Universidad de Cuenca

Facultad de Ingeniería

Maestría en Gestión Estratégica de Tecnologías de la Información

Proyecto de Tesis

PLATAFORMA PARA LA VISUALIZACIÓN DE DATOS MULTIDIMENSIONALES BASADOS EN TECNOLOGÍAS SEMÁNTICAS

Autor:

Ing. Andrea Daniela Morales Rodríguez

C.I. 0102789419

Director:

Ing. Víctor Hugo Saquicela Galarza, PhD.

C.I. 0103599577

Grado académico: Magíster

Cuenca, junio 2017

RESUMEN

La plataforma “Repositorio Ecuatoriano de Investigadores”, permite identificar las áreas de conocimiento mediante la agrupación de las palabras claves definidas en las publicaciones científicas que pertenecen a los investigadores ecuatorianos, a través de la utilización de técnicas de Minería de datos y Tecnologías Semánticas. Actualmente, la información de los autores y sus publicaciones son almacenadas en un repositorio en formato del Marco de Descripción de Recursos (RDF). La visualización de la información en la actualidad se realiza mediante consultas estáticas desarrolladas por los programadores del proyecto, por lo que no le permite al usuario interactuar a través de consultas dinámicas. Con esta problemática, el presente trabajo define una arquitectura de software para mejorar las herramientas actuales de visualización basadas en el concepto de cubos de datos multidimensionales, utilizando el vocabulario de cubo de datos en RDF, permitiendo implementar búsquedas dinámicas por parte de los usuarios.

Palabras clave: Tecnología Semántica, RDF, Vocabulario de Cubo de datos en RDF, QB, Opencube Toolkit, modelos multidimensionales.

ABSTRACT

The "Researcher's Ecuadorian Repository" platform allows the identification of knowledge's areas through the grouping of key words defined in the scientific publications carried out by Ecuadorian researchers through the use of data mining techniques and Semantic Technology. Currently, the information of the authors is stored in a repository form called "Resource Description Framework" (RDF). The visualization of the information at present is made by static requests developed by the programmers of the project, reason why it does not allow to make dynamic requests by the users, therefore, the present work defines a software architecture to improve the current visualization tools based on the concept of multidimensional data cubes described using the Cubo de datos en RDF vocabulary, allowing to implement dynamic searches.

Keywords: Semantic Technology, RDF, Cubo de datos en RDF Vocabulary, QB, Opencube Toolkit, Multidimensional models.

CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE CONSULTAS SPARQL

CLÁUSULA DE LICENCIA Y AUTORIZACIÓN PARA LA PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL

Universidad de Cuenca
Maestría en Gestión Estratégica de Tecnologías de la Información

CLÁUSULA DE LICENCIA Y AUTORIZACIÓN PARA LA PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL

Andrea Daniela Morales Rodríguez, en calidad de autora y titular de los derechos morales y patrimoniales del trabajo de titulación "Plataforma para la visualización de datos multidimensionales basados en Tecnologías Semánticas", de conformidad con el Art. 114 del CÓDIGO ORGÁNICO DE LA ECONOMÍA SOCIAL DE LOS CONOCIMIENTOS, CREATIVIDAD E INNOVACIÓN reconozco a favor de la Universidad de Cuenca una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos.

Asimismo, autorizo a la Universidad de Cuenca para que realice la publicación de este trabajo de titulación en el Repositorio Institucional, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Cuenca, 13 junio 2017

Andrea Daniela Morales Rodríguez

C.I: 0102789419

CLÁUSULA DE PROPIEDAD INTELECTUAL

Universidad de Cuenca
Maestría en Gestión Estratégica de Tecnologías de la Información

CLÁUSULA DE PROPIEDAD INTELECTUAL

Andrea Daniela Morales Rodríguez, autora del Trabajo de Titulación "Plataforma para la visualización de datos multidimensionales basados en Tecnologías Semánticas", certifico que todas las ideas, opiniones y contenidos expuestos en la presente investigación son de exclusiva responsabilidad de su autora.

Cuenca, 13 junio 2017

Andrea Morales

Andrea Daniela Morales Rodríguez

C.I: 0102789419

DEDICATORIA

Dedico este trabajo a Dios por brindarme la oportunidad de mejorar y las fuerzas para conseguirlo

… A mi esposo que con su apoyo y amor incondicional me ha permitido cumplir la meta propuesta.

… A mis hijas Camila y Manuela, que son y serán el motor de mi vida, quienes me impulsan a mejorar día a día.

… A mis padres y hermanas por el apoyo permanente brindado para concluir con este objetivo.

Andrea.

AGRADECIMIENTOS

Agradezco a todas las personas que colaboraron para la realización de este trabajo en especial al Ing. Víctor Saquicela Galarza, PhD. Director de Tesis, quien ha sabido guiarme y apoyarme en el desarrollo de este trabajo, y a la Red Nacional de Investigación y Academia del Ecuador – RedCEDIA, quien me ha apoyado mediante la utilización de la plataforma REDI, sobre la cual se sustenta el presente trabajo de tesis.

1 INTRODUCCIÓN

En la actualidad se cuenta con la plataforma "Red Ecuatoriana de Investigadores" (REDI), que permite identificar áreas de conocimiento mediante la agrupación de palabras claves definidas en las publicaciones científicas que realizan los investigadores ecuatorianos, a través de la utilización de técnicas de Minería de datos y Tecnologías Semánticas. La información es almacenada en un repositorio en formato Marco de descripción de recursos (RDF) en base de tripletas (sujeto, predicado, objeto), que permite la manipulación de grandes cantidades de datos, interoperabilidad entre aplicaciones a través del intercambio de información.

La visualización de esta información en la actualidad se realiza mediante consultas estáticas o pre-definidas por los programadores del proyecto, es decir, si se desea incluir una nueva visualización ésta deberá ser implementada por los desarrolladores con conocimientos de Tecnologías Semánticas, y ésta no se visualizará de manera inmediata. Con esta problemática, el presente trabajo de tesis tiene como propósito principal, mejorar las herramientas actuales de visualización de la información almacenada en el repositorio REDI continuando el trabajo con la implementación de Tecnologías Semánticas, RDF y la utilización de cubos de datos multidimensionales basados en RDF que permiten implementar búsquedas dinámicas por parte de los usuarios.

Para lograr con este cometido, se analizó a profundidad la arquitectura de la plataforma REDI para entender su funcionamiento y determinar que mejoras se pueden implementar, adicionalmente, se estudió las herramientas existentes para realizar la visualización de modelos multidimensionales basadas en Cubo de datos en RDF, se analizó los diferentes tipos de Arquitectura de software para determinar la o las mejores para este propósito y combinarlas con Tecnologías Semánticas.

Basados en la Arquitectura de Software Semántica definida se desarrolló un proceso de transformación de la información que actualmente se encuentra almacenada en el repositorio REDI en formato RDF a Cubo de datos en RDF, información que se guarda en un almacén de datos semántico (SDW), para que mediante herramientas se realice la visualización de la información disponible de forma dinámica.

1.1 ESTRUCTURA DEL TRABAJO DE TESIS

El presente trabajo de tesis se encuentra estructurado en 7 capítulos, en el capítulo 1 se determina la investigación a realizar, los objetivos y el alcance del trabajo; en el capítulo 2 se encuentra los antecedentes del proyecto y trabajos relacionados; en el capítulo 3 se trata el marco teórico para contextualizar los términos que se van a utilizar a lo largo de este trabajo; en el capítulo 4, se presenta el estado del arte donde

se indica el estado actual de las herramientas, plataformas y arquitecturas existentes que se relacionan con en este estudio; en el capítulo 5; se presenta la arquitectura de software escogida para realizar el desarrollo de la plataforma planteada; en el capítulo 6 se desarrolla el prototipo de acuerdo a la arquitectura de software semántica planteada y, finalmente, en el capítulo 7 se presenta las conclusiones del trabajo realizado y se plantean trabajos futuros.

1.2 OBJETIVOS

En esta sección se plantean los objetivos del presente trabajo de tesis a desarrollar.

1.2.1 Objetivo General

Desarrollar un prototipo de una plataforma que permita la visualización de datos multidimensionales basados en Tecnologías Semánticas.

1.2.2 Objetivos Específicos

- Analizar las herramientas existentes para realizar la visualización de modelos multidimensionales en Cubo de datos en RDF[1] (QB).
- Analizar las herramientas que permiten realizar la transformación de diferentes fuentes de datos a Cubo de datos en RDF
- Analizar Arquitecturas de Software para determinar la más favorable para la elaboración de la plataforma requerida
- Definir una Arquitectura de Software con Tecnología Semántica
- Definir un proceso que permita transformar los datos que se encuentran en RDF a Cubo de datos en RDF
- Definir la visualización de la información del proyecto REDI, mediante modelos multidimensionales

1.3 ALCANCE

Este trabajo de tesis se desarrolla sobre la plataforma REDI, e iniciará con un análisis general de la misma para determinar las mejoras que se puedan realizar en torno a la visualización de los datos,

[1] *https://www.w3.org/TR/vocab-data-cube/*

seguidamente, se investigará sobre las Arquitecturas de Software disponibles para determinar la más favorable para la definición de la plataforma requerida donde se incorporará Tecnologías Semánticas.

Posteriormente, se definirá un proceso intermedio para transformar los datos que se encuentran en RDF a RDF DataCube, de esta manera se dispondrá de datos multidimensionales. Seguidamente, se realizará un análisis de herramientas de visualización para modelos multidimensionales para determinar la idónea de acuerdo a los datos disponibles. Finalmente, se desarrollará un prototipo que permita la visualización de los datos multidimensionales basados en Tecnología Semántica.

1.4 PREGUNTAS DE INVESTIGACIÓN

Como se describió anteriormente el proyecto REDI actualmente almacena la información en un repositorio RDF que permite la generación de visualizaciones de las publicaciones de los investigadores de manera predefinida. Es necesario realizar mejoras en las técnicas de visualización a partir del repositorio RDF disponible, y continuar trabajando con Tecnologías Semánticas. Para disponer de visualizaciones dinámicas sobre los datos, es necesario utilizar conceptos de modelos multidimensionales sobre el repositorio RDF disponible, para esto se requiere de la utilización de conceptos basados en Cubo de datos en RDF, el cual permite el manejo de modelos multidimensionales basados en RDF.

Con estos antecedentes se pretende responder:

a) ¿Es posible realizar el proceso de transformación de la información almacenada en el repositorio RDF a modelos multidimensionales continuando con el uso de Tecnologías Semánticas?;
b) ¿La implementación de modelos multidimensionales continuando con el uso de Tecnologías Semánticas, ayudará al usuario a obtener mayor información mediante el uso de consultas dinámicas?

2 ANTECEDENTES

En este capítulo se presentan los antecedentes sobre el presente trabajo de tesis, y los trabajos relacionados al proyecto "Repositorio de Investigadores del Ecuador (REDI)", que permite un mejor entendimiento de este trabajo.

2.1 ANTECEDENTES

En la actualidad existe una gran cantidad de profesionales ecuatorianos que han realizado sus estudios de postgrados a nivel internacional con el afán de mejorar sus aptitudes. Una parte de estos profesionales son investigadores/docentes de las Instituciones de Educación Superior del Ecuador (IES) que regresan con el interés de realizar investigación en el país, una forma para hacerlo es participar en proyectos de I+D+i que reúnen a investigadores internos o externos a la Institución que pertenece el investigador. En muchos de los casos los investigadores desconocen con quien trabajar dentro de una misma área de conocimiento y se complica más si se trata de diferentes áreas.

A pesar del desconocimiento, los investigadores buscan a sus pares en diferentes fuentes de información, si la institución dispone de esta información entrega al interesado, caso contrario el investigador continúa en la búsqueda de información mediante motores de búsqueda (*Google Scholar, Research Gate*, etc.) o en bases de datos digitales a los investigadores o temas de interés para averiguar quién está trabajando en su temática. Una vez cuente con la información necesaria, pueden desarrollar proyectos de I+D+i o redes de colaboración que generan en algunos casos la publicación de artículos científicos en congresos, revistas científicas, repositorio institucional, etc.

Actualmente, en el Ecuador existe un gran problema para los investigadores que desean buscar pares académicos para realizar su investigación, puesto que no se dispone de una plataforma que permita detectar con una sola herramienta las áreas de conocimiento en las que se encuentren trabajando los investigadores ecuatorianos, permita identificar las publicaciones que realizan, y visualice la información encontrada de manera dinámica. Para tratar de resolver este problema, la Red Nacional de Investigación y Educación del Ecuador – RedCEDIA con el apoyo de la Universidad de Cuenca, desarrollaron una plataforma basados en Tecnologías Semánticas, a la que llamaron "Red Ecuatoriana de Investigadores"[2] (REDI), que permite

[2] http://redi.cedia.org.ec/

identificar las áreas de conocimiento mediante la agrupación de palabras claves definidas en las publicaciones científicas, a través de la utilización de técnicas de Minería de datos.

2.2 PROYECTO "REPOSITORIO ECUATORIANO DE INVESTIGADORES - REDI"

El presente trabajo de tesis toma como base los resultados obtenidos del proyecto REDI, cuyo objetivo es generar una herramienta que permita la creación de un repositorio semántico con datos de investigadores con sus respectivas publicaciones, permitiendo identificar diferentes investigadores y áreas similares de conocimiento de las Instituciones de Educación Superior (IES). Para lograr los objetivos del proyecto REDI, los investigadores realizaron el acceso a diferentes fuentes de información como: *Google Scholar, Microsoft Academics, DBLP, Scopus* entre otras, para extraer las publicaciones realizadas por sus autores.

Arquitectura del proyecto REDI

La arquitectura del proyecto REDI, se encuentra compuesta por la fuente de información, y cinco módulos de desarrollo: extracción de autores, extracción de publicaciones, integración de las publicaciones, detección de áreas similares de conocimiento y la visualización de la información.

Inicialmente, la información base de autores disponible para realizar la extracción de autores y publicaciones se encuentra almacenada en los repositorios institucionales de las universidades miembros de CEDIA, que, mediante un proceso de transformación utilizando reglas y principios de Datos enlazados[3] se transformó a formato del Marco de descripción de recursos (RDF). Seguidamente, ésta nueva información se almacenó en un repositorio RDF que permite realizar el proceso de extracción y validación de la información de autores y publicaciones, adicionalmente, se almacena información de otras fuentes de información como: *Google Scholar, Microsoft Academics, DBLP y Scopus*. A continuación, cuando se realiza la integración de las publicaciones en el repositorio es necesario realizar un proceso de validación para evitar que la información se encuentre errónea y duplicada. Posteriormente, se realiza la detección de áreas similares de conocimiento mediante la utilización de técnicas de minería de datos a partir de las palabras claves de las publicaciones. Finalmente, se realiza la visualización de los datos procesados mediante la utilización de nube de etiquetas, grafos y agrupaciones que permite al investigador hacer uso de la información disponible (Figura 2.1).

[3] *http://www.w3c.es/Divulgacion/GuiasBreves/LinkedData*

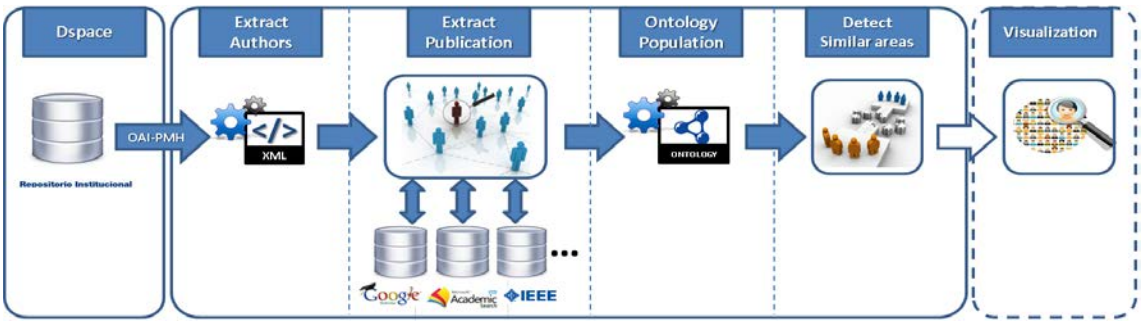


Figura 2.1.- Arquitectura de software del proyecto REDI (CEDIA, 2015)

A continuación, se presenta con más detalle los módulos que conforman la arquitectura del proyecto REDI.

Módulo de extracción de autores

Actualmente, la información de los autores que han realizado publicaciones se encuentra almacenada en un repositorio RDF, que, mediante consultas realizadas a través de SPARQL [4] se extraen los Identificadores de recursos uniformes (URIs) de los autores ecuatorianos de dicho repositorio que son validados para obtener las propiedades y sus valores del recurso como el nombre, apellido, etc. Con la información extraída, es necesario realizar el almacenamiento de la información recolectada en un repositorio temporal, cuando se realiza una actualización de la información se compara los nuevos registros con los registros existentes, para evitar que se realicen ingresos duplicados del autor.

Módulo de extracción de publicaciones

Con la información de los autores ecuatorianos, se inicia el proceso de extracción de sus publicaciones mediante la extracción de información de artículos científicos de diversas fuentes de información como: *Google Scholar*, *Microsoft Academics*, *Scopus*, etc. Cada fuente de información tiene su particularidad por lo que fue necesario realizar un análisis sobre cada una de ellas para identificar el modelo de datos y la forma de acceso.

La extracción la realizaron mediante la generación de búsquedas o consultas sobre las diferentes fuentes de información, como por ejemplo, identificar al autor a través de su nombre y apellido, devolviendo como resultado un listado de autores con su identificador de acuerdo al criterio de búsqueda solicitado; este identificador permite obtener una lista de las publicaciones con información acerca del resumen, palabras claves, y colaboradores de cada publicación.

[4] *https://www.w3.org/TR/rdf-sparql-query*

Integración de publicaciones

Para realizar el proceso de integración de las publicaciones extraídas, los investigadores realizaron algoritmos de desambiguación de los datos de autores y sus publicaciones, que evita almacenar información con errores, para esto compararon las propiedades de un autor y su publicación con la información de otro autor almacenado en una fuente de datos diferente, si estas propiedades coinciden se puede identificar al autor de la misma publicación pero de diferente fuente, por lo que una de estas se elimina y las publicaciones procesadas se ingresan al listado de ese autor y se procede al almacenamiento de la información en un grafo central.

Detección de áreas similares de conocimiento

Este módulo permite detectar áreas similares de conocimiento a partir de las palabras claves que se encuentran en las publicaciones de los autores, para esto, se utilizaron técnicas de minería de datos mediante el uso del servicio *Discovery Research Areas of Knowledge (*KODAR*)*, a través de la aplicación de algoritmos de clustering como K-means que permite agrupar las publicaciones de acuerdo a sus palabras clave, los mismos que tienen que ser etiquetados y categorizados para su visualización.

Visualización de los datos

Con la información almacenada de los autores ecuatorianos y sus publicaciones en el repositorio RDF se utiliza herramientas para su visualización, como por ejemplo:

- Grafos de pastel, para visualizar la información de acuerdo a una categoría;
- Arboles exploratorios, que permite el descubrimiento de entidades;
- Nubes de etiquetas que permiten la búsqueda de acuerdo a agrupaciones de palabras clave;
- Agrupaciones que permite la visualización de publicaciones basados en los autores o geolocalización de ellos, y
- Vista de mapas que permite identificar las áreas de estudio según la localización de los autores del área seleccionada.

Del análisis realizado al proyecto REDI (CEDIA, 2015) se han identificado algunas fortalezas y limitaciones con respecto a los objetivos de este trabajo sobre la manipulación de grandes volúmenes de información y su visualización representados a través de RDF, estas se detallan a continuación:

- Fortalezas (CEDIA, 2015):
    - Localiza la fuente de las publicaciones de los investigadores que hayan sido integrados;

- Visibiliza y promueve la producción científica nacional;
- Ofrece un conjunto de visualizaciones que permiten al usuario navegar y descubrir información relacionada con investigadores ecuatorianos para generar grupos de investigación colaborativa;
- Permite extraer información asociada a la producción científica de cada institución, necesaria para la obtención de estadísticas e indicadores de acreditación;
- Permite al usuario consultar, navegar y exportar información de su interés;
- Permite detectar áreas similares de conocimiento de los investigadores ecuatorianos, y
- Permite agrupar las publicaciones por palabras claves para identificar el universo de investigadores que trabajan sobre cierta área de conocimiento.

