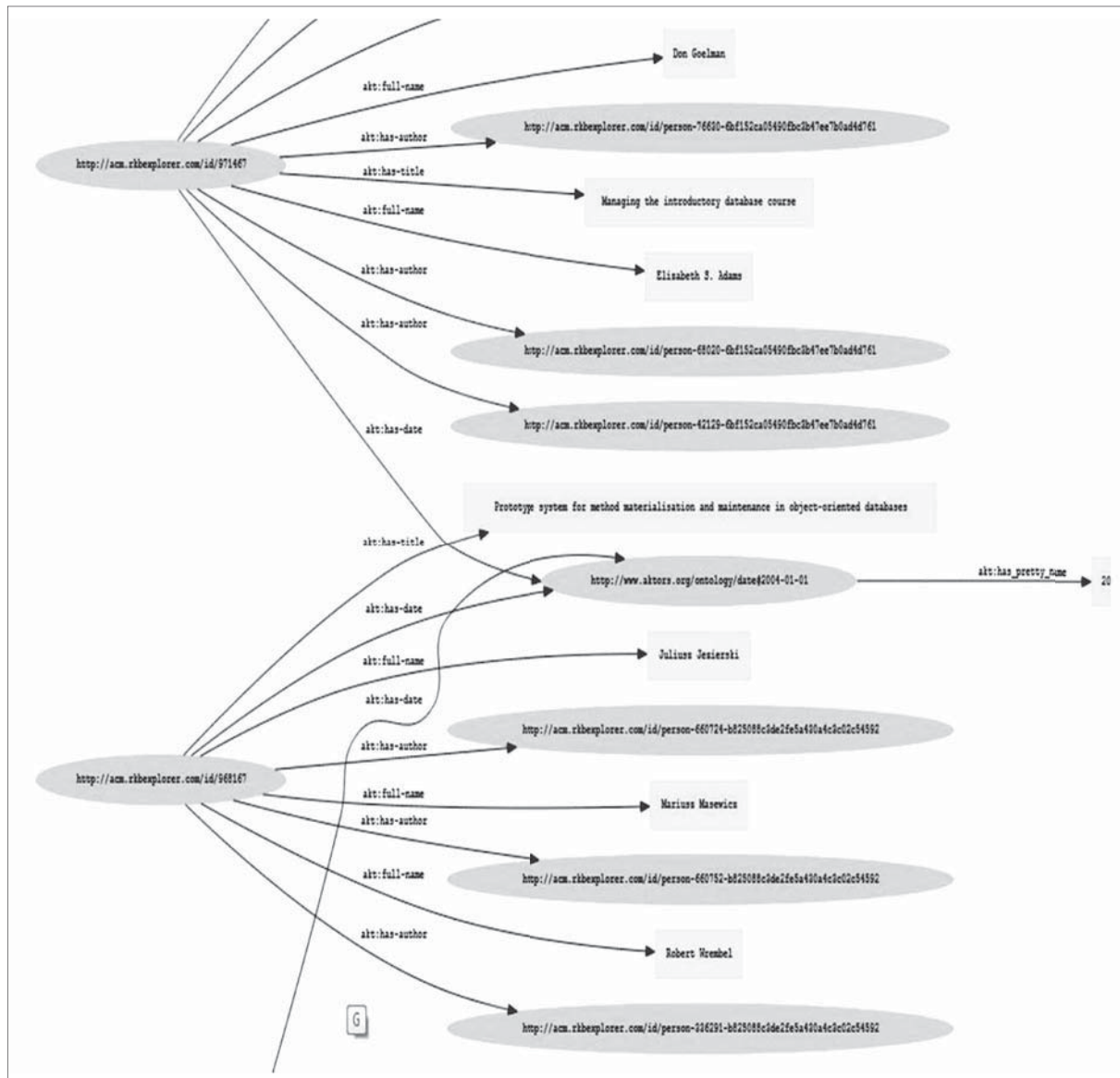


Figura 11. Grafo generado a partir de un documento RDF

Fuente: tomada de [11].

ron los requerimientos y se definieron los casos de uso base para establecer al final de la fase de inicio la arquitectura base del sistema.

En la fase de elaboración se detallaron los requerimientos y el modelo de casos de uso con el objetivo de ajustar la arquitectura y obtener una versión inicial del diseño del sistema. En esta

fase se desarrolló un prototipo que permitió encontrar nuevos requerimientos y refinar tanto la arquitectura como el diseño de la solución.

La fase de construcción permitió realizar un refinamiento mayor al diseño de la solución final, se desarrolló una versión mejorada del prototipo, se evaluaron los artefactos y se ejecutaron

los ajustes según las modificaciones realizadas a los modelos.