- Limitaciones:
  - Las búsquedas son predefinidas por lo que no es posible crear una nueva búsqueda de manera inmediata, por parte de los usuarios;
  - No dispone de herramientas necesarias para realizar búsquedas dinámicas, y
  - No es posible identificar el área de conocimiento en la que se encuentran trabajando los investigadores de manera general a nivel nacional.

3 ESTADO DEL ARTE

Para la definición de la plataforma que permita la visualización dinámica de datos multidimensionales basados en Tecnologías Semánticas (TS) se utiliza el actual repositorio RDF del proyecto REDI y se utiliza conceptos de TS, Modelos multidimensionales y Almacén de datos semántico para el almacenamiento de los datos utilizando el vocabulario en Cubo de datos en RDF, adicionalmente, se define la Arquitectura de Software de la plataforma y finalizando con la generación de un prototipo para realizar la transformación y visualización de la información (Figura 3.1).

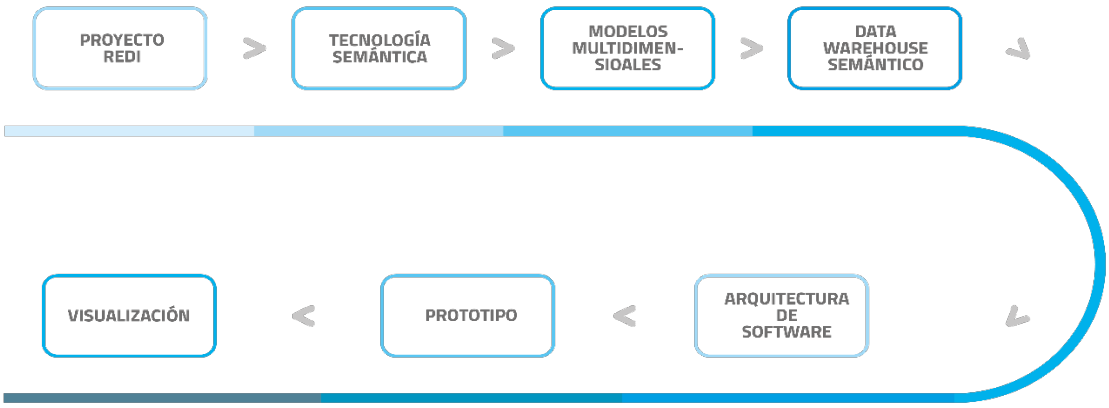


Figura 3.1.- Marco teórico para la generación de la plataforma que permita la visualización de datos multidimensionales basados en Tecnologías Semánticas

3.1 TECNOLOGÍA SEMÁNTICA

En esta sección se detalla el origen y evolución del World Wide Web más conocido como Internet y las Tecnologías Semánticas que se utilizan como base para el desarrollo del presente trabajo de tesis, lenguajes, vocabularios y estándares de la W3C, que permiten estructurar los datos para que puedan ser organizados, interpretados, relacionados y consultados en la web, conceptos necesarios para el desarrollo del presente desarrollo de esta tesis.

World Wide Web – más conocida como WWW, fue creada en 1991 por Tim Berners Lee, denominada la Web 1.0, ésta consiste en un sistema para compartir documentos, enlazar páginas o documentos localizados y trabaja con los protocolos HTTP[5] y HTML[6]. Algunas de las características más importantes de la Web 1.0 es que existen pocos productores de contenidos, los sitios web disponibles son estáticos, es decir solo de lectura, y la actualización de éstos no se realizan en manera periódica (Aghaei, Ali Nematbakhsh, & Khosravi Farsani, 2012).

Posteriormente, evoluciona a la Web 2.0 que se basa en una comunidad de usuarios, una nueva web social, con aplicaciones potentes y sencillas para los usuarios finales, y el nacimiento de los Gestores de contenidos (GC) , que pasan de ser páginas estáticas (HTML) a páginas dinámicas (Aghaei, Ali Nematbakhsh, & Khosravi Farsani, 2012). En las características de la Web 2.0 más importantes se tiene que

[5] *https://www.w3.org/Protocols/*
[6] *https://www.w3.org/html/*

el usuario es primordial para el desarrollo, no es necesario disponer de gran conocimiento para crear sitios web y la actualización de éstos sitios se realizan de manera periódica. Utiliza tecnologías como XHTML[7], XML[8], JavaScrip[9], entre otras (Aghaei, Ali Nematbakhsh, & Khosravi Farsani, 2012)

Actualmente, la web esta evolucionado y a esta se le ha denominado Web 3.0 conocida como la Web Semántica (WS), esta es una extensión de la web actual donde proporciona información bien definida y explícita sobre la información que se está buscando, esta se encuentra de manera estructurada para que pueda ser entendida por la máquina y el usuario, es decir, disponer de la información de manera rápida y eficaz, basándose en conceptos y no en términos, manteniendo los principios de la web actual como la descentralización, compartición, compatibilidad, facilidad de acceso y contribución (Corchuelo, 2007) (Salazar Argonza, 2011). Para lograr el entendimiento entre los usuarios, desarrolladores, etc. es necesario el uso de ontologías, que proporciona vocabularios de clases y relaciones para describir un dominio (Castells, 2003). Más adelante se detalla este concepto.

La diferencia más importante entre la Web Sintáctica (web actual) y la Web Semántica es que la primera es un conjunto de páginas HTML que pueden referenciar a una o más páginas, pero solo puede ser entendido por el usuario, mientras que la Web semántica utiliza etiquetas que permiten que la información publicada pueda ser entendida por la máquina, es decir, la búsqueda en la Web Sintáctica lo realiza en la comparación de cadena de caracteres, mientras que en la Web Semántica, la búsqueda se realiza sobre conceptos (Codina & Rovira, 2006).

Entre las tecnologías más importantes en la Web Semántica, se tiene:

Lenguaje de Marcado Extensible – XML

XML es una recomendación de la W3C, junto con su norma asociada Schema XML, permiten definir tipos de documentos y etiquetas para codificar a esos documentos. Esto es necesario para procesar la información de acuerdo a las necesidades requeridas y proponer diferentes tipos de uso mediante la utilización de diferentes programas informáticos (Codina & Rovira, 2006).

[7] *http://www.w3c.es/Divulgacion/GuiasBreves/XHTML*
[8] *https://www.w3.org/XML/*

[9] *https://www.javascript.com/*

Este meta lenguaje permite estructurar a los documentos con etiquetas sencillas como , <h1>, <h2>, etc. y también con etiquetas un poco más estructuradas y explicitas como <nombre>, <apellido>, <lugar_nacimiento>, etc. Al igual que HTML el texto que se coloca en el documento se encuentra dentro de pares de etiquetas, la primera es de entrada del texto y la final donde indique que el texto concluyó, de acuerdo a los ejemplos anteriores serían *texto1* , <h1>*texto2*</h1>, <nombre>*texto3*</nombre>, <apellido>*texto4*</apellido> (Codina & Rovira, 2006).

Marco de descripción de recursos – RDF

RDF es una recomendación de la W3C, que utiliza XML como un sistema de comunicación que permite la compatibilidad y modelado de metadatos. Proporciona una arquitectura genérica para su utilización y permite la interoperabilidad entre aplicaciones, a través, del intercambio de información y reutilización de metadatos estructurados en el Internet (Senso, 2003). Es decir, define un modelo de datos y establece un mecanismo que permita describir recursos que tengan como principios la multiplataforma y la interoperabilidad de metadatos, el mismo que debe ser neutral con respecto al área de aplicación (software) y flexible para describir cualquier tipo de información (Senso, 2003) (Peis, 2003).

Entre las ventajas de trabajar con RDF se tiene: la capacidad de conciliar contra valores comunes permitiendo la estandarización e intercambio de información entre organizaciones; el almacenamiento de datos se realiza en formatos abiertos y no propietarios; la máquina y la lógica puede interpretar los datos cuando se utilizan ontologías y motores de razonamiento, y un modelo de datos en RDF es infinitamente extensible, entre otras (Williams, 2014).

Para realizar los enlaces a los metadatos en Internet, es necesario utilizar los identificadores de recursos uniformes (URI), que es un sistema de direccionamiento e identificación de recursos únicos, que trabajan a partir de una arquitectura genérica, una tripleta (Figura 3.2) que se encuentra compuesta por un sujeto, predicado y objeto (Castells, 2003) (Peis, 2003).

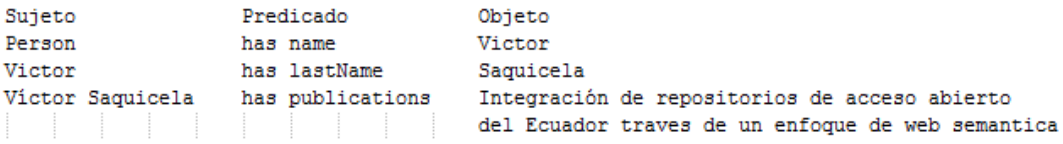


Figura 3.2.- Arquitectura genérica, sujeto, predicado y objeto

Los URI pueden ser nombres demasiados largos (Figura 3.3), y pueden generar algún tipo de dificultad en el momento de la programación, para evitar estos problemas, es necesario realizar una

serialización, que es un mecanismo de abreviación del URI (Figura 3.4) y se asigna a éste, variables denominadas prefijos y direcciones (Castells, 2003). De esta manera se trabaja solamente con las variables y no con toda la dirección del URI.

RDF también permite trabajar con grafos, que ayuda a entender de una mejor manera los enlaces existentes (Figura 3.5).

Variable	URIs
Person	<http://xmlns.com/foaf/0.1/Person>
has name	<http://xmlns.com/foaf/0.1/name>
has lastName	<http://xmlns.com/foaf/0.1/lastName>
has publications	<http://xmlns.com/foaf/0.1/publications>

Figura 3.3.- Ejemplo de utilización de URIs.

```
PREFIX Foaf: <http://xmlns.com/foaf/0.1/>

Foaf:firstName ?Nombre
Foaf:lastName ?Apellido
Foaf:publications ?Publicaciones
```

Figura 3.4.- Ejemplo de serialización

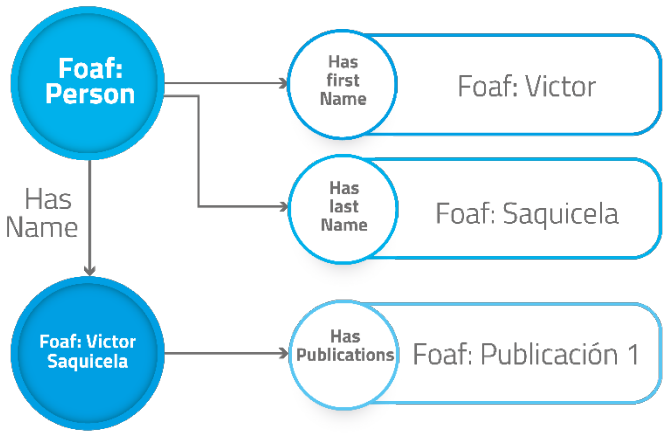


Figura 3.5.- Ejemplo de un grafo en RDF

Pero sería muy poco óptimo describir cientos o miles de recursos mediante la utilización de grafos, por tal razón, es mejor utilizar líneas de texto (Figura 3.6).

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
    xmlns:mm="http://marmotta.apache.org/vocabulary/sparql-functions#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:sesame="http://www.openrdf.org/schema/sesame#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:fn="http://www.w3.org/2005/xpath-functions#">

<rdf:Description rdf:about="http://190.15.141.66:8899/ucuenca/contribuyente/SAQUICELA__VICTOR">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">AUTOMATIC SEMANTIC ANNOTATION</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">LINKED HEALTH DATA CLOUD</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">RDF-IZATION</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMANTICWEB</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">VISUALIZATION</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">WEB SERVICES</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">EPG</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">NLP</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">ONTOLOGIES</subject>
    <subject xmlns="http://purl.org/dc/terms/" rdf:datatype="http://www.w3.org/2001/XMLSchema#string">SEMANTIC ENRICHMENT</subject>
    <owl:sameAs rdf:resource="http://190.15.141.66:8899/ucuenca/contribuyente/SAQUICELA_GALARZA__VICTOR_HUGO"/>
    <foaf:firstName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Víctor</foaf:firstName>
    <foaf:lastName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Saquicela</foaf:lastName>
    <provenance xmlns="http://purl.org/dc/terms/" rdf:resource="http://ucuenca.edu.ec/wkhuska/endpoint/470b898e033c9a126ab3517d55375332"/>
    <foaf:name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Saquicela, Víctor</foaf:name>
    <foaf:nick rdf:datatype="http://www.w3.org/2001/XMLSchema#string">victor saquicela</foaf:nick>
</rdf:Description>
```

Figura 3.6.- Ejemplo de utilización de la codificación en líneas de texto

RDF Schema

RDF Schema (RDFS), es una extensión de RDF que permite describir propiedades y clases de recursos e identificar la relación que existen entre ellos, adicionalmente, introduce los conceptos de subclase y subpropiedad, lo que permite la descripción de las jerarquías de clases y propiedades. También define las propiedades de dominio y el rango de una propiedad. Estos se utilizan para afirmar que objeto de una propiedad específicamente, pertenecen a una o más clases (McBride, 2004).

SPARQL

El protocolo SPARQL y el lenguaje de consultas RDF, forman parte de la familia de recomendaciones de la W3C, es un lenguaje de consulta estandarizado que permite realizar consultas sobre grafos RDF en la web o en almacenes de RDF utilizando diferentes orígenes de datos. Las consultas las realiza de igual manera que el lenguaje RDF basado en tripletas (sujeto, predicado, objeto), con la diferencia que en estas tripletas todas puedes ser variables a consultar (W3C, 2009). Permite realizar operaciones como actualización, inserción, eliminado, copia y cambio de datos de un grafo RDF. (W3C, 2013).

Metadatos

Se parte del conocimiento de que los metadatos son datos acerca de los datos, que pueden denotar cualquier tipo de conocimiento para buscar información sobre la estructura y contenido de cualquier colección de documentos (Samper Zapater, 2005). Los metadatos permiten la interoperabilidad entre diferentes entidades sin importar su tecnología y conocimiento (Samper Zapater, 2005).

Se puede nombrar dos aportaciones importantes de los metadatos:

1. Permite presentar información descriptiva sobre un objeto o un recurso (Tabla 3.1).

DATOS	**METADATOS**
Víctor Saquicela	Nombre
Honorato Loyola 1- 120	Dirección
Cuenca	Ciudad
Azuay	Provincia
Ecuador	País

Tabla 3.1.- Descripción de un recurso

2. Permiten el etiquetado o catalogado

   Como se indicó anteriormente los metadatos solo estructuran contenidos, por lo que es necesario algo que permita estructurar la semántica en un recurso, ese algo se denomina ontología, que tienen como objetivo principal el compartir y reutilizar el conocimiento mediante el uso de un vocabulario común con la posibilidad de ampliar e integrar otras ontologías. Con el uso de las ontologías, las computadoras pueden asemejarse a la interpretación o razonamiento humano mediante la utilización de diferentes vocabularios (Samper Zapater, 2005).

Ontologías

El término Ontología en el campo computacional según Gruber en 1993, indica que: "*las ontologías se definen como una especificación explícita de una conceptualización*"; según Borst en 1997, "*las ontologías se definen como una especificación formal de una conceptualización compartida*". Independientemente, de la variedad de definiciones que se logre encontrar, una ontología siempre incluye un vocabulario de términos y una especificación de su significado indicando las definiciones de los conceptos y como éstos interrelacionan (Samper Zapater, 2005) (Peis, 2003).

De acuerdo a la cantidad y tipo de conceptualización las ontologías se pueden clasificar en (Samper Zapater, 2005):

- Terminológicas: especifican términos para representar conocimiento en el universo de un discurso
- De información: especifican la estructura del almacenamiento en las bases de datos
- De modelado del conocimiento: especifican conceptualizaciones del conocimiento y poseen una rica estructura interna
- Dominio: se representa el conocimiento de un tema específico
- General: se representa el conocimiento general, estructura parte/todo, cuantificación, procesos, tipos de objetos independientes de un dominio en particular.

La ontología posee componentes para representar el conocimiento (Lozano Tello, 2001) como:

- Conceptos: ideas básicas
- Relaciones: interacción y enlace entre los conceptos
- Funciones: tipo concreto de relación
- Instancias: representan objetos determinados de un concepto.
- Axiomas: permiten inferir el conocimiento que no se encuentra explícito.

Cuando se detecta que una ontología existe y se desea crear una similar se debe indicar que "es igual a", o "*same as*" de esta manera se puede realizar el enlace a otra ontología existente (Lozano Tello, 2001).

Lenguaje de Ontologías Web – OWL

OWL, utiliza un sistema de ontologías que permite la definición de conceptos y relaciones de algún dominio, de forma compartida y consensuada (Tello, 2001).

OWL es una recomendación de la W3C, está diseñado para ser utilizado por aplicaciones que necesitan procesar el contenido de la información y no solamente presentar la información a los humanos, ayuda a generar una mayor interoperabilidad a las máquinas sobre el contenido de la web y es soportado por XML, RDF y RDFs. OWL, tiene 3 sub lenguajes como OWL Lite, OWL DL y OWL Full[10].

[10] https://www.w3.org/TR/owl-features/

3.2 MODELO MULTIDIMENSIONAL DE DATOS

En esta sección se detalla qué es y cómo está estructurado un modelo multidimensional de datos, para su organización y utilización de los datos durante el desarrollo del presente trabajo de tesis.

En un Modelo multidimensional de datos (MMD), estos se organizan alrededor de los temas de una organización en particular. El modelo multidimensional está representado por una estructura lógica que utiliza conceptos de cubos, llamado cubos dimensionales o hipercurbos (Figura 3.7), que son un conjunto de celdas que contienen un valor de la medida analizada entre las combinaciones de dichas dimensiones (Dourado, 2014), (Tamayo & Moreno, 2006) (Moreno & Arango, 2007) (Chaudhuri, 1997), (Vaisman & Zimányi, 2014). Los MMD están estructurados por (Posilio Gellida, 2014):

- Hechos: es el objeto a analizar y único, en su mayoría son valores numéricos, se obtienen generalmente por la aplicación de una función estadística como el número de publicaciones de un investigador, pero puede darse el caso de ser valores textuales. Un hecho es de interés primario para la toma de decisión de un negocio, estos datos se almacenan en la tabla de hechos.
- Dimensiones: Son los ejes del cubo, un conjunto de características, miembros o una colección de atributos textuales del mismo tipo que ayudan a identificar, buscar o analizar a los hechos, como por ejemplo, universidad, área de investigación, etc., los diferentes valores se almacenan en la tabla de cada dimensión.
- Jerarquía de dimensiones: las dimensiones se pueden relacionar entre sí mediante jerarquía y niveles. Una jerarquía es un conjunto de miembros de la misma dimensión la cual tiene relevancia el lugar que ocupa cada miembro, permitiendo llegar a mayor detalle al final de la misma. Varios cubos pueden compartir dimensiones y éstos pueden juntarse en multicubos (Figura 3.7).
- Medidas: valores numéricos que permiten realizar una evaluación cuantitativa. Representa el comportamiento del negocio con respecto a una dimensión

Como ejemplo se puede apreciar en la (Figura 3.8) tres dimensiones como son el año de publicación de una publicación, autor de la publicación, y área de conocimiento a la que pertenece dicha publicación y una dimensión como es el número de publicaciones. Las dimensiones representan la granularidad o el nivel de detalle de las medidas para cada dimensión del cubo.

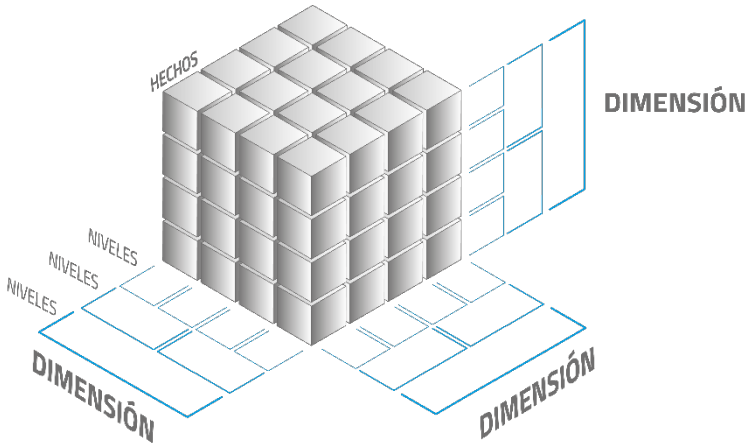


Figura 3.7.- Representación de un modelo multidimensional en un cubo de datos.

Las celdas de un cubo de datos son los hechos, normalmente valores numéricos denominados medidas, que permiten realizar el análisis pertinente, en esta figura, se representa el número de publicaciones realizados por investigador, tiempo e IES a la que pertenece. Estos datos son almacenados de acuerdo a su origen, si son hechos se almacenan en la tabla de hechos y si son dimensiones lo propio en las tablas de dimensiones (Tamayo & Moreno, 2006).

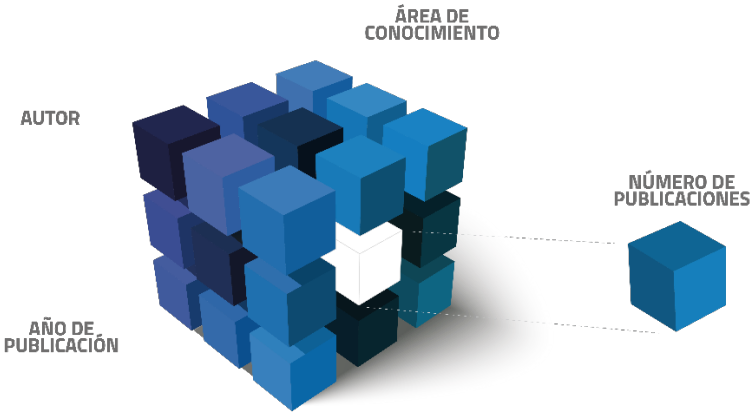


Figura 3.8.- Cubo de dato multidimensionales Publicaciones

3.3 ALMACÉN DE DATOS

En esta sección se detalla qué es y cómo está estructurado un almacén de datos, para que una se dispongan de los datos en el modelo multidimensional, puedan ser guardados en él.

Como definición se puede encontrar que un Almacén de datos (DW) es una *"colección de datos orientada al tema, integrada, temporal y no volátil, usada principalmente para la toma de decisiones"* donde se puede indicar lo siguiente (Vaisman & Zimányi, 2014), (Espinoza, 2010), (Tamayo & Moreno, 2006):

- Colección de datos orientados al tema: la organización de los temas se realiza de acuerdo a su semántica, de manera independiente al uso de las aplicaciones.
- Integrada: Los datos se integran de diferentes fuentes de datos sean estos internos o externos a la organización.
- Temporal: los datos se almacenan de acuerdo a un tiempo específico.
- No volátil: Mantener diferentes versiones temporales de los datos permite recuperar el estado de los mismos en la organización en cualquier momento, de modo que se pueden deshacer efectos indeseados si se han realizado procesamientos erróneos.

Es decir, un DW no es un producto (hardware o software) por lo tanto no es posible comprarlo, éste es creado o se desarrollado desde la propia información de una empresa (Dourado, 2014) (Espinoza, 2010), (Vaisman & Zimányi, 2014). La importancia del análisis de los datos disponibles en una empresa, sin importar el sector al que pertenezcan, es esencial para mejorar su proceso para la toma de decisiones, para que, de ésta manera puedan mantener su ventaja competitiva (Chaudhuri, 1997), (Vaisman & Zimányi, 2014).

Por lo tanto, se puede decir que un DW es una base de datos que utiliza modelos multidimensionales para almacenar datos de interés de una organización en especial, los datos provienen de diferentes fuentes como bases de datos relacionadas, archivos Excel, archivos de texto, etc. los mismos que pasan por un proceso de limpieza y que se encuentran integrados y organizados. Debido a que se trabaja con datos históricos es una excelente fuente para la toma de decisiones (Dourado, 2014), (Espinoza, 2010), (Vaisman & Zimányi, 2014) (Chaudhuri, 1997).

Las bases de datos tradicionales no brindan el soporte necesario para el análisis de datos que requieren las empresas, están diseñadas y ajustadas para apoyar las operaciones diarias de una organización, su principal objetivo es la de garantizar el acceso rápido y de manera simultánea a los datos. Para esto, se

requiere de un procesamiento transaccional y brindar capacidades de control de concurrencia, así como técnicas de recuperación que garanticen la consistencia de datos, estos sistemas son conocidos como Procesamiento transaccional en línea OLTP (Chaudhuri, 1997).

Los sistemas OLTP soportan una gran cantidad de carga transaccional, son altamente normalizadas y tienen un desarrollo muy vago en el momento de ejecutar consultas complejas en la unión de varias tablas relacionales, estas bases de datos operacionales incluyen detalle de los datos, pero no de sus históricos dificultando realizar toma de decisiones con los datos disponibles (Chaudhuri, 1997).

Para corregir estas deficiencias se ha desarrollado un nuevo paradigma denominado Procesamiento analítico en línea conocido como OLAP, el cual soporta el procesamiento de grandes cantidades de consultas particularmente en las analíticas, a diferencia de los sistemas OLTP que se basan en transacciones y no es necesario realizar la normalización de las tablas para realizar dichas consultas (Chaudhuri, 1997).

Los sistemas OLAP, permite ver y analizar los datos de diferentes perspectivas y niveles de detalle, por ejemplo, para definir el nivel de granularidad se utilizan las operaciones de agregación y disgregación, para analizar las dimensiones y creación o eliminación de filtros las operaciones de corte y (Vaisman & Zimányi, 2014), (Posilio Gellida, 2014), como se puede detallar a continuación:

- Agregación*:* Permite agregar medidas a lo largo de una jerarquía de dimensión para obtener medidas con una granularidad más gruesa, es decir, si el cubo inicial indicaba el número de publicaciones de un investigador por trimestre por institución de educación superior, utilizando la operación de agregación, presenta el número de publicaciones de un investigador por trimestres por ciudad.
- Disgregación: es la operación complementaria de la agregación, reduce a agregación de los datos, es decir, de acuerdo al ejemplo anterior, mostraría las publicaciones de un investigador por mes de enero a mayo por institución de educación superior por ciudad.
  Adicionalmente, si el usuario necesita analizar los resultados de los análisis desde otra perspectiva es posible intercambiar la visualización de los datos de acuerdo a los ejes iniciales, es decir, intercambiar (X, Y) a (Y, X), a esta operación se lo conoce intercambio. (Vaisman & Zimányi, 2014).
- Corte: Esta operación permite cortar al cubo original o disminuir las dimensionalidad en dos específicas para el análisis del usuario.
- División: Permite la creación de un subcubo con respecto al original con al menos dos dimensiones.

Con estas mejoras es necesario un modelo de base de datos que soporte el sistema OLAP, permitiendo la generación de un DW o un repositorio que permite el almacenamiento de grandes cantidades de datos de diferentes fuentes internas o externas a la organización, esto es posible debido a que los sistemas OLAP y

el Data Warehouse se basan en modelos multidimensionales de datos (Chaudhuri, 1997), (Vaisman & Zimányi, 2014).

Los DW tienen estructuras de diferentes tipos que permiten representar un modelo multidimensional, a estas estructuras se les conoce como esquemas, los cuales permiten definir como los datos son almacenados, utilizados y representados. El tipo más común de éstos esquemas se lo conoce como: *esquema de estrella* (Figura 3.9), este se encuentra constituido por una sola tabla centralizada de hechos y una tabla por cada dimensión conectada a la tabla central, la tabla de hechos está compuesta por filas que son los identificadores de las tablas externas (dimensiones), y almacena las medidas numéricas, en la tabla de dimensiones corresponde a las columnas de los atributos de esas dimensiones (Dourado, 2014), (Espinoza, 2010) (Chaudhuri, 1997), (Vaisman & Zimányi, 2014).

3.4 VOCABULARIO CUBO DE DATOS EN RDF

En esta sección se detalla qué es y cómo se encuentra estructurado el vocabulario RDF, base fundamental para el desarrollo del presente trabajo de tesis.

El uso principal del vocabulario Cubo de datos en RDF (QB), es compartir y publicar modelos multidimensionales tal como se mencionó en el apartado 3.2 basado en el estándar RDF en la web, a través de la utilización del estándar para el intercambio de datos y metadatos estadísticos (SDMX ISO), el cual es un conjunto de datos estadísticos organizados con valores direccionales y agrupaciones en categorías (rebanadas del cubo) con sus respectivos metadatos que permiten el intercambio, integración, comparación, recolección y procesamiento de la información estadística. El vocabulario es general por lo que se puede utilizar en datos de encuestas, hojas de cálculo, cubos OLAP, etc. (Bayerl & Granitzer, 2015) (Cyganiak, Dave , & Tennison, 2013) (Williams, 2014).

El Vocabulario Cubo de datos en RDF[11] se construyó sobre vocabularios existentes (Cyganiak, Dave , & Tennison, 2013), que permite reutilizar los recursos necesarios, como:

- SKOS[12], para incorporar conceptos de esquemas o estructuras