La fase de transición permitió incorporar nuevas funcionalidades al prototipo, refinar los componentes desarrollados y efectuar el despliegue de la solución en un servidor de prueba. Durante el desarrollo de cada una de las fases del proyecto se realizaron tareas continuas de documentación, verificación y evaluación de artefactos.

Se efectuó seguimiento al desarrollo del proyecto mediante la programación de reuniones formales e informales lideradas por el director del proyecto. En estas reuniones se revisaron y evaluaron los artefactos desarrollados y se hicieron los ajustes necesarios a las iteraciones de cada fase.

5. RESULTADOS

Se realizó el proceso de consulta de repositorios de recursos web semánticos (en particular aquellos que contienen información sobre publicaciones científicas) mediante el diseño y desarrollo de un mecanismo basado en el uso de consultas SPARQL, el lanzamiento de peticiones asincrónicas y la técnica cross-site request, mecanismo que permite a la aplicación realizar peticiones a repositorios que poseen un punto final SPARQL que acepte consultas y retorne los resultados de las mismas a través de HTTP. Con el objetivo de reducir el tiempo de transferencia de los datos obtenidos, a cada petición realizada al repositorio se le agregó un atributo que establece el formato de los datos retornados. Para este proyecto resultó conveniente en términos de rendimiento el uso del formato ligero para intercambio de datos JavaScript Object Notation (JSON). Las pruebas realizadas demostraron que el uso de JSON reduce el tiempo de transferencia de los datos desde el repositorio hasta el cliente de la aplicación en comparación con el uso de XML.

El proceso de integración de recursos web fue resuelto mediante el uso del framework para desarrollo de aplicaciones semánticas Jena [17]. La clase Java encargada de ejecutar las funciones de Jena se encuentra alojada en el servidor y es invocada por un servicio RPC que es llamado desde el cliente de la aplicación. El proceso en términos generales recibe un conjunto de triplas en formato de cadena de texto y lo transforma en un grafo RDF/XML almacenado en un archivo. Esta operación puede ser realizada sobre un grafo generado, lo cual permite extenderlo manteniendo la estructura anterior. El archivo generado cumple el estándar definido por World Wide Web Consortium (W3C) para la creación de grafos RDF, lo cual permite que el archivo pueda ser leído por otra aplicación, de modo que los nodos y relaciones incluidos en él puedan ser interpretados, consultados mediante SPARQL y visualizados como un grafo conceptual.

La visualización de recursos web semánticos mediante grafos conceptuales se logró utilizando la librería RAP (RDF API for PHP) [18], la cual genera grafos conceptuales a partir de un documento XML correctamente estructurado. El grafo generado es un archivo de gráficos vectoriales escalables SVG [19], recomendación de la W3C habilitada en la mayoría de navegadores web, que permite mostrar imágenes bidimensionales en dicho formato tanto estáticas como dinámicas.

La redacción del estado del arte sobre las técnicas, tecnologías y lenguajes que permiten realizar consultas sobre repositorios de recursos semánticos, así como la integración e intercambio de dichos repositorios, se realizó gracias a una base bibliográfica y de recursos electrónicos lo más fiable y actualizada posible. En todo momento se verificaron las fuentes y origen de la información y la construcción del documento se basó en la secuencia lectura, reflexión, discusión y redacción.

6. CONCLUSIONES

El éxito de las iniciativas encaminadas a proveer el uso y publicación de datos semánticos depende en gran medida del desarrollo, divulgación y apropiación de los estándares abiertos en los que se basa el crecimiento del espacio global de datos de la Web. De modo recíproco, el mejoramiento de los estándares abiertos para publicación de datos relacionados depende del aumento de proyectos que adopten e implementen el concepto.

El aumento de datos semánticos disponibles incrementa en igual medida la necesidad de contar con agentes inteligentes en los cuales delegar la búsqueda de información. Por consiguiente las investigaciones que dirijan sus objetivos en esa dirección permitirán complementar los resultados logrados hasta el momento y ayudarán a construir la visión de una web de datos global, abierta e interoperable.

La utilización de ontologías para representar conocimiento en una red de datos semánticos relacionados, permitirá brindar el nivel de formalización conceptual requerido para que agentes de software proporcionen inteligencia a la Web.

El documento del estado del arte es solo un punto de partida para el mejoramiento, ampliación y actualización de este, de modo que trabajos

futuros mejoren el contenido y calidad del texto corrigiendo las deficiencias existentes en este.

Las peticiones realizadas mediante la técnica cross-site request representan un riesgo de seguridad debido a la capacidad que tienen los repositorios consultados de ejecutar código javascript arbitrariamente. Este riesgo debe ser debidamente tratado en aplicaciones de producción mediante el establecimiento de políticas y lineamientos de seguridad que permitan verificar la confiabilidad de los repositorios consultados y promuevan la auditoría continua de estos.

7. FINANCIAMIENTO

El presente artículo tuvo origen en el proyecto de grado titulado con el mismo nombre, el cual fue presentado ante el proyecto curricular de Ingeniería de Sistemas de la Universidad Distrital Francisco José de Caldas, el cual lo avaló y brindó el apoyo académico, tecnológico y logístico requerido. Asimismo, el grupo de investigación Interoperabilidad tecnológica y semántica (INTECSE), adscrito al proyecto curricular de Ingeniería de Sistemas, respaldó este proyecto y mediante la asesoría brindada por sus integrantes, contribuyó en gran medida en la consecución de los objetivos planteados.

REFERENCIAS

- | | |
|---|---|
| <p>[1] T. Berners-Lee. "Linked Data". [Online]. Available: www.w3.org/DesignIssues/LinkedData.html, Jul. 27, 2006, [Dec 10, 2010].</p> | <p>[3] R. Cyganiak. "About the linking open data cloud". [Online]. Available: http://richard.cyganiak.de/2007/10/lod/, Oct. 2007, [Sep 1, 2011]</p> |
| <p>[2] "Linked Data - Connect Distributed Data across the Web". [Online]. Available: www.linkeddata.org, [Dec 1, 2011]</p> | <p>[4] P. Castells. "La Web Semántica", en C. Vázquez, <i>Sistemas Interactivos y Colaborativos en la Web</i>, 1ª ed., vol. 1. Castilla-La</p> |

- Mancha: Universidad de Castilla-La Mancha, 2005, pp. 195-211.
- [5] “ResourceDescription Framework (RDF)”. [Online]. Available: <http://www.w3.org/RDF/>, Feb. 10, 2004 [Jul. 10, 2011].
- [6] “RDF/XML Syntax Specification (Revised)”. [Online]. Available: <http://www.w3.org/TR/REC-rdf-syntax/>, Feb. 10, 2004 [Jul. 10, 2011].
- [7] “RDF Vocabulary Description Language 1.0: RDF Schema”. [Online]. Available: <http://www.w3.org/TR/rdf-schema/>, Feb. 10, 2004 [Jul. 10, 2011].
- [8] “OWL Web Ontology Language Overview”. [Online]. Available: <http://www.w3.org/TR/owl-features/>, Feb. 10, 2004 [Jul. 10, 2011].
- [9] “RDF Data Access Working Group”. [Online]. Available: <http://www.w3.org/2001/sw/DataAccess/issues> [Sep. 10, 2011]
- [10] “SPARQL Query Language for RDF”. [Online]. Available: <http://www.w3.org/TR/rdf-sparql-query/>, Jan. 15, 2008 [Sep. 12, 2011].
- [11] A. Poveda, “Diseño y desarrollo de un prototipo para la integración y navegación semántica de recursos web mediante grafos conceptuales”, Tesis Pregrado. Bogotá. Universidad Distrital Francisco José de Caldas, 2011.
- [12] J.J. Garrett. “Ajax: A New Approach to Web Applications”. [Online]. Available: http://www.robertspahr.com/teaching/nmp/ajax_web_applications.pdf, Feb. 18, 2005 [Sep 4, 2011]
- [13] “Introducing JSON”. [Online]. Available: <http://www.json.org/>, [Sep 4, 2011]
- [14] ACM Digital Library, *ACM Computing Classification System*. [Online]. Available: <http://dl.acm.org/>, [Sep 2, 2011]
- [15] R. Balduino. “Introduction to Open UP (Open Unified Process)”. [Online]. Available: <http://www.eclipse.org/epf/general/OpenUP.pdf>, Ago. 2007 [Sep 10, 2011]
- [16] G. Moreno Hernández, “Cómo investigar Técnicas documentales y de campo”. México: Edere, 1997.
- [17] Jena, *Semantic Web Framework*, [Online]. Available: <http://incubator.apache.org/jena/> [Sep. 1, 2011]
- [18] RAP, *RDF API for PHP*, [Online]. Available: <http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/>, [Sep 1, 2011]
- [19] “Scalable Vector Graphics (SVG) 1.1” [Online]. Available: <http://www.w3.org/TR/SVG/>, Aug. 16, 2001 [Oct. 15, 2011]