- SCOVO[13], para el core de la estructura estadística

[11] https://www.w3.org/TR/vocab-data-cube/
[12] *https://www.w3.org/2004/02/skos/*
[13] *http://sw.joanneum.at/scovo/schema.html*

- Dublin Core Terms, para metadatos
- VOID[14]: para acceso a datos
- FOAF[15], para obtener datos de personas y enlazarlas
- ORG[16], para obtener información de organizaciones

El vocabulario Cubo de datos en RDF, se conforma por: observaciones que son datos reales o los valores medidos; una estructura organizacional que permite localizar una observación en un hipercubo mediante su dimensión; metadatos estructurales que ayudan a interpretar la observación una vez ha sido encontrada, como la unidad de medida, si es un valor real o un estimado, como atributos de la observación y metadatos de referencia que describen al conjunto de datos como un todo y permite que se realicen consultas SPARQL sobre ellos (Cyganiak, Dave , & Tennison, 2013).

Adicionalmente, este vocabulario permite trabajar con un subconjunto de un grupo de observaciones manteniendo la misma estructura de un cubo completo de observaciones, a esto se denomina corte. Dentro de un corte se disponen de menos datos y permite agrupar todas las observaciones sobre un determinado indicador que representa una serie temporal de esos valores observados (Cyganiak, Dave , & Tennison, 2013).

Estructura del Vocabulario Cubo de datos en RDF

El vocabulario Cubo de datos en RDF, está compuesto por clases y propiedades propias y otras reutilizadas de los vocabularios en los que se basó su creación (Cyganiak, Dave , & Tennison, 2013). Tal como se puede observar en las tablas (Tabla 3.2, Tabla 3.3, Tabla 3.4, Tabla 3.5, Tabla 3.6, Tabla 3.7, Tabla 3.8, Tabla 3.9, Tabla 3.10) se detallan las clases y propiedades con su propósito, más delante de acuerdo a esta estructura se presentan ejemplos de su uso..

[14] *https://www.w3.org/TR/void/*
[15] *http://xmlns.com/foaf/spec/*
[16] *https://www.w3.org/ns/org#*

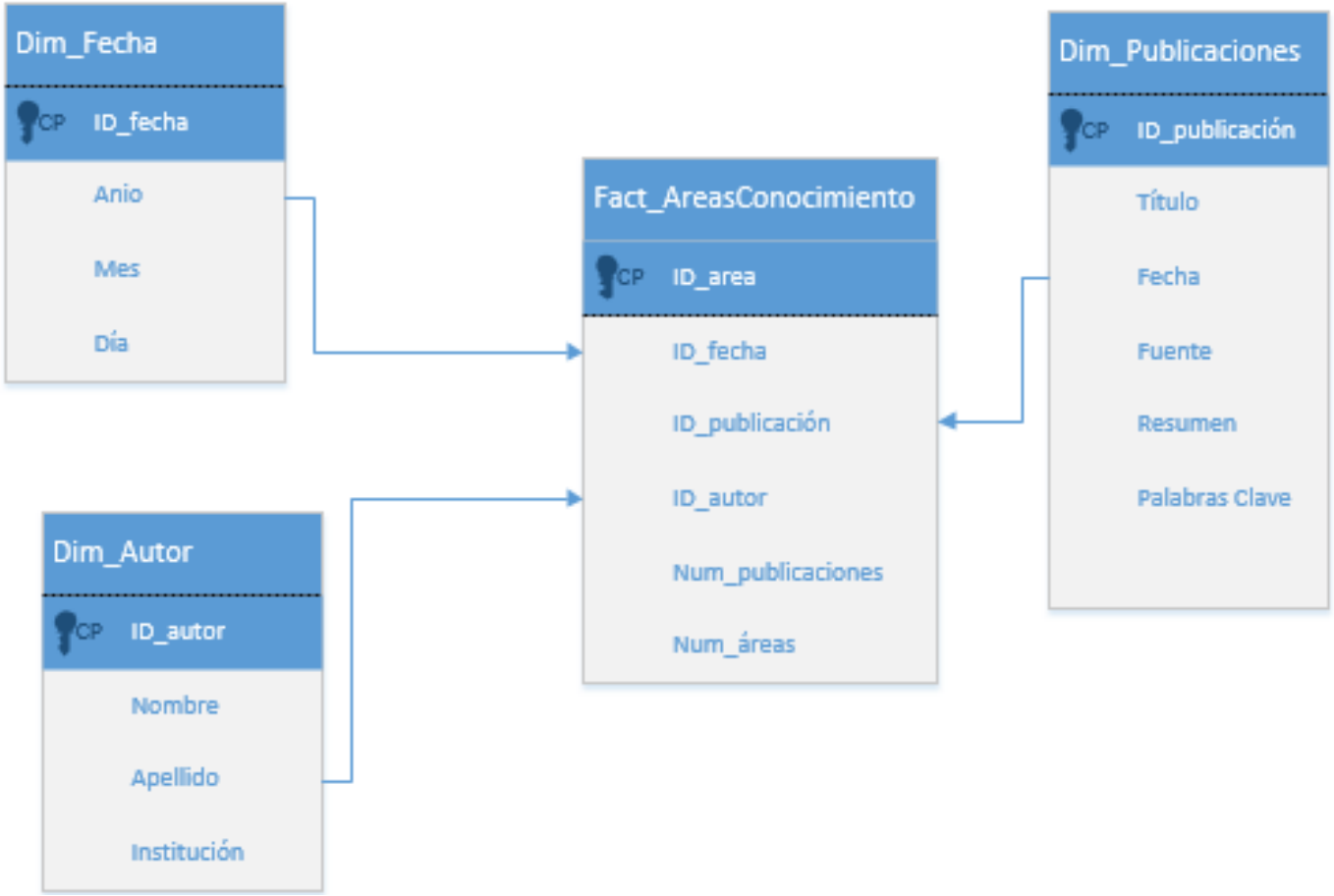


Figura 3.9.- Star Schema Área de conocimiento

3.4.1.1 Dimensiones, atributos y medidas

Clase	Subclase de	Concepto
qb:attachable	N/A	Es una superclase abstracta que puede ser utilizada por todo lo que contenga atributos y dimensiones
qb:ComponentProperty	rdf:Property	Superclase abstracta de todas las propiedades que representen dimensiones, atributos o medidas
qb:DimensionProperty	rdf:ComponetProperty rdf:CodedProperty	Clase de propiedades de los componentes que representan las dimensiones del cubo
Qb:ArributeProperty	qb:ComponentProperty	Clase de propiedades de los componentes que representan los atributos en un cubo
qb:MeasureProperty	qb:ComponentProperty	Clase de propiedades de los componentes que representan un valor medido de un fenómeno que será observado
Qb:CodedProperty	qb:ComponentProperty	Superclase de todas las propiedades de los componentes codificados.

Tabla 3.2.- Clases: dimensiones, atributos y medidas

3.4.1.2 Propiedades de propósito general reusables

Propiedad	**Dominio:Rango**	**Concepto**
qb:measureType	qb:MeasureProperty	Medida de dimensión genérica, indica que medida está dada por la observación

Tabla 3.3.- Propiedades reusables

3.4.1.3 Definiciones de estructura de datos

Clase	**Subclase de**	**Concepto**
qb:DataStructureDefinition	qb:ComponentSet	Define la estructura de un Dataset o slice

Tabla 3.4.- Clases de la estructura de datos

Property	**Domain: Range**	**Concepto**
qb:structure	qb:DataSet qb:DataStrucureDefinition	Indica la estructura de los datos a los que pertenecen los datos del rango
qb:component	qb:DataStrucureDefinition qb:ComponentSpecification	Indica la especificación de un componente que está incluido en la estructura de un dataset

Tabla 3.5.- Propiedades de la estructura de datos

3.4.1.4 Especificaciones de componentes

Clase	Subclase de	Concepto
qb: ComponentSpecification	qb:ComponentSet	Utilizado para definir propiedades de un componente ya sea atributo, dimensión, etc. dentro de la definición de la estructura de datos
qb:ComponentSet	N/A	Clase abstracta de algo que referencia uno o más ComponentPropierties

Tabla 3.6.- Clases, especificación de componentes

Property	Domain: Range	Concepto
qb:componentProperty	qb:ComponentSet qb:ComponentProperty	Indica un ComponentProperty como un atributo o dimensión esperada en un DataSet o un dimensión combinada en un SliceKey
qb:order	qb:ComponentSpecification xsd:int	Indica una prioridad en el orden de los componentes, de descendente a ascendente
qb:componentRequired	qb:ComponentSpecification xsd:boolean	Indica que cualquier propiedad de un componente puede ser o no requerido, true-false
qb:componentAttachment	qb:ComponentSpecification rdfs:Class	Indica el nivel del unión que tenga una propiedad de un componente qb:DataSet; qb:Slice; qb:Observation or qb:MesaruraProperty
qb:dimension	qb:DimensionProperty Subpropiedad de: qb:componentProperty	Es una alternativa a qb:componetProperty para realizar de manera explícita una dimension
qb:measure	qb:MeasureProperty subpropiedad de qb:componentProperty	Es una alternativa a qb:componetProperty para realizar de manera explícita una medida
qb:attribute	qb:AttributeProperty Subpropiedad de qb:componentProperty	Es una alternativa a qb:componetProperty para realizar de manera explícita un atributo
qb:mesureDimension	qb:DimensionProperty Subpropiedad de qb:componetProperty	Es una alternativa a qb:componetProperty para realizar de manera explícita una dimensión de medida

Tabla 3.7.- Propiedades, especificación de componentes

3.4.1.5 Definiciones de rebanadas/cortes

Clase	**Subclase de**	**Concepto**
qb:SliceKey	qb:ComponetSet	Subconjunto de una propiedad de un componente de un DataSet

Tabla 3.8.- Clases, definiciones de slices

Propiedad	**Dominio:Rango**	**Concepto**

qb:SliceKey	qb:Slice qb:SliceKey	Indica que esa slice pertenece a ese slicekey
qb:SliceKey	qb:DataStructureDefinition qb:SliceKey	Indica un slice key que será utilizada para los slices de un dataset

Tabla 3.9.- Propiedad, definiciones de los cortes

3.4.1.6 Definiciones de conceptos

Propiedad	Dominio:Rango	Concepto
qb:concept	qb:ComponentProperty skos:Concept	Da un concepto que será medido o indicado por un ComponetProperty
qb:codeList	qb:CodeProperty owl:unionOf(skos:ConceptSchema skosCollection qb:HierarchicalCodeList)	Da un codList asociado a un CodedProperty

Tabla 3.10.- Propiedades, conceptos

Estas clases y propiedades detalladas en forma de tabla se puede presentar de manera gráfica para un mejor entendimiento (Figura 3.10).

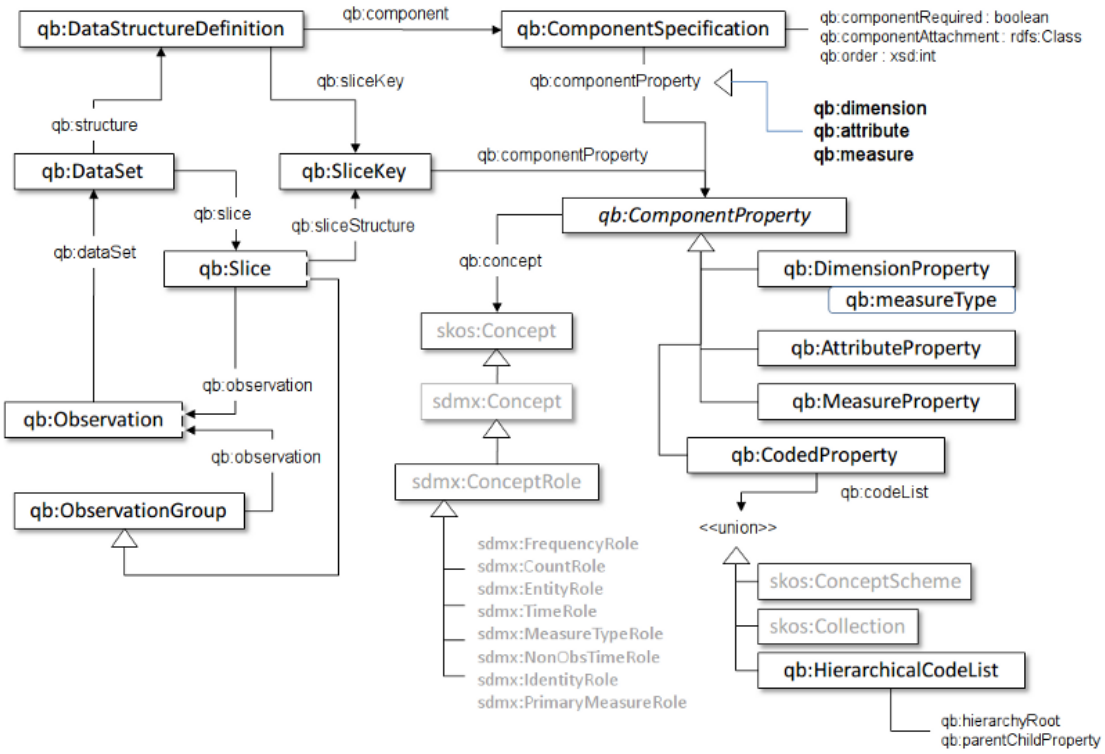


Figura 3.10.- Estructura del vocabulario Cubo de datos en RDF (W3C, 2014).

3.5 VOCABULARIO CUBO DE DATOS EN RDF PARA OLAP

En esta sección se menciona qué es el vocabulario RDF para OLAP, el mismo que dispone de algunas mejoras con respecto al vocabulario RDF, pero debido a que aún no es un estándar de la W3C, no se utiliza en el desarrollo del presente trabajo de tesis, se nombra por la relación que existe con el vocabulario anterior.

El vocabulario Cubo de datos en RDF para OLAP – QB4OLAP, es una extensión del vocabulario Cubo de datos en RDF (QB), que ofrece facilidades para realizar el análisis de datos estadísticos, superando las limitaciones de QB sobre las operaciones OLAP en el modelo multidimensional, es decir, si puede realizar las operaciones de agregación, disgregación, corte y división (Vaisman & Zimányi, 2014) (Etcheverry, Vaisman, & Zimányi, 2014).

Para trabajar con QB4OLAP (Figura 3.11), es necesario definir dos tipos de tripletas para establecer el esquema y las instancias del cubo, de esta manera se estructura al cubo considerando las dimensiones, niveles y medidas, considerando las jerarquías de las dimensiones que se propone en OLAP. El trabajo realizado en cubos con QB es totalmente compatible con QB4OLAP, es decir, se puede continuar el desarrollo sin afectar las aplicaciones realizadas en ellos.

Este vocabulario presenta una distribución de clases, propiedades, niveles, miembros como extensión del estándar QB, y se lo identifica con el prefijo "*qb4o*" (Vaisman & Zimányi, 2014), (Etcheverry, Vaisman, & Zimányi, 2014), (Etcheverry & Vaisman, 2012).

Se puede concluir que estos dos vocabularios RDF y QB4OLAP pueden ser utilizados para representar modelos multidimensionales, que permiten mediante la utilización de clases y propiedades, clasificar y organizar la información disponible a través del uso de tripletas, para que esta información pueda ser visualizada en un momento dado. Adicionalmente, es necesario indicar que el vocabulario QB4OLAP brinda el soporte adecuado a las operaciones ofrecidas por OLAP, pero debido a que no es un estándar internacional como es el vocabulario Cubo de datos en RDF – QB, no fue utilizado en el desarrollo del presente trabajo de tesis.

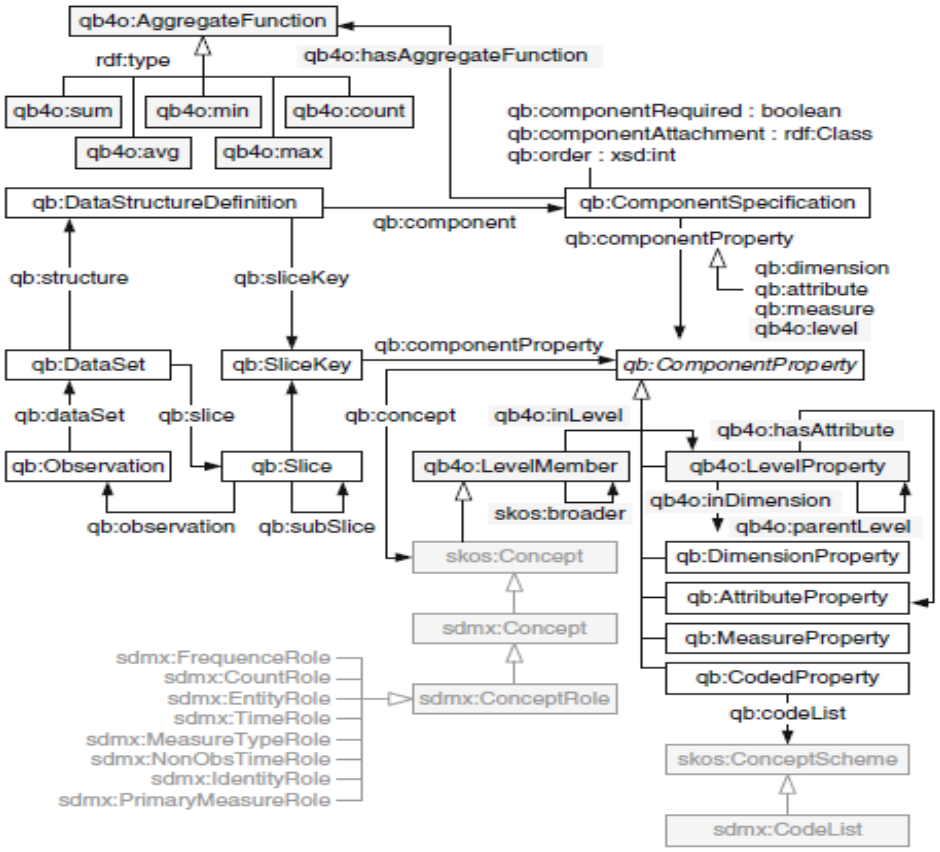


Figura 3.11.- Estructura del vocabulario QB4OLAP (Vaisman & Zimányi, 2014).

3.6 ARQUITECTURA DE SOFTWARE DISPONIBLES PARA LA ADAPTACIÓN DE TECNOLOGÍA SEMÁNTICA

En esta sección se indica qué es y cuáles son los tipos de Arquitectura de Software existentes, necesario para el desarrollo del presente trabajo de tesis, debido a que ayuda a estructurar el proceso de visualización de datos multidimensionales, a la que se incorporará Tecnología Semántica.

El concepto de Arquitectura de Software nace como parte fundamental del éxito para el desarrollo de sistemas o programas grandes y complejos. Es una disciplina en desarrollo y cada vez más aceptada por parte de los investigadores y la industria como una práctica profesional (Clements, Garlan, Reed, Nord, & Stafford, 2003). De acuerdo a (Bosch, 2004) define a la Arquitectura de Software (AS) como "*una organización fundamental de un sistema que incluye componentes y sus relaciones con el medio ambiente y los principios que guían su diseño y evolución*". Según

(Pressman, 2010) la arquitectura de software es "*la estructura o estructuras del sistema, lo que comprende a los componentes del software, sus propiedades externas visibles y las relaciones entre ellos*", de acuerdo a estos dos conceptos se puede indicar que la arquitectura de software es la estructura del software que se desea construir y la interrelación que existe entre los componentes que lo conforman.

Se puede identificar ciertos beneficios de su uso como un marco de trabajo para satisfacer requerimientos: como una base técnica para el diseño, estimación de costos y administración de procesos; como base de una efectiva reutilización y un análisis consistente y dependiente (Kruchten, Obbink, & Stafford, 2006), similar a lo establecido por el Grupo de Trabajo 2.10 de la Federación Internacional de Tratamiento de la Información (IFIP), que determinó cinco sub áreas de la arquitectura de software como son: el diseño de la arquitectura, análisis, realización, representación y economía (Perry & Wolf, 1992).

Estilos o taxonomía de la Arquitectura de Software

Se identificó de acuerdo a citas de varios autores un número pequeño de estilos o taxonomías de Arquitectura de Software, entre estos se tiene:

- Arquitectura centrada en los datos: Se basa en la integración de los datos, utilizando un repositorio central donde los componentes acceden a éste, para realizar operaciones de agregación, eliminación y edición (Figura 3.12) a los datos que se encuentran almacenados en él (Pressman, 2010), (Reynoso, 2004).

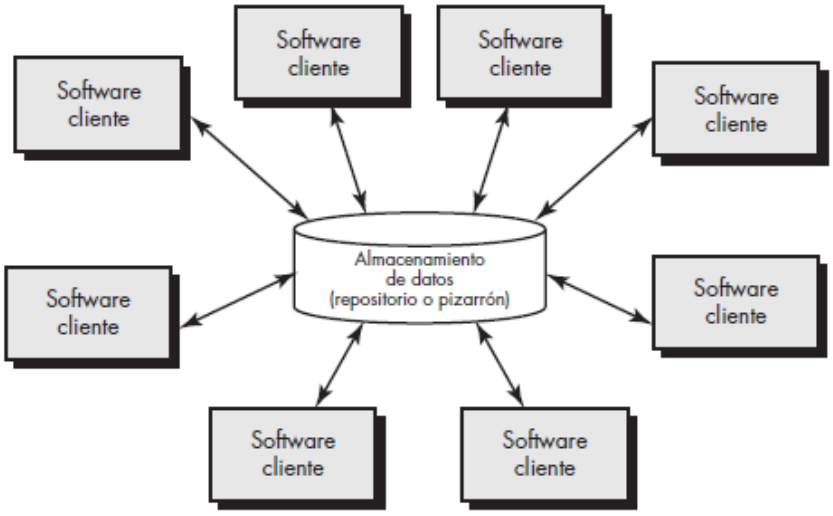


Figura 3.12.- Arquitectura de Software centrada en los datos (Pressman, 2010).

- Arquitectura de flujo de datos: Se basa en la reutilización y modificabilidad, se utiliza cuando los datos de entrada se van a convertir en datos de salida, utilizan un patrón denominado tubo y filtro, los datos pasan por los filtros a través de los tubos que los interconectan y permite el acceso al siguiente filtro, se encuentran conectados por una sola línea de transformaciones que se denomina lote secuencial (Figura 3.13) (Pressman, 2010), (Reynoso, 2004).

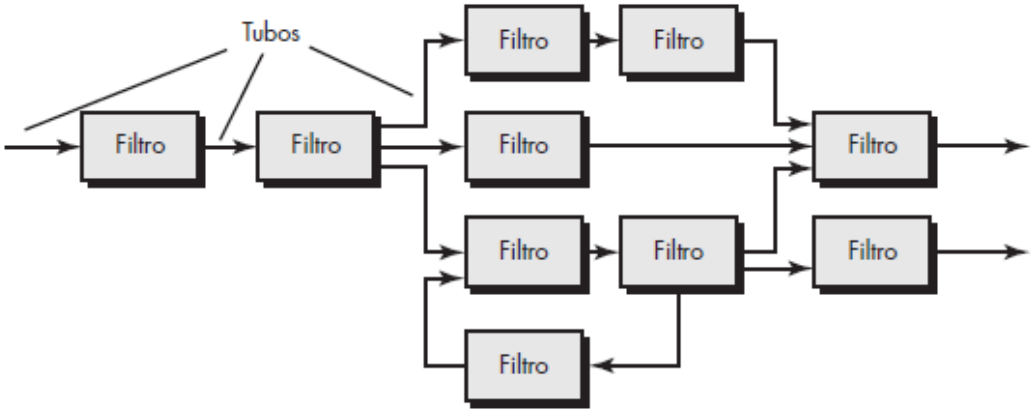


Figura 3.13.- Arquitectura de flujo de datos (Pressman, 2010).

- Arquitectura de llamar y regresar: Se basa en la modificabilidad y la escalabilidad, esta estructura dispone de un programa principal que llama a subprogramas para realizar tareas específicas (Figura 3.14) esta modalidad también se utiliza a través del uso remoto de subprogramas (Pressman, 2010), (Reynoso, 2004).

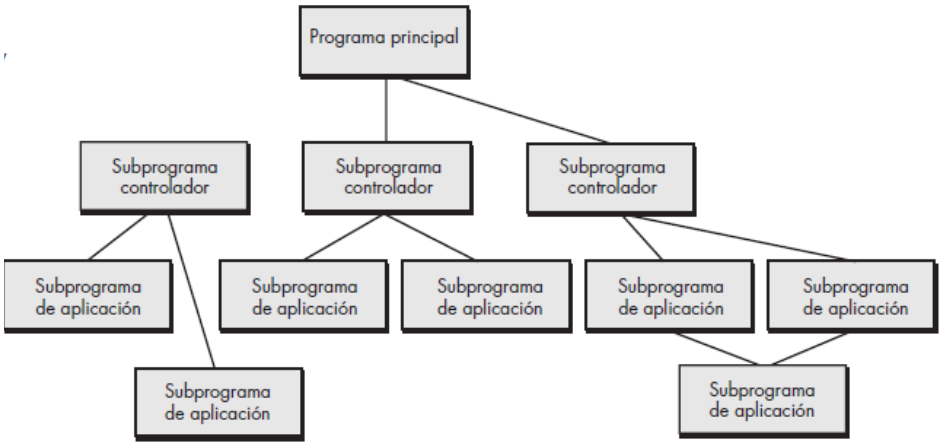


Figura 3.14.- Arquitectura llamar y regresar (Pressman, 2010).

- Arquitectura en capas: Una capa es considerada como una unidad lógica, y física, la primera tiene bien definidos sus propios objetivos y funcionalidades y establece claramente sus dependencias y colaboradores hacia el sistema donde se asignan sus roles y responsabilidades; la segunda indica que físicamente puede estar en diferentes maquinas o servidores, a esto se le conoce como niveles (Cardacci, 2015), (Moquillaza, Vega Huerta, & Guerra Grados, 2010). La relación entre una capa y un nivel puede ser de uno a varios o de varios a varios todo depende del diseño arquitectónico de la solución planteada (Cardacci, 2015).

  La arquitectura en capas, se basa en una organización jerárquica, es decir, la capa anterior provee servicios a la capa inmediatamente superior, los conectores están definidos por los protocolos que permiten la interacción. La capa externa se preocupa de las operaciones de la interfaz de usuario, la capa interna de la interfaz del sistema operativo, y las capas intermedias provee servicios de soporte y funciones del software de aplicación (Pressman, 2010), (Reynoso, 2004). El objetivo principal de la arquitectura en capas (Figura 3.15) consiste en separar la capa lógica del negocio de la capa lógica del diseño, es decir,

separa la capa donde se encuentran las fuentes de los datos, de la capa de negocios y esta a su vez de la capa de presentación al usuario (Moquillaza, Vega Huerta, & Guerra Grados, 2010).

Según (Acosta Gonzaga, Alvarez Cedillo, & Gordillo Mejía, 2006) se puede clasificar a la arquitectura de software de acuerdo al número de capas en:

- Arquitecturas de una capa: Aquí se encuentran programas sencillos que como su nombre lo indica están constituidos por una sola capa, como los procesadores de texto, compiladores que no necesitan de acuerdo a su naturaleza mantener acceso a la red de la empresa, internet, etc. no sobrecargan la red ni deben esperar un turno para comunicarse con el servidor. Debido a su sencillez no es necesario el uso ni mantenimiento de protocolos.
- Arquitectura de dos capas: se conforma por tres partes: cliente, servidor y protocolo de comunicación entre las dos capas, a esto se le conoce como una arquitectura cliente/servidor, donde en el lado del cliente se almacenan los programas para la interfaz gráfica o aplicaciones de red, que son ejecutados de manera rápida, mientras, que en el lado del servidor se almacena la lógica del negocio, donde se encuentran bajo la seguridad que este le ofrece y les brinda el acceso a todos sus recursos.
- Arquitectura de tres capas: Esta se utiliza cuando es necesario almacenar la información en una base de datos especializada para evitar problemas de integración cuando varios usuarios soliciten al servidor realizar tareas de manera simultánea (Acosta Gonzaga, Alvarez Cedillo, & Gordillo Mejía, 2006). Esta arquitectura permite separar las tres capas que lo conforman, datos, negocio y presentación. El almacenamiento de la información en una base de datos indica una mejora sustancial con respecto a la arquitectura de dos capas, puesto permite realizar índices para obtener los datos de una manera óptima, realizar respaldos, crear redundancia, y permite acceder a los datos por otras aplicaciones o servicios del propio sistema (Acosta Gonzaga, Alvarez Cedillo, & Gordillo Mejía, 2006).
- Arquitectura de N capas: Se basa en realizar la distribución de los roles y responsabilidades de manera jerárquica para resolver de una manera efectiva los problemas encontrados. Los roles permiten determinar el tipo y la manera en la que las capas van a interactuar con otras y las responsabilidades indicaran claramente las

funcionalidades que desarrollaran cada una (Moquillaza, Vega Huerta, & Guerra Grados, 2010) (De la Torre Llorente, Zorrilla Castro, Ramos, & Calvarro, 2010). En esta arquitectura es indispensable desarrollar objetos reutilizables para que en un futuro se puedan utilizar nuevamente, permitiendo que el sistema inicial sea la escalable, creciendo a medida de las necesidades mediante la incorporación de nuevos módulos, los cuales pueden estar distribuidos en varias máquinas, e interactúen entre si utilizando estándares predefinidos y marcos de comunicación (Acosta Gonzaga, Alvarez Cedillo, & Gordillo Mejía, 2006).

Según (Moquillaza, Vega Huerta, & Guerra Grados, 2010), los tipos de capas de una arquitectura de software, pueden ser:

- Capa de datos: Esta capa se encarga de realizar el acceso a los datos, la función fundamental de esta capa es almacenar y recuperar toda la información del sistema. Aquí se desarrollan las conexiones entre el servidor y las fuentes de información, esta capa físicamente puede estar en la misma máquina de la capa superior, pero se recomienda que de acuerdo a su nivel de complejidad se encuentre físicamente en otro servidor.
- Capa de Negocio: puede ser una o varias capas intermedias, responsables del procesamiento de las aplicaciones del sistema. Contiene los objetos que son reutilizables y establece métodos para realizar los cálculos, variables y se encuentra en constante comunicación con la capa de datos para realizar el almacenamiento, edición, eliminación, consultas, etc. Se denomina capa de negocio porque aquí se establecen, desarrollan y ejecutan todas las reglas que deben cumplirse, esta capa procesa todos los requerimientos solicitados por su capa superior y puede encontrarse físicamente en otro servidor con respecto a la capa de datos.
- Capa de presentación: Es la capa que se presenta a los usuarios del sistema, captura todas sus necesidades y requerimientos mediante formularios, etc. para pasar a la capa de negocio a que ejecuten las reglas establecidas.

Estas arquitecturas de acuerdo a las necesidades del software a desarrollar se pueden utilizar y combinar a conveniencia incluyendo las mejores características de cada una.

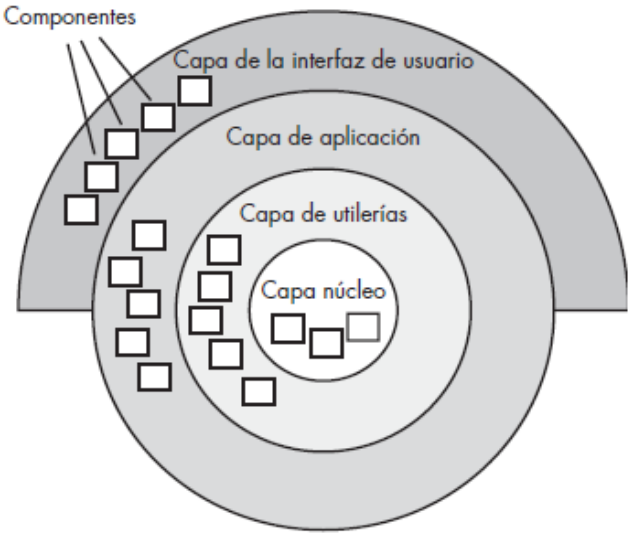


Figura 3.15.- Arquitectura en capas (Pressman, 2010).

3.7 HERRAMIENTAS PARA LA VISUALIZACIÓN DE MODELOS MULTIDIMENSIONALES BASADAS EN TECNOLOGÍA SEMÁNTICA

En esta sección se detallan las aplicaciones que permiten la visualización de modelos multidimensionales basados en Tecnología Semántica y su funcionamiento, que, de acuerdo a sus características, se determina la idónea para utilizarla en el desarrollo del prototipo planteado.

Existen aplicaciones que permiten la visualización de información estadística basadas en tecnologías semánticas haciendo uso del vocabulario Cubo de datos en RDF (QB), entre esas se encuentran:

OpenCube Toolkit

Esta aplicación web integra componentes y herramientas de acceso abierto, dispone un SDK para desarrollar aplicaciones personalizadas y generar funcionalidades de bajo nivel como acceso a datos compartidos, registro de usuarios y monitorización de los datos. Permite trabajar con diferentes

fuentes de información de entrada como CSV, TSV, RDB, JSON-stat, R2RML que mediante un mapeo transforman a formato RDF. (OpenCubeProject, 2013).

Entre las características principales de esta aplicación se tiene (OpenCubeProject, 2013):

- Expande los datos fuentes:
    - Dado un cubo inicial, permite buscar información compatible en *Datos enlazados* en la web para mejorar al cubo;
    - Enlaza dicha información con la información compatible;
    - Dado un cubo inicial, con *n* dimensiones, genera 2n-1 cubos teniendo en cuenta todas las posibilidades de *n* dimensiones;
    - Dado un cubo con una jerarquía de dimensiones, se crea nuevas observaciones para todos los atributos de la jerarquía, y
    - Crea un nuevo cubo expandido fusionando dos cubos compatibles.
- Explora los datos (OpenCubeProject, 2013):
    - Mediante la utilización de plantillas permite la administración de los metadatos sobre Cubo de datos en RDF
    - Visualiza Cubo de datos en RDF, *slices*.
    - Permite operaciones OLAP
    - Permite trabajar con el programa estadístico “R”.
    - Permite la visualización geoespacial de los datos.

Payola

Es una herramienta que permite realizar visualizaciones de información estadística basada en Cubo de datos en RDF, admite como fuente de información de entrada varios conjuntos de datos basado en RDF. En la plataforma se registra dos tipos de usuarios el invitado y el registrado, teniendo este último grandes ventajas sobre el primero como la creación de sus propias instancias y filtrado de datos para la visualización. Payola también es conocida como una herramienta colaborativa que permite compartir con otros usuarios fuentes de datos, análisis realizados, ontologías personalizadas para realizar la visualización, etc. (Helmich, 2013).

OLAP2DataCube

Esta herramienta es un plugin de la plataforma OntoWiki, que permite la visualización y edición de información utilizando los principios de Datos enlazados. Esta plataforma permite convertir una base de datos relacional (no RDF) con contenido estadístico en QB, más conocido por los autores como triplificación. Con la información ya mapeada la plataforma realiza la visualización de los datos estadísticos en formato QB (Helmich, 2013).

Tabels

Esta aplicación web permite convertir datos que se encuentran en tablas a formato RDF, debido a que dispone de su propio lenguaje específico de dominio (DSL) para realizarlo, posteriormente permite realizar el mapeo a Cubo de datos en RDF para finalmente visualizar la información estadística. Una de las limitaciones de esta herramienta, es que se necesita la participación del usuario para editar el script de mapeo para realizar la vinculación de los datos y éste debe tener grandes conocimientos de programación (Helmich, 2013).

CubeViz

Esta herramienta se basa en la aplicación web OntoWiki[17], y en principios de *Datos enlazados y QB*, permite definir una estructura visualización personalizada (DCV), donde el usuario puede filtrar datos para cada dimensión, y establecer los criterios adecuados para realizar la división (slice) del cubo para la visualización (Rivera Salas, y otros, 2012), (Helmich, 2013). Adicionalmente, permite generar un link al grafo obtenido de la visualización para que este resultado sea compartido con otros usuarios. Una de las grandes limitaciones de esta herramienta es que permite trabajar con hasta dos dimensiones para la visualización, si se desea más dimensiones la plataforma no puede procesarlas (Helmich, 2013).

Otras herramientas de visualización

Existen otras herramientas que permiten realizar la visualización de información estadística pero estas no necesariamente hacen uso del Vocabulario Cubo de datos en RDF, como por ejemplo: Geo

[17] http://ontowiki.net/

Globe, VisualBox, ViDaX, LodVis, etc. que permiten trabajar con fuentes de información basados en RDF. La obtención de la información la realizan en su mayoría mediante consultas SPARQL, permiten identificar la zona geográfica de los datos, obtenidos, estas aplicaciones pueden ser de escritorio o web, entre otras características (Helmich, 2013), debido a esta limitación de no trabajar con QB, estas herramientas no son consideradas parte de este trabajo de tesis.

4 TRABAJOS RELACIONADOS

Como se mencionó en el capítulo 2, actualmente, se dispone de una plataforma denominada REDI que presenta las publicaciones realizadas por autores ecuatorianos, identifica áreas de conocimiento en las que se encuentra trabajando un investigador, genera nubes de palabras claves para identificar las áreas de investigación, entre otras. Con esta plataforma se ha tratado de cubrir en parte el problema general encontrado, para que los investigadores ecuatorianos puedan encontrar pares académicos para realizar investigación.

Las búsquedas que se realizan actualmente en la plataforma son preestablecidas, si bien cumplen su función de visualizar la información deseada, no permiten al usuario realizar búsquedas dinámicas. Esta limitación de la plataforma REDI pretende ser resuelta mediante el desarrollo del presente trabajo de tesis incorporando nuevas herramientas de búsquedas utilizando modelos multidimensionales basado en Tecnología Semántica.

En este capítulo se detalla el trabajo realizado por diferentes autores sobre el proceso de transformación de datos que se encuentran actualmente en RDF a Cubo de datos en RDF, la visualización de modelos multidimensionales, la arquitectura de software de la plataforma y el almacenamiento de la información en el Data Warehouse, todas estas herramientas y metodologías basadas en Tecnología Semántica.

4.1 PROCESO DE TRANSFORMACIÓN DE RDF A CUBO DE DATOS EN RDF

La transformación de los datos que se encuentran en RDF a Cubo de datos en RDF - QB trabaja muy de cerca con la visualización de la información estadística, puesto que, algunas aplicaciones permiten que en el proceso de visualización se realice el mapeo de diferentes formatos como CSV, XLS, PX, KML, OLAP, RDB, RDF, para visualizar los datos en QB (Helmich, 2013), (Rivera Salas, y otros, 2012), (Martin, y otros, 2015), (OpenCubeProject, 2013).

De acuerdo a (Kämpgen & Harth, 2011), un Modelo de Datos Multidimensional - MDM está conformado por uno o varios cubos relacionados entre sí por sus temáticas, un número de datos de este conjunto elegidos para la integración y análisis que definen un MDM. El cubo de datos en un MDM es definido únicamente por la instancia del cubo de datos (***qb:DataSet***), que está relacionada

por la propiedad ***qb:structure*** que pertenece a la definición de la estructura de los datos (***qb:DataStructureDefinition***). Adicionalmente, se utilizan propiedades que describen elementos multidimensionales como *rdfs:label* para definir un nombre único del cubo y *rdfs:comment* que permite incorporar una descripción detallada del cubo.

En *QB* un hecho es una instancia de ***qb:Observation*** las cuales son conectadas al ***qb:DataSet*** a través de la propiedad ***qb:dataset*** mientras que las Dimensiones son predefinidas en la estructura del conjunto de datos a través de las instancias de ***qb:DimensionProperty***. Las estadísticas se realizan a través de medidas, un cubo puede realizar varias operaciones como suma, mínimo, promedio, conteo, etc. que indican como el hecho es medido a través de la propiedad ***qb:ComponentProperty*** y ***qb:MeasureProperty*** (Kämpgen & Harth, 2011).

Según (Helmich, 2013), indica que la aplicación web *Tabels* permite como fuente de información de entrada conjuntos de datos en RDF para realizar el proceso de transformación a QB. Generó un script de conversión, pero éste se desarrolló con errores, ya que el mismo se encontraba vacío, por lo que la transformación la generó manera manual todo el script de transformación basándose en consultas SPARQL, encontrando de esta manera limitaciones con esta herramienta.

Según (OpenCubeProject, 2013), la herramienta OpenCube Toolkit permite realizar el mapeo de diferentes tipos de fuentes de información incluyendo RDF a QB, basándose en el ciclo de vida de Datos estadísticas enlazados (LODS) (Figura 4.1).

Según (Helmich, 2013) e (Kämpgen & Harth, 2011) indican que es necesario establecer las especificaciones del mapeo que el usuario va a utilizar, es decir, se define la estructura de los datos y la relación que existen entre ellos. Se debe identificar las dimensiones y sus propiedades, las medidas a presentar, establecer rangos, definir componentes, etc. (Figura 4.2), seguidamente se debe seleccionar un patrón de búsqueda para la visualización.

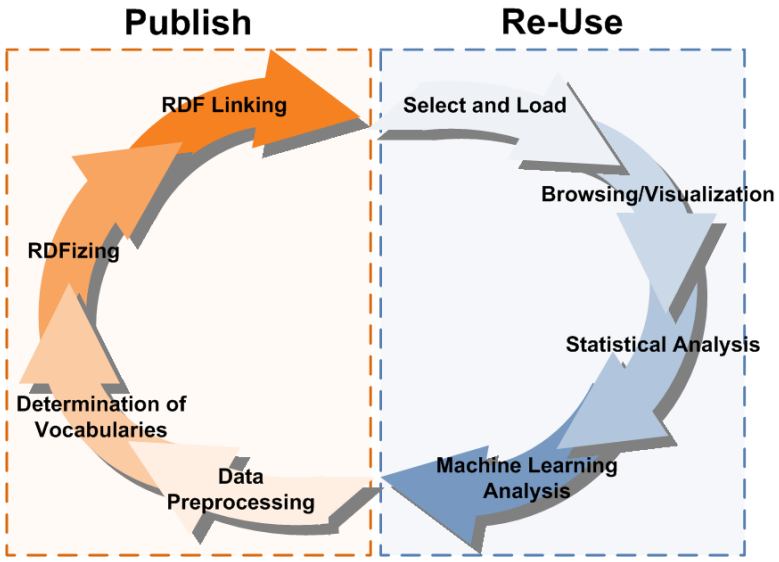


Figura 4.1.- Ciclo de vida de datos enlazados estadísticos (OpenCubeProject, 2013)

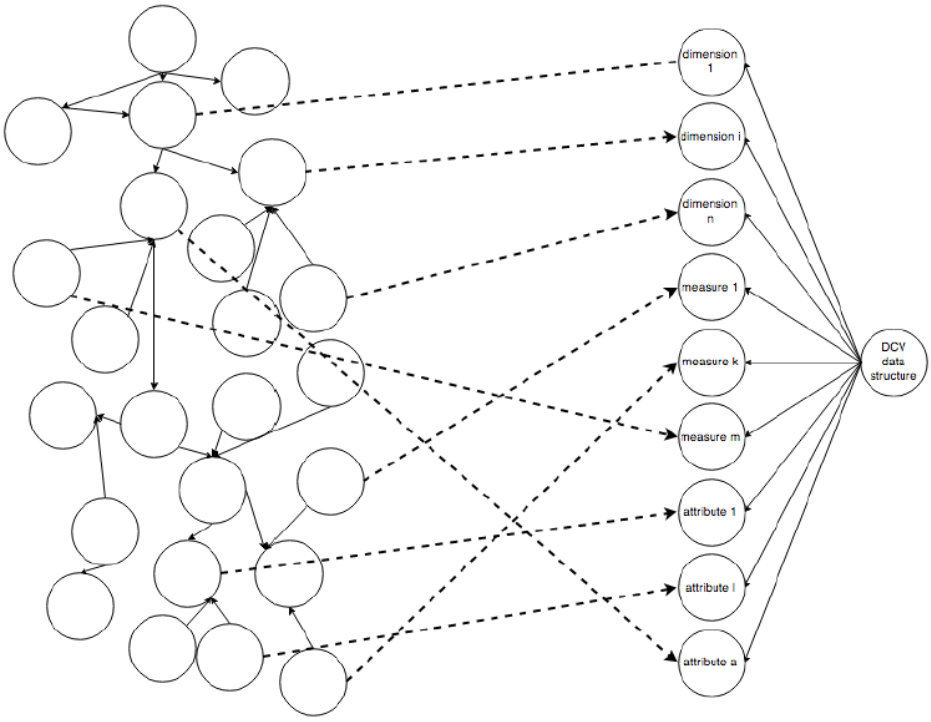


Figura 4.2.- Definición de la estructura de los datos (Helmich, 2013)

Con esta información disponible, se genera las consultas en SPARQL, tantas como sean necesarias, para acceder a los datos que finalmente generan un nuevo grafo en QB (Figura 4.3). Esta visualización según (Helmich, 2013), se puede realizar en la plataforma web Payola[18] que permite, entre sus características, utilizar diferentes fuentes de datos como RDF para generar la visualización en QB; permite instalar varios tipos de plugins para que sea posible realizar el mapeo; dispone de varios componentes que facilita al usuario en definir la estructura de los datos, y permite compartir el resultado del mapeo con otros usuarios, entre otras (Figura 4.4). Una de las limitaciones más importantes de esta herramienta es que no permiten el procesamiento de grandes cantidades de datos.

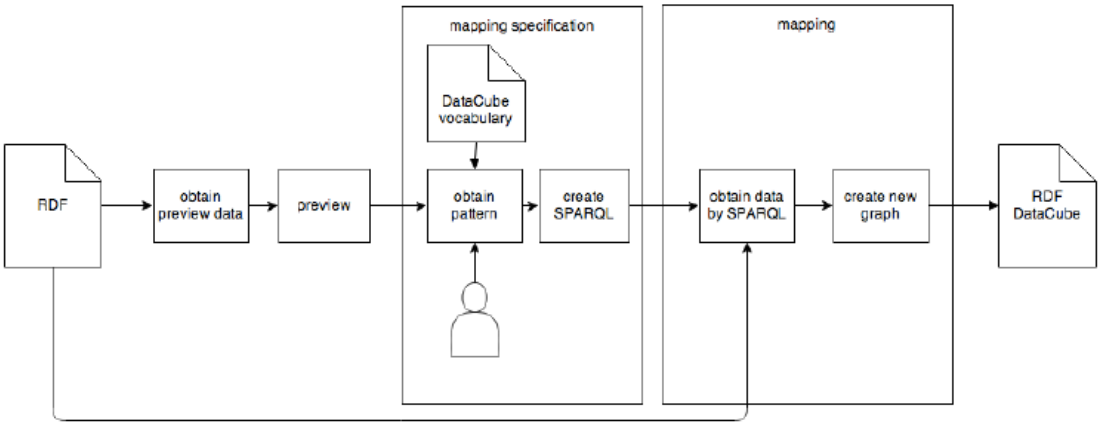


Figura 4.3.- Sistema propuesto para la visualización y generación de un grafo Cubo de datos en RDF basado en RDF (Helmich, 2013).

También es posible transformar datos que se encuentren en bases de datos relacionales a QB, primero deberá pasarlos a RDF mediante la utilización del estándar RDF Lenguaje de mapeo- *R2RML* (Souripriya , Seema , & Richard , 2012). Segundo, se propone trabajar sobre el estándar R2RML y realizar un mapeo de los datos multidimensionales existentes a cubos OLAP, denominado Lenguaje de mapeo multidimensional - M2RML utilizando el vocabulario del cubo de datos – *QB* (Ghasemi, 2014).

[18] https://github.com/payola/Payola

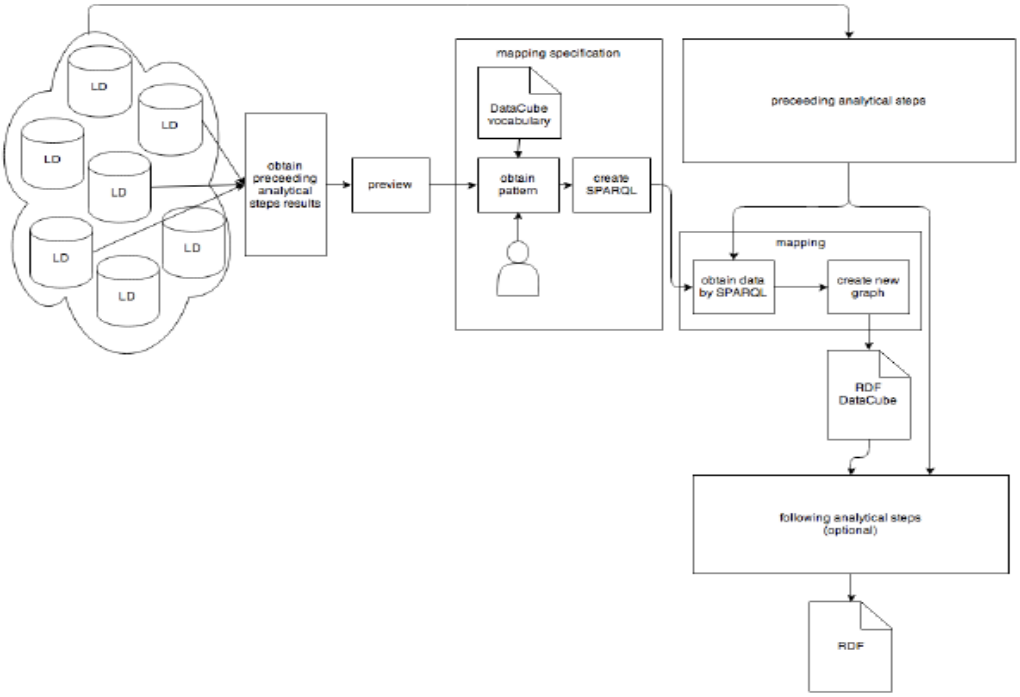


Figura 4.4.- Proceso de mapeo dentro de la plataforma Payola (Helmich, 2013).

(Ghasemi, 2014) separa en dos partes el mapeo, el primero lo utiliza para definir las clases abstractas y generales basados en R2RML, que puede ser utilizado para cualquier tipo de origen de datos y la segunda para asignar datos multidimensionales de acuerdo a *QB.* De acuerdo al lenguaje de mapeo propuesto M2RML existen algunas clases y propiedades específicas para trabajar, entre las más importantes se tiene (Tabla 4.1):

Estructura QB	Estructura M2RML
qb:DataSet	m2r:DataSetMap;
qb:DataStructureDefinition	m2r:PropertyMapDSD;
qb:Slice	m2r:PropertyMapSlice;
qb:ComponentSpecification	m2r:ComponentSpecificationMap
qb:DimensionProperty	m2r:PropertyMapDimProperty
qb:MeasureProperty	m2r:PropertyMapMeasureProperty
qb:AttributeProperty	m2r:PropertyMapAttributeProperty

Tabla 4.1.- Estructura del lenguaje de mapeo M2RLM (Ghasemi, 2014)

4.2 ALMACÉN DE DATOS SEMÁNTICO

De acuerdo al trabajo realizado por (Nebot, Berlanga, Pérez, Aramburu, & Pedersen, 2009), diseñaron un almacén de datos semántico - SDW para almacenar fuentes de datos basado en ontologías semánticas, los autores proponen una metodología que consiste en:

- Identificar los diferentes tipos de fuentes de datos; expresarlos en formato *RDF* o *XML*, mediante la generación de tripletas (Sujeto, Predicado, Objeto),
- Definir la estructura de los datos; dimensiones, hechos, medidas y niveles,
- Definir las operaciones *OLAP* que se pretenden realizar; y almacenar los datos con notaciones semánticas en un *SDW*.

Los autores definen cuatro fases para determinar un modelo conceptual para la generación de un *SDW* (Figura 4.5*)*, la primera, consiste en establecer la estructura de la Ontología Multidimensional integrada – *MIO*, la segunda, permite identificar y extraer las ontologías necesarias para la correcta elaboración de los cubos de datos; la tercera, es un validador de los cubos generados; la cuarta, es un análisis de los cubos con la información que se encuentra en el SDW mediante la utilización de las ontologías seleccionadas.

Según (Bellatreche, Selma, & Berkani, 2013), para diseñar un SDW, es necesario establecer cinco pasos: análisis de los datos; diseño conceptual; diseño lógico, proceso *ETL*, desarrollo y diseño físico (Figura 4.6).

En el análisis de los requerimientos se identifican las ontologías necesarias para la definición y conexión de los datos, aquí se establece que información es importante para ser almacenada en el SDW. En el diseño conceptual, se define una ontología para el almacén de datos DWO, y se comprueba si existen errores mediante la revisión de las clases e instancias propuestas y las relaciones que existan entre ellos.

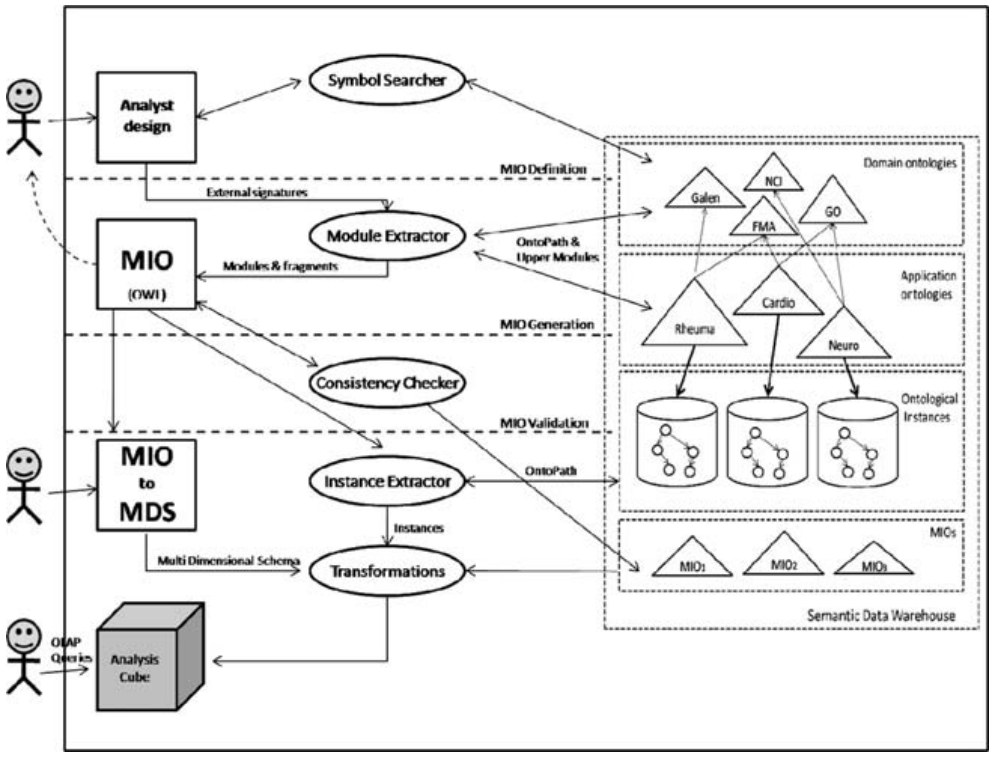


Figura 4.5.- Framework para diseñar un SDW (Nebot, Berlanga, Pérez, Aramburu, & Pedersen, 2009)

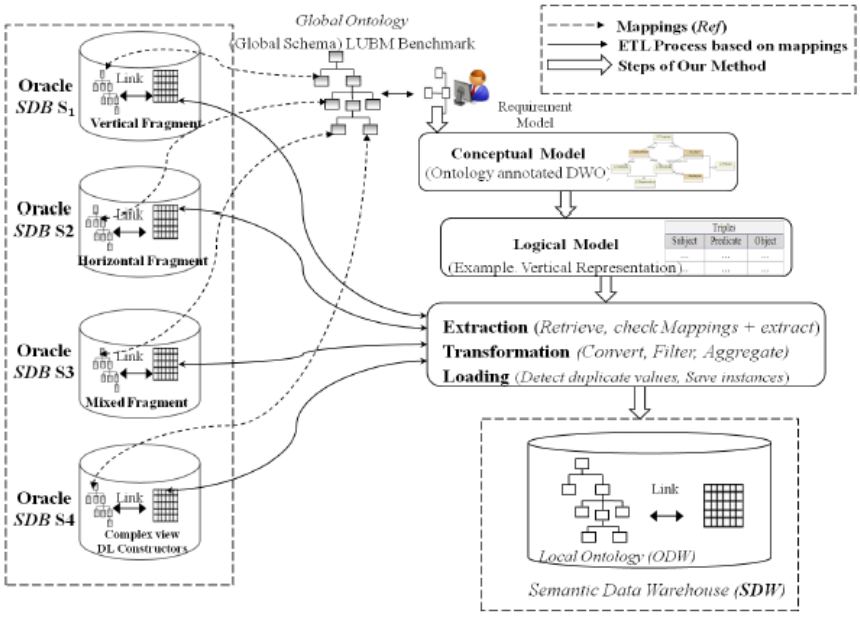


Figura 4.6.- Diseño de un SDW (Bellatreche, Selma, & Berkani, 2013)

El diseño lógico, consiste en pasar el DWO a un modelo relacional el mismo que es analizado y validado. El proceso de ETL, consiste en poblar el SDW mediante operaciones como extracciones, uniones, almacenamiento, agregaciones, etc. El desarrollo y diseño físico, permite crear las reglas para realizar el almacenamiento con la ventaja que en el SDW se puede realizar varios diseños de almacenamiento de manera independiente.

4.3 VISUALIZACIÓN DE MODELOS MULTIDIMENSIONALES BASADAS EN TECNOLOGÍAS SEMÁNTICAS

Según (Rivera Salas, y otros, 2012) (Helmich, 2013), para realizar el proceso de visualización en la plataforma CubeViz[19] debe disponer de la información en formato Cubo de datos en RDF, y utilizar el componente de visualización desarrollado exclusivamente para este propósito como una extensión de OntoWiki.

Según (Rivera Salas, y otros, 2012), (Martin, y otros, 2015), para iniciar el uso con esta plataforma de visualización primero se necesita definir la estructura de los datos (**qb:Data StructureDefinition)**, el conjunto de datos (**qb:DataSet**), agregar los componentes necesarios, a estos componentes se definen tipos de componentes (**qb:ComponentSpecification**); las propiedades de los tipos que pueden ser **qb:DimensionProperty, qb:MeasurementProperty y qb:AttributeProperty** (Figura 4.7).

CubeViz, realiza el procesamiento de las consultas SPARQL en la estructura del cubo, mas no en las observaciones para evitar demoras en la generación del grafo resultante, para la visualización tiene la opción de presentar grafos de barras, circulares, líneas. La plataforma permite realizar operaciones matemáticas como suma, promedio, el valor mínimo y máximo.

También permite la instalación de plugins para integrar más operaciones matemáticas y opciones de generación de nuevos grafos. Según (Helmich, 2013), una de las más grandes limitaciones de esta herramienta es que permite trabajar con hasta dos dimensiones para la visualización de información estadística.

[19] *https://github.com/AKSW/cubeviz.ontowiki*

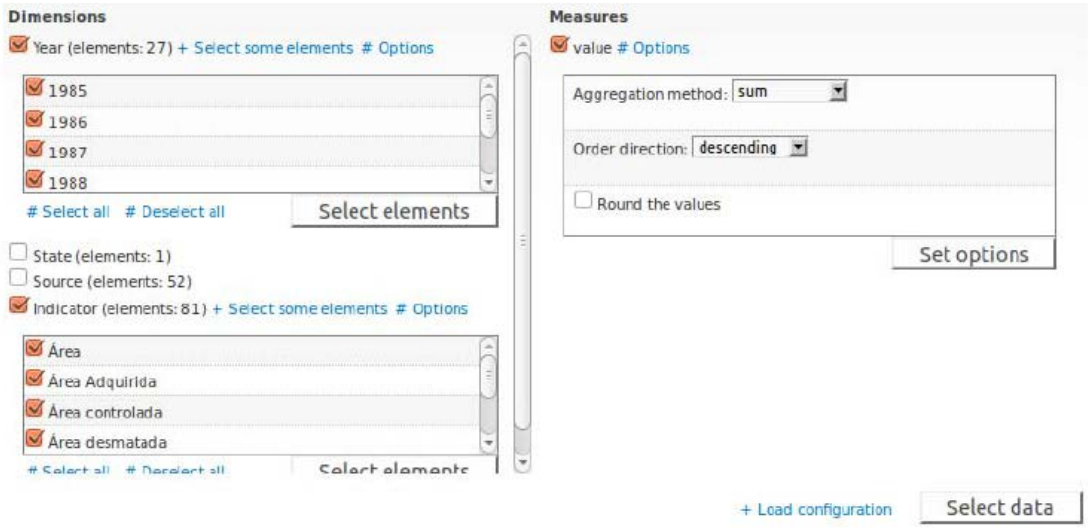


Figura 4.7.- Plataforma CubeViz, selección de la estructura de datos (Rivera Salas, y otros, 2012)

De acuerdo a (Helmich, 2013) para realizar un proceso general de visualización basándose en QB, en primera instancia, se debe disponer como entrada del proceso un grafo RDF y un vocabulario de cubo de datos para definir la estructura de los datos, con esta estructura el usuario puede definir la especificación del mapeo que se va a realizar en el paso siguiente de la implementación del mapeo para la obtención del resultado que sería la visualización en Cubo de datos en RDF (Figura 4.8).

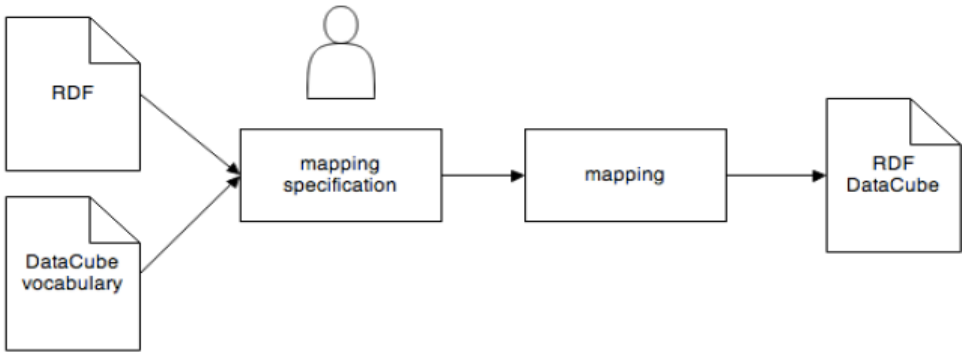


Figura 4.8.- Estructura general para la visualización de información en RDF a Cubo de datos en RDF (Helmich, 2013)

El autor (Helmich, 2013), realiza una comparación de las diferentes herramientas/plataformas que permiten la visualización de un conjunto de datos en QB (y otras fuentes), relacionando las características que éstas comparten (Tabla 4.2).

Tool	Cube support	Cube mapping	Mapping input	Analysing	Create datasets	Application type	Sharing	Custom vocabulary	Visualize	Faceted browsing	Large datasets
Payola	Y	Y	LD	Y	Y	W	Y	N	Y	N	N
OLAP2DataCube	Y	Y	R	N	N	W	N	Y	Y	Y	Y
Tabels	Y	Y	A	N	Y	W	N	Y	Y	Y	Y
CubeViz	Y	N	-	N	N	W	Y	-	Y	Y	Y
Geo Globe	N	-	-	N	-	W	N	-	Y	Y	N
Visualbox	N	-	-	N	-	W	Y	-	Y	N	Y
ViDaX	N	-	-	N	-	D	N	-	Y	Y	Y
LodVis	N	-	-	Y	-	W	N	-	Y	Y	Y
Rhizomer	N	-	-	N	-	W	N	-	N	Y	N
Sgvizler	N	-	-	N	-	W	Y	-	Y	N	N
Exhibit	N	-	-	N	-	W	Y	-	Y	Y	*
Explorator	N	-	-	N	-	W	N	-	Y	Y	Y
Tabulator	N	-	-	Y	-	W	N	-	Y	N	Y

Tabla 4.2.- Comparación de características comunes entre herramientas que permiten la visualización de información estadística, donde Y= Si; N= No; LD= Datos enlazados; A= Formato Arbitrario; R= Datos relacionales; W= aplicación web; D= aplicación de escritorio (Helmich, 2013)

De los datos presentados por (Helmich, 2013), se va a desestimar las herramientas que se encuentran en la segunda parte de la (Tabla 4.2), como se indicó en el capítulo tres, estas herramientas no permiten el uso de QB para su visualización. Adicionalmente, se incorpora en la tabla el análisis de una nueva herramienta denominada *OpenCube Toolkit* descrita en el apartado 3.7 (Tabla 4.3).

Tool	Cube support	Cube mapping	Mapping input	Analysing	Create datasets	Application type	Sharing	Custom vocabulary	Visualize	Faceted browsing	Large datasets
OpenCube Toolkit	Y	Y	DF	Y	Y	D	Y	N	Y	Y	Y
Payola	Y	Y	LD	Y	Y	W	Y	N	Y	N	N
OLAP2DataCube	Y	Y	R	N	N	W	N	Y	Y	Y	Y
Tabels	Y	Y	A	N	Y	W	N	Y	Y	Y	Y
CubeViz	Y	N	-	N	N	W	Y	-	Y	Y	Y

Tabla 4.3.- Herramientas para la visualización de Cubo de datos en RDF (Helmich, 2013), donde Y= Si; N= No; LD= Datos enlazados; A= Formato Arbitrario; R= Datos relacionales; W= aplicación web; D= aplicación de escritorio, y DF= Diferentes Fuentes.

También existe la posibilidad de partir directamente desde la información que se encuentre en QB, se inicia con la selección de la estructura de datos, conjunto de datos, tipo de componentes y sus propiedades, posteriormente se procede a realizar las consultas SPARQL necesarias para que finalmente se presente el grafo en QB.

4.4 ARQUITECTURA DE SOFTWARE BASADA EN TECNOLOGÍA SEMÁNTICA

De acuerdo a (Kämpgen & Harth, 2011), la arquitectura propuesta en su trabajo para realizar el mapeo desde un MDM a QB, consta de dos partes, la primera consiste en extraer, transformar y cargar (ETL) los datos que son almacenados en un Data Warehouse dados por los URIs, y la segunda consiste en la ejecución donde se realizan las consultas a los datos una vez se haya realizado el proceso ETL.

Como se puede apreciar el autor maneja una arquitectura en capas y combina en su segunda capa con una arquitectura de flujo de datos. De acuerdo a (Kämpgen & Harth, 2011), en la capa 1 el usuario define los conjuntos de datos a ser integrados y permite recuperar la información del MDM para que puedan ser consultados a través de SPARQL. Tomando como base los principios Datos enlazados, el sistema recupera los archivos RDF relevantes y los almacena en repositorios RDF. En la capa 2, con el resultado de las consultas se puebla el MDM. En la capa 3 a través de XMLA[20] que proporciona al usuario una interfaz amigable para que pueda consultar los datos en el MDM. A través del servidor OLAP Mondrian[21], serializa los datos obtenidos para incorporar los hechos en la tabla de hechos y las dimensiones en la tabla de Dimensiones con su miembros y valores. En la capa 4, a través del lenguaje de consultas para datos multidimensionales (MDX[22]) se realizan las consultas necesarias sobre los datos almacenados en el DW para la visualización de los mismos.

Según (Vdovjak & Houben, 2001) y (Aggoume, Bouramoul, & Kholladi, 2016), la integración de información provenientes de diferentes fuentes ha tenido un crecimiento exponencial, por ello y con ayuda de Tecnología Semántica han desarrollado una arquitectura para realizar la integración semántica.

De acuerdo a (Vdovjak & Houben, 2001), proponen una arquitectura de cinco capas, en la primera capa fuente , se encuentran las diferentes fuentes de información que se requieren integrar. En la segunda capa de instanciación XML, puede ser opcional y ser una sola capa con la anterior, esta sirve para tratar los datos mediante el meta lenguaje XML, la tercera capa *XML2RDF*, esta permite realizar el mapeo de XML a RDF, de acuerdo al modelo conceptual de los datos originales. La cuarta capa de inmediación/inferenciación, este es un componente central de la arquitectura propuesta, donde se encuentran las reglas que permiten inferir los resultados que son utilizados por la siguiente capa, finalmente la quinta capa de aplicación se incorporan todo tipo de aplicaciones para la utilización de los datos, y permite el acceso desde cualquier infraestructura.

(Aggoume, Bouramoul, & Kholladi, 2016), a diferencia del autor anterior, proponen una arquitectura conformada por 3 capas, la primera capa fuente de datos, se encuentran las diferentes fuentes de información a ser tratadas, en la segunda capa de encapsulamiento es una capa intermedia

[20] *http://xmla.org/*
[21] *http://mondrian.pentaho.com/*
[22] *https://msdn.microsoft.com/es-es/library/bb500184.aspx*

entre la capa mediadora y la de la fuente de información, que realiza las consultas en su capa predecesora en SPARQL y presenta la siguiente capa, finalmente la tercera capa, de mediación, la más importante de esta arquitectura propuesta, conformada por dos componentes, el primero permite dividir la consulta inicial en subconsultas y son enviadas a la segunda capa para que sean ejecutadas, y el segundo, permite reconstruir las consultas para presentar una sola respuesta al usuario, permitiendo mejorar el desempeño incorporando nuevos conceptos, para las consultas.

De acuerdo a (Ghasemi, 2014), propone trabajar sobre una arquitectura general para publicar y acceder a datos multidimensionales enlazados en la web basado en *QB.* Se encuentra conformada por tres capas:

- **Fuente de datos**, en esta capa se almacenan los datos multidimensionales pueden ser un repositorio RDF, Cubos OLAP, etc.
- **Mapeo y transformación,** en esta capa se realiza la transformación o mapeo, donde se debe establecer el vocabulario base escogido para la transformación (DCV*);* utilizar un lenguaje de mapeo (*M2RML*), determinar las especificaciones del mapeo (*M2RML y DCV*) y un script para realizar la conversión.
- **Destino*:*** en esta capa se genera el gráfico como resultado del proceso de transformación o mapeo, con el cual pueden interactuar de diferentes maneras, adicionalmente, el resultado se puede almacenar en repositorios RDF.

4.5 CONCLUSIÓN

Con el análisis realizado sobre las herramientas de visualización (Tabla 4.3) se pudo identificar la idónea para utilizarla en el prototipo planteado, siendo esta la herramienta OpenCube ToolKit, que permite manejar grandes cantidades de datos, interactúa con diferentes fuentes de datos, trabaja con *n*-número de dimensiones de acuerdo a la información que se desea visualizar, permite manipular al cubo de datos para que éste sea expandido o explorado, a diferencia de las otras herramientas presentadas. Adicionalmente, se realizó el análisis de los procesos para realizar la transformación de datos que se encuentren en RDF a QB, y se adoptarán de acuerdo a lo explicado por los autores (Helmich, 2013) e (Kämpgen & Harth, 2011). Como resultado de este análisis se identificó características o parámetros que permiten el cumplimiento del objetivo de este trabajo de tesis (Tabla 4.4).

Características del REDI:

- Búsquedas mediante palabras clave (BPC), como autores, publicaciones, áreas de conocimiento, esto permite a la herramienta identificar variables que el usuario puede utilizar para obtener la información necesaria;
- Detección de áreas similares de conocimiento (DASC), esto permite identificar las áreas de conocimiento en las que se desarrolla un investigador.
- Ubicación geográfica por área de conocimiento (UGAC) permite detectar las áreas de conocimiento en las que se encuentran trabajando los investigadores por la ubicación geográfica en el Ecuador.

Características comunes:

- Tecnología Semántica (TS), permite identificar si la herramienta se ha desarrollado utilizando TS.
- Visualizar la información (VI), indica si la herramienta utilizada permite visualizar la información requerida.

Características de las herramientas analizadas:

- Modelos multidimensionales basados en QB que permite indicar si la herramienta utiliza el objeto de estudio de este trabajo.
- Visualización de la información basada en QB (VQB): permite indicar si la herramienta permite visualizar la información basándose en modelos multidimensionales basados en Tecnología Semántica.

Herramientas analizadas	**BPC**	**DASC**	**UGAC**	**TS**	**QB**	**VI**	**VQB**
Google Scholar	Si	No	No	No	No	Si	No
Research Gate	Si	No	No	Si	No	Si	No
Proyecto REDI actual	Si	Si	Si	Si	No	Si	No

Tabla 4.4.- Análisis de las herramientas estudiadas

Como se puede apreciar en la (Tabla 4.4) ninguna de las herramientas analizadas en el presente trabajo de tesis presenta todos los parámetros necesarios para resolver el problema planteado, por tal motivo, se vio la necesidad de mejorar la plataforma existente del proyecto REDI donde se deben incorporar los parámetros faltantes para resolver el problema planteado. Para esto se necesita una

Arquitectura de Software que incorpore Tecnología Semántica para su funcionamiento, adicionalmente, herramientas de visualización y procesos necesarios para realizar la transformación de RDF a QB, que otras herramientas/aplicaciones estudiadas no disponen, de esta manera con la implementación de éstos parámetros se logrará resolver el problema planteado en este trabajo de tesis (Tabla 4.5).

Herramientas analizadas	**BPC**	**DASC**	**TS**	**QB**	**VI**	**VQB**
Proyecto REDI propuesto	Si	Si	Si	Si	Si	Si

Tabla 4.5.- Propuesta de la nueva plataforma para el proyecto REDI

5 ARQUITECTURA DE SOFTWARE PLANTEADA

En este capítulo se indica el tipo de arquitectura que se emplea para el desarrollo de la plataforma definida, los componentes a utilizar y la relación que existen entre ellos.

5.1 ARQUITECTURA DE LA PLATAFORMA PROPUESTA

De acuerdo al análisis realizado en el estado del arte y las necesidades acorde al marco de este trabajo de tesis, se ha establecido que la arquitectura de software a utilizar es la arquitectura basada en capas y flujo de datos. La arquitectura de capas está compuesta por la capa de datos, capa de aplicaciones, capa de presentación y la capa cliente; la arquitectura de flujo de datos está compuesta por el primer filtro para realizar la extracción de datos SPARQL; esta información que sirve de entrada para el segundo filtro para realizar la transformación de información RDF a QB, este resultado sirve de entrada para el tercer filtro, para genera el SPARQL para almacenamiento en el SDW en la capa de datos que permite la extracción de datos QB, que es utilizado por la capa de presentación (Figura 5.1).

Dentro de la capa de datos se encuentran el actual repositorio *Apache Marmotta*[23] donde se almacenan las tripletas del proyecto REDI, y el Almacén de datos semántico - SDW, que sirve para el almacenamiento de los datos transformados a QB.

[23] http://marmotta.apache.org/index.html

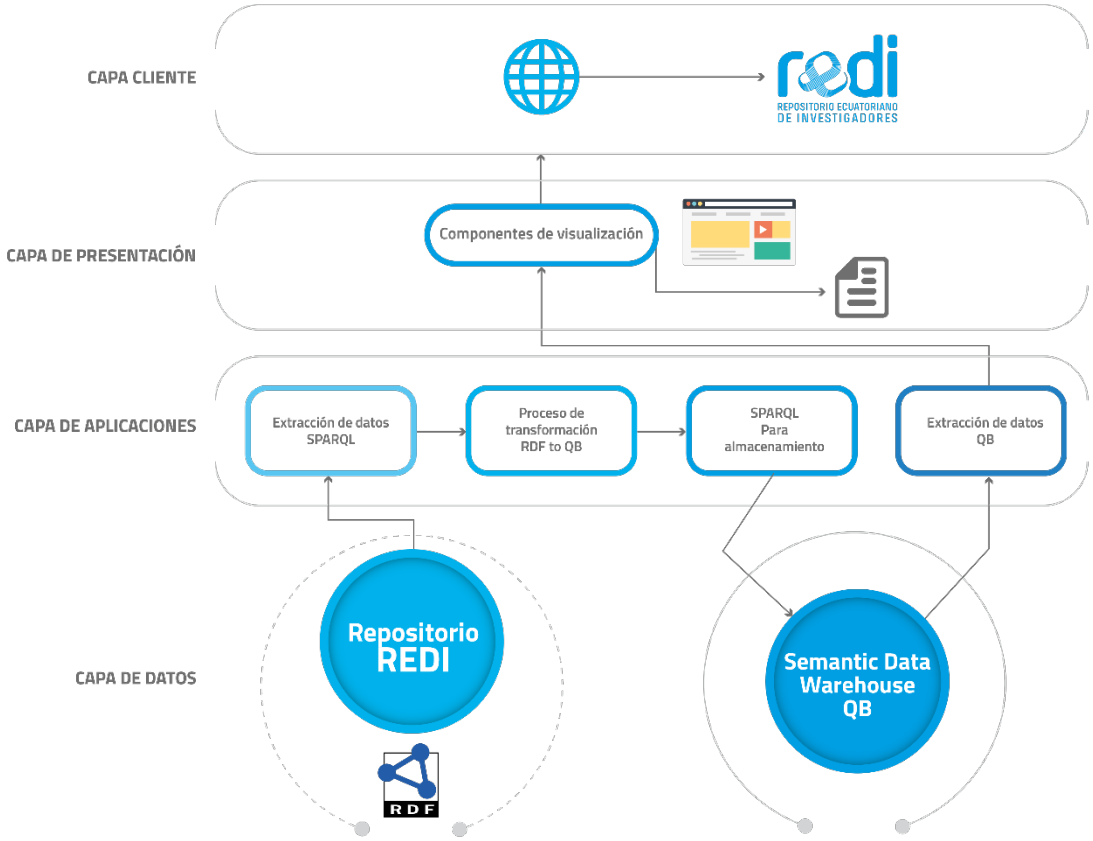


Figura 5.1.- Arquitectura REDI propuesta

En la capa de aplicaciones, se trabaja con una arquitectura de flujo de datos que está conformado por cuatro componentes, inicia el primer filtro con la extracción de los datos del repositorio *Apache Marmotta*, el resultado de esta extracción, ingresa al segundo filtro para realizar el proceso de transformación de RDF a QB, esta información sirve de entrada para el tercer filtro, para generar el SPARQL de almacenamiento en el SDW, para finalmente, ser extraídos en el último filtro para realizar la visualización respectiva. En la capa de presentación, se encuentran los componentes para realizar la visualización de los datos multidimensionales basado en Tecnologías Semánticas. En la capa cliente, es el usuario quien realiza la petición de la información requerida para lograr la visualización de la información de interés.

Descripción detallada de la arquitectura

En este apartado se detalla la arquitectura propuesta con cada una de las cuatro capas definidas para realizar el proceso de transformación de la información almacenada en RDF a QB.

5.1.1 Capa de datos

Actualmente, se dispone de un repositorio en RDF, donde se almacenan los datos de los autores con sus publicaciones en un formato de tripletas (sujeto, predicado, objeto). Este repositorio se encuentra en la plataforma *Apache Marmotta* que brinda las facilidades necesarias para aplicaciones de Datos enlazados. Este repositorio se actualiza de forma periódica a través de procesos automáticos.

Adicionalmente, se encuentra el SDW donde se almacenan los datos transformados en QB, para que estos sean extraídos por el componente de la capa superior. Estos datos mantienen el formato de tripletas (sujeto, predicado, objeto) y, de igual manera se encuentra en la plataforma *Apache Marmotta* en un grafo diferente.

5.1.2 Capa de aplicaciones

La capa de aplicaciones se encuentra conformada por una arquitectura de flujo de datos, la misma que se conforma por cuatro componentes, que son:

a) Extracción de datos SPARQL.

El primer filtro inicia con el proceso de extracción de los datos que se encuentran almacenados en el repositorio RDF mediante consultas SPARQL. Para realizar este proceso, primero es necesario determinar qué información va a ser consultada, como se encuentra estructurada y su disponibilidad.

1. Identificar la información con la que se va a trabajar

Para identificar la información con la que se va a trabajar en el modelo multidimensional se consultó a distintos usuarios finales para identificar qué información de interés se desea visualizar en la plataforma; como resultado de ese trabajo se obtuvieron las siguientes preguntas:

- ¿Cuál es el número de publicaciones realizadas por autor, en un período determinado?
- ¿Cuál es el número de publicaciones generadas por cada Institución de Educación Superior, por año?

- ¿Cuál es el número de publicaciones generadas por año?
- ¿Cuáles son las áreas de conocimiento en las que trabaja un autor, en un período determinado?
- ¿Cuáles es el número de publicaciones por año, por autor de la Universidad de Cuenca en un período determinado?

2. Disponibilidad de la información

Como es conocido la plataforma REDI, es un repositorio donde se almacena información de los autores con sus publicaciones (http://redi.cedia.org.ec/sparql/admin/squebi.html); para el presente trabajo es necesario confirmar que la información que se desea visualizar a través de cubo de datos, se encuentre disponible y esto es posible mediante la ejecución de las siguientes consultas SPARQL.

- Para verificar la disponibilidad de los datos del autor como el nombre y apellido se realizó la siguiente (Consulta SPARQL 5.1)

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?Nombre ?Apellido
WHERE {?algo a foaf:Person;
            foaf:firstName ?Nombre;
            foaf:lastName ?Apellido.
}ORDERBY ?Apellido
LIMIT 5
```

Consulta SPARQL 5.1.- Disponibilidad del nombre y apellido de autores

Para el ejemplo, se pide listar cinco nombres y apellidos de autores que se encuentran almacenados en el repositorio REDI (Tabla 5.1)

Nombre	**Apellido**
Víctor	Saquicela
Mauricio	Espinoza
Rodrigo	Fonseca
Xavier	Ochoa
Verónica	Ochoa

Tabla 5.1.- Resultado de Consulta SPARQL 5.1

- Para verificar la disponibilidad de las publicaciones generadas por cada uno de los autores almacenados en el repositorio REDI se realiza la (Consulta SPARQL 5.2)

```
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX bibo: <http://purl.org/ontology/bibo/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
SELECT ?Nombre ?Apellido ?Titulo
WHERE {?algo a foaf:Person;
            foaf:firstName ?Nombre;
            foaf:lastName ?Apellido;
            foaf:publications ?Publicaciones.
           ?Publicaciones a bibo:Document;
             dct:title ?Titulo.
 }ORDERBY ?Apellido ?Nombre
LIMIT 5
```

Consulta SPARQL 5.2.- Disponibilidad de publicaciones por autor

Para el ejemplo se lista cinco publicaciones de un autor que reposa en el repositorio REDI (Tabla 5.2).

Nombre	Apellido	Titulo
Víctor	Saquicela	An automatic method for the enrichment of DICOM metadata using biomedical ontologies.
Víctor	Saquicela	Enriching Electronic Program Guides using semantic technologies and external resources
Víctor	Saquicela	Interlinking Geospatial Information in the Web of Data
Víctor	Saquicela	Lightweight Semantic Annotation of Geospatial RESTful Services
Víctor	Saquicela	GeoDatos enlazados and INSPIRE through an application case

Tabla 5.2.- Resultado Consulta SPARQL 5.2

- Para verificar la disponibilidad de la fecha de publicación de cada uno de los artículos almacenados en el repositorio REDI se realiza la (*Consulta SPARQL 5.3).*

```
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX bibo: <http://purl.org/ontology/bibo/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
SELECT ?Nombre ?Apellido ?Titulo ?AnioPublicacion
```

```
WHERE {?algo a foaf:Person;
            foaf:firstName ?Nombre;
            foaf:lastName ?Apellido;
            foaf:publications ?Publicaciones.
            ?Publicaciones a bibo:Document;
                  dct:title ?Titulo;
                  dct:created ?AnioPublicacion.
 }ORDERBY ?Apellido ?Nombre
LIMIT 05
```

Consulta SPARQL 5.3.- Disponibilidad de la fecha de las publicaciones por autor

Para el ejemplo se lista cinco publicaciones con las respectivas fechas de publicación de un autor del cual reposa información en el repositorio REDI (Tabla 5.3).

Nombre	Apellido	Titulo	AnioPublicacion
Verónica	Ochoa	Familial clustering of autoimmune diseases in patients with type 1 diabetes mellitus	2006
Verónica	Ochoa	Chapter 18 type 1 diabetes mellitus at the crossroad of polyautoimmunity	2008
Verónica	Ochoa	Mucopolisacaridosis tipo iv como causa de talla baja patologica reporte de un caso y revision de la literatura	2008
Verónica	Ochoa	Effects of child and adolescent onset endogenous cushing syndrome on bone mass body composition and growth a 7 year prospective study into young adulthood	2006
Verónica	Ochoa	Glucocorticoid excess during adolescence leads to a major persistent deficit in bone mass and an increase in central body fat	2001

Tabla 5.3.- Resultado de la Consulta SPARQL 5.3

- Para verificar la disponibilidad del área de conocimiento en las que se encuentran trabajando los autores se realiza la (Consulta SPARQL 5.4)

```
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX uc: <http://ucuenca.edu.ec/ontology#>
SELECT DISTINCT ?area WHERE {
  GRAPH <http://ucuenca.edu.ec/wkhuska/clusters> {
        ?grupo foaf:publications  ?pub;
          rdfs:label  ?area; }}
```

Consulta SPARQL 5.4.- Disponibilidad de las áreas de conocimiento de los autores

Para el ejemplo se lista cinco áreas de conocimiento en las que se encuentra trabajando un autor del repositorio REDI (Tabla 5.4).

Nombre	Apellido	Areas
Mauricio	Espinoza	Carry
Mauricio	Espinoza	Semantic Web
Mauricio	Espinoza	recognition
Mauricio	Espinoza	support
Mauricio	Espinoza	Ontology localization

Tabla 5.4.- Resultado de la Consulta SPARQL 5.4

- Para verificar la disponibilidad de los autores que se encuentran almacenados por Institución de Educación Superior, se realiza la siguiente (Consulta SPARQL 5.5)

```
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
PREFIX dct: <http://purl.org/dc/terms/>

SELECT ?Nombre ?Apellido  ?Institucion
WHERE {?algo a foaf:Person;
            foaf:firstName ?Nombre;
            foaf:lastName ?Apellido;
           dct:provenance ?provenance.
           ?provenance <http://ucuenca.edu.ec/ontology#name>
            ?Institucion
 }ORDERBY ?Apellido ?Nombre
 LIMIT 05
```

Consulta SPARQL 5.5.- Disponibilidad de autores por IES

Para el ejemplo se lista cinco autores y la institución a la que pertenecen (Tabla 5.5)

Nombre	Apellido	Institucion
Victor	Saquicela	UCUENCA
Mauricio	Espinoza	UCUENCA
Rodrigo	Fonseca	ESPE
Xavier	Ochoa	ESPOL
Verónica	Ochoa	UDA

Tabla 5.5.- Disponibilidad de autores por IES

Para verificar la disponibilidad de áreas de conocimiento en las que se encuentra trabajando las Instituciones de Educación Superior se realiza la (Consulta SPARQL 5.6). Para el ejemplo se lista cinco áreas de conocimiento de una IES que se encuentre almacenada en el repositorio (Tabla 5.6).

```
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX bibo: <http://purl.org/ontology/bibo/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
PREFIX uc:<http://ucuenca.edu.ec/ontology#>
SELECT DISTINCT   ?name ?area
WHERE { GRAPH <http://ucuenca.edu.ec/wkhuska/clusters>
    { ?grupo foaf:publications  ?pub;
     rdfs:label  ?area.
   ?pub uc:hasPerson ?autor.
    {   SELECT *  {GRAPH <http://ucuenca.edu.ec/wkhuska>
      { ?autor dct:provenance ?provenance.}}}
  { SELECT *  {GRAPH <http://ucuenca.edu.ec/wkhuska/endpoints>
      { ?provenance uc:name ?name.}}}}}
   Orderby ?name
```

Consulta SPARQL 5.6.- Disponibilidad de las áreas de conocimiento por IES

Institución	**Areas**
UCUENCA	method
UCUENCA	data
UCUENCA	system
UCUENCA	blind
UCUENCA	history

Tabla 5.6.- Resultado de la Consulta SPARQL 5.6

De acuerdo a las consultas realizadas anteriormente se puede confirmar que se dispone de la información necesaria para formular las respuestas a las preguntas planteadas. Como ejemplo se ha tomado los datos disponibles del autor “Víctor Saquicela” perteneciente a la Universidad de Cuenca para ejemplificar todas las respuestas a las consultas planteadas (Figura 5.2).

Actualmente, el usuario no puede realizar este tipo de consultas directamente en la plataforma REDI, tal como se había indicado anteriormente, las consultas que se encuentran disponibles en la plataforma son pre-configuradas por los desarrolladores del proyecto y para que el usuario tenga acceso a esta información es necesario que disponga de conocimiento en SPARQL y las ejecute.

Es necesario indicar que los datos presentados en las tablas anteriores corresponden a la información tal cual ha sido almacenada en el repositorio, sin modificaciones ni correcciones en su escritura y concepto.

3. Establecer dimensiones y medidas a utilizar

Basado en las preguntas identificadas en la sección 5.1.2, se definió un modelo multidimensional que reflejan las dimensiones y medidas necesarias para realizar este estudio (Figura 5.3):

- Dimensiones:
  - Autor: se refiere a la persona que genera publicaciones;
  - Área de conocimiento: es el área de conocimiento en la que se encuentra trabajando el autor;
  - Año de publicación: se refiere al año en el que se generó la publicación, y
  - Institución de Educación Superior: es la institución a la que pertenece el autor.
- Medida:
  - Número de publicaciones: se refiere al número de publicaciones que ha generado un autor, la misma que puede ser calculada por cada dimensión propuesta.

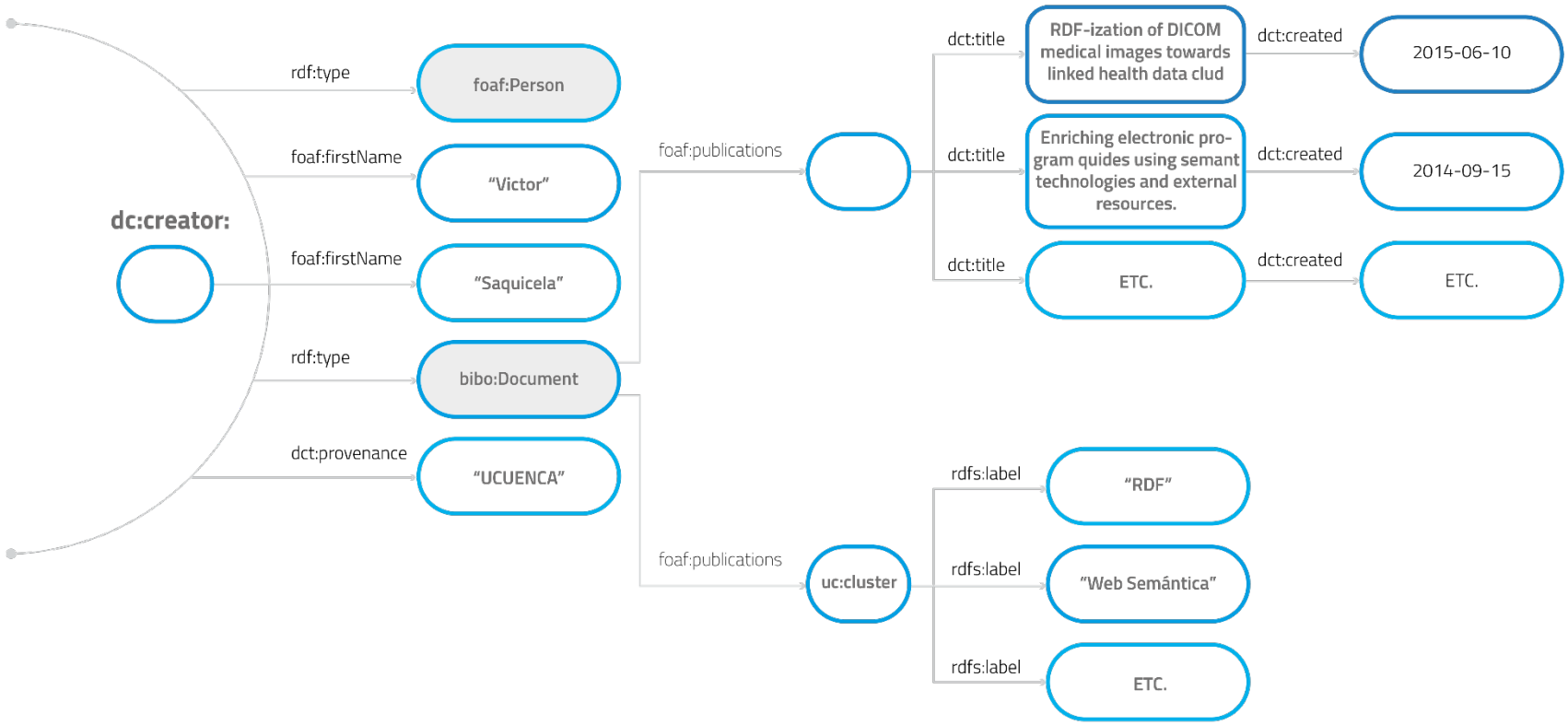


Figura 5.2.- Grafo en RDF de la información disponible en el repositorio RDF del proyecto REDI

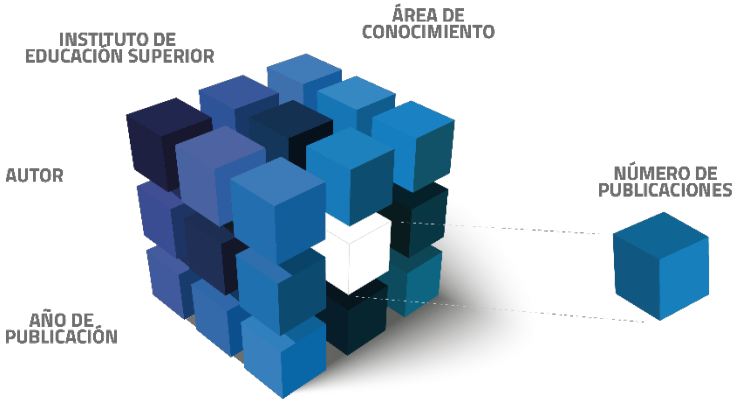


Figura 5.3.- Dimensiones y hechos del cubo multidimensional

b) Proceso de transformación de RDF a QB

Para realizar el proceso de transformación de acuerdo a la literatura estudiada 5.1, se debe identificar claramente los pasos a seguir para dicho proceso, basándose en el vocabulario QB simplificado (Figura 5.4). Tal como se indicó en el apartado 4.4, el vocabulario Cubo de datos en RDF se basa en diferentes vocabularios que son incorporados de acuerdo a las necesidades de este proceso de transformación, esto depende de las clases y propiedades que posean de cada uno de ellos (Tabla 5.7).

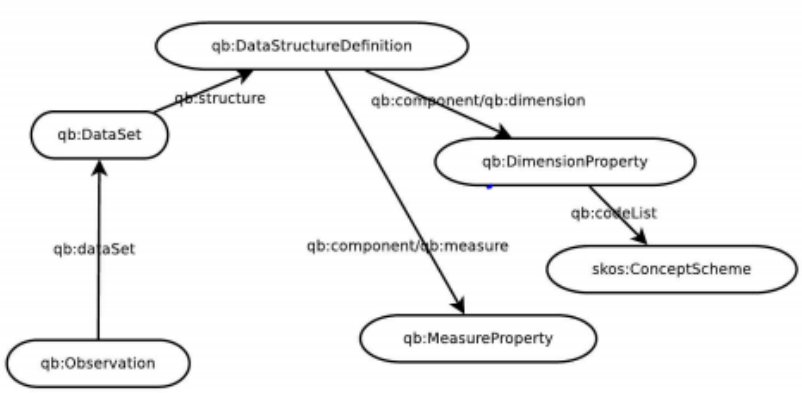


Figura 5.4.- Vocabulario QB simplificado (Kämpgen, 2015)

1	prefix pub: <http://190.15.141.66:8899/ucuenca/>
2	prefix : <http://purl.org/dc/elements/1.1/>
3	prefix foaf: <http://xmlns.com/foaf/0.1/>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
6	prefix dct: <http://purl.org/dc/terms/>
7	prefix qb: <http://purl.org/linked-data/cube#>
8	prefix skos: <http://www.w3.org/2004/02/skos/core#>
9	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
10	prefix sdmxattribute: <http://purl.org/linked-data/sdmx/2009/attribute#>
11	prefix sdmxmeasure: <http://purl.org/linked-data/sdmx/2009/measure#>
12	prefix sdmxdimension: <http://purl.org/linked-data/sdmx/2009/dimension#>
13	prefix sdmxcode: <http://purl.org/linked-data/sdmx/2009/code#>
14	prefix sdmxconcept: <http://purl.org/linked-data/sdmx/2009/concept#>

Tabla 5.7.- Vocabularios a utilizar para realizar la transformación a QB

En la (Figura 5.5), se detalla el proceso realizado para la generación de la estructura y observaciones del cubo de datos multidimensional basado en Tecnología Semántica.

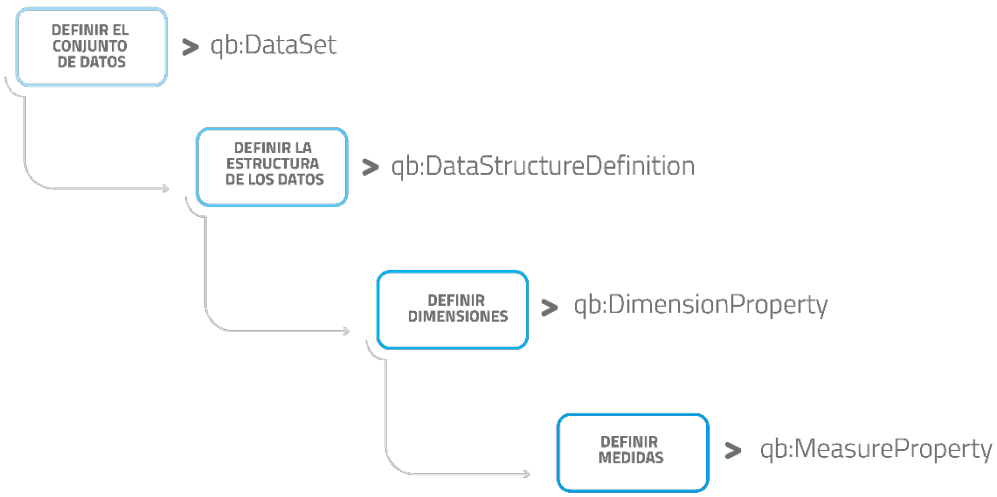


Figura 5.5.- Pasos para la transformación de RDF a QB

1. Definir el conjunto de datos: Para definir que algo es un conjunto de datos se utiliza la clase **qb:DataSet**, y se asigna un nombre para identificarlo dentro de la estructura "*dataset-np1*"**;** con la propiedad **rdfs:label** se define un nombre único al conjunto de datos para identificarlos en el momento de la visualización; con la propiedad **rdfs:comment** se indica más

información sobre el conjunto de datos; con la propiedad **qb:structure** se indica que algo pertenece a ese conjunto de datos, en este caso sería, "*pub:dsd-np*" nombre de la definición de la estructura de datos que se define más adelante (Tabla 5.8).

```
pub:dataset-np1 a qb:DataSet;
 rdfs:label "Total de publicaciones1"@es;
 rdfs:comment "Conteo de publicaciones de los autores del REDI"@es;
 qb:structure pub:dsd-np ;
 sdmxattribute:unitMeasure <http://dbpedia.org/page/Number> .
```

Tabla 5.8.- Generación del conjunto de datos; DataSet

2. Definir la estructura de los datos: Para definir que algo es una estructura de datos se utiliza la clase **qb:DataStructureDefinition** y se asigna un nombre para que sea identificad dentro de la estructura del cubo "*dsd-np*". Las dimensiones, medidas y atributos son componentes por lo que es necesario declararlos, una estructura de datos puede conformarse por varios componentes con la propiedad **qb:component***;* con la propiedad **qb:dimension** se indica que algo es una dimensión, y tiene un nombre, adicionalmente, se declara un orden o nivel de importancia de cada dimensión creada con la propiedad **qb:order**.

   Con la propiedad **qb:measure** se indica que algo es una medida, y se asigna un nombre. Con la propiedad **qb:attribute** se indica que algo es un atributo y se declara la unidad de la medida. Con la propiedad **qb:componentAttachment** se indica que esta definición de estructura de datos pertenece a un DataSet (Tabla 5.9).

```
pub:dsd-np a qb:DataStructureDefinition;
# The dimensions
qb:component [qb:dimension pub:refAutor; qb:order 1];
qb:component [qb:dimension pub:refAreaConocimiento; qb:order 2];
qb:component [qb:dimension pub:refFuente; qb:order 3];
qb:component [qb:dimension pub:refPeriod; qb:order 4];
# The measure(s)
qb:component [qb:measure pub:numPublicaciones];
# The attributes
qb:component [qb:attribute sdmxattribute:unitMeasure; qb:componentAttachment
qb:DataSet;] .
```

Tabla 5.9.- Definición de la estructura de datos y del conjunto de datos

3. Definir dimensiones: Tal como se indicó anteriormente se determinó cuatro dimensiones para realizar el trabajo propuesto, como se presenta:

- Dimensión "*Autor*", se refiere a la persona que genera publicaciones, se renombró a *refAutor*.
- Dimensión "*Área de conocimiento"*, es el área de conocimiento en la que se encuentra trabajando el autor, se renombró a *refAreaConocimiento.*
- Dimensión "*Año de publicación*", se refiere al año en el que se generó la publicación, se renombró a *refPeriod.*
- Dimensión "*Institución de Educación Superior*", es la institución a la que pertenece el autor, se renombró a *rdfFuente.*

Identificadas las dimensiones y la medida se proceden a declararlas dentro de la estructura de datos definida (Tabla 5.10).

```
pub:refAutor a rdf:Property, qb:DimensionProperty;
rdfs:label "Autor de Publicaciones"@es;
rdfs:subPropertyOf sdmxdimension:refAutor;
rdfs:range skos:Concept;
qb:concept sdmxconcept:refAutor .
```

```
pub:refAreaConocimiento a rdf:Property,
qb:DimensionProperty;
rdfs:label "Area de conocimiento"@es;
rdfs:subPropertyOf sdmxdimension:refAreaConocimiento;
rdfs:range skos:Concept;
qb:concept sdmxconcept:refAreaConocimiento .
```

```
pub:refFuente a rdf:Property, qb:DimensionProperty;
rdfs:label "IES"@es;
rdfs:subPropertyOf sdmxdimension:refFuente;
rdfs:range skos:Concept ;
qb:concept sdmxconcept:refFuente .
```

```
pub:refPeriod a rdf:Property, qb:DimensionProperty;
rdfs:label "Anio de la publicacion"@es;
rdfs:subPropertyOf sdmxdimension:refPeriod;
rdfs:range skos:Concept;
qb:concept sdmxconcept:refPeriod .
```

Tabla 5.10.- Definición de las dimensiones

En la (Tabla 5.10) se indica el nombre de cada dimensión creada previamente en el "*dsd-np*" y se indica que es parte de la clase **rdf:Property** y **qb:DimensionProperty**, se asigna a cada dimensión con un nombre para la visualización de los datos con la propiedad **rdfs:label,** se indica que es parte del vocabulario SDMX con las propiedades **sdmxdimension:** y **sdmxconcept.**

4. Definir medidas: Para realizar el trabajo propuesto se identificó una medida, como se presenta:
   - Medida "*Número de publicaciones*", se refiere al total de publicaciones que ha generado un autor, puede ser calculada por cada dimensión propuesta, se renombró a *numPublicaciones* (Tabla 5.11)

```
pub:numPublicaciones a rdf:Property, qb:MeasureProperty;
rdfs:label "Total de publicaciones Autores"@es;
rdfs:subPropertyOf sdmxmeasure:obsValue;
rdfs:range xsd:integer .
```

Tabla 5.11.- Definición de la medida

En la (Tabla 5.11) se indica el nombre de la medida para identificarla dentro de la estructura del cubo, y se indica que forma parte de la clase **rdf:Property** y **qb:MeasureProperty** se asigna un nombre para que sea identificado en el momento de realizar la visualización de los datos, se indica que es un valor a ser observado con la propiedad **sdmxmeasure:obsValue**, e indica la unidad de la medida con **xsd:integer.**

Con la estructura del cubo lista se procede a generar la información/observaciones de acuerdo a cada una de las dimensiones y la medida. Para identificar que es una observación se utiliza la clase **qb:Observation**. Para la visualización de la información en este documento, se ha recogido información del autor "Víctor Saquicela" de la Universidad de Cuenca, con dos publicaciones (Tabla 5.12).

```
pub:extraccion-de-preferencias-televisivas-desde-los-perfiles-de-redes-sociales
pub:numPublicaciones "1";
 qb:measureType pub:numPublicaciones ;
 pub:refAreaConocimiento "Digital" ;
 pub:refFuente "UCUENCA" ;
 pub:refPeriod "2014" ;
```

pub:refAutor <http://ucuenca.edu.ec/resource/author/victor-saquicela>; qb:dataSet pub:dataset-np1 ; a qb:Observation .
pub:adding-semantic-annotations-into-geospatial-restful-services pub:numPublicaciones "1" ; qb:measureType pub:numPublicaciones ; pub:refAreaConocimiento "Semantic"; pub:refFuente "UCUENCA" ; pub:refPeriod "2012" ; pub:refAutor <http://ucuenca.edu.ec/resource/author/victor-saquicela>; qb:dataSet pub:dataset-np1 ; a qb:Observation .

Tabla 5.12.- Observaciones del cubo de datos

La transformación de la información que se encuentra en el repositorio, se puede realizar de manera íntegra de todo el repositorio RDF o por cada consulta que se realice.

c) SPARQL para almacenamiento

Con la información disponible y transformada en QB, es necesario publicar por primera vez en la plataforma *Apache Marmotta* la estructura de los datos y las observaciones para que estos sean almacenados en tripletas en el SDW. Para realizar la inserción de los datos es necesario especificar el grafo donde se va a almacenar la información requerida, para el presente trabajo de tesis se trabajó en *<http://lapproy01.cedia.org.ec:8080/marmotta/context/Redi1pQB>* (Tabla 5.13).

La inserción de la estructura y de los datos se puede realizar por separado en el mismo grafo o de manera completa en una única inserción.

```
1  prefix pub: <http://190.15.141.66:8899/ucuenca/>
2  prefix : <http://purl.org/dc/elements/1.1/>
3  prefix foaf: <http://xmlns.com/foaf/0.1/>
4  prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5  prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
6  prefix dct: <http://purl.org/dc/terms/>
7  prefix qb: <http://purl.org/linked-data/cube#>
8  prefix skos: <http://www.w3.org/2004/02/skos/core#>
9  prefix xsd: <http://www.w3.org/2001/XMLSchema#>
10 prefix sdmxattribute: <http://purl.org/linked-data/sdmx/2009/attribute#>
```

```
prefix sdmxmeasure: <http://purl.org/linked-data/sdmx/2009/measure#>
prefix sdmxdimension: <http://purl.org/linked-data/sdmx/2009/dimension#>
prefix sdmxcode: <http://purl.org/linked-data/sdmx/2009/code#>
prefix sdmxconcept: <http://purl.org/linked-data/sdmx/2009/concept#>


insert  data {
   graph <http://lapproy01.cedia.org.ec:8080/marmotta/context/Redi1pQB>
{
#Definicion del Conjunto de Datos DataSet
pub:dataset-np1p a qb:DataSet;
                        rdfs:label "Total publicaciones"@es;
                        rdfs:comment "Conteo de publicaciones de los autores del REDI"@es;
                        qb:structure pub:dsd-np1p ;
                        sdmxattribute:unitMeasure <http://dbpedia.org/page/Number> .
#Definicion de la Estructura de Datos DataStructureDefinition
pub:dsd-np1p a qb:DataStructureDefinition;
# The dimensions
qb:component [qb:dimension pub:refAutor; qb:order 1];
qb:component [qb:dimension pub:refAreaConocimiento; qb:order 2];
qb:component [qb:dimension pub:refFuente; qb:order 3];
qb:component [qb:dimension pub:refPeriod; qb:order 4];
# The measure(s)
qb:component [qb:measure pub:numPublicaciones];
# The attributes
qb:component [qb:attribute sdmxattribute:unitMeasure; qb:componentAttachment
qb:DataSet;] .
# Definicion de Dimensiones
pub:refAutor a rdf:Property, qb:DimensionProperty;
                        rdfs:label "Autor de Publicaciones"@es;
                        rdfs:subPropertyOf sdmxdimension:refAutor;
                        rdfs:range skos:Concept;
                        qb:concept sdmxconcept:refAutor .

pub:refAreaConocimiento a rdf:Property, qb:DimensionProperty;
                        rdfs:label "Area de conocimiento"@es;
                        rdfs:subPropertyOf sdmxdimension:refAreaConocimiento;
                        rdfs:range skos:Concept;
                        qb:concept sdmxconcept:refAreaConocimiento .

pub:refFuente a rdf:Property, qb:DimensionProperty;
                        rdfs:label "IES"@es;
                        rdfs:subPropertyOf sdmxdimension:refFuente;
```

```
53                      rdfs:range skos:Concept ;
54                      qb:concept sdmxconcept:refFuente .
55
56  pub:refPeriod a rdf:Property, qb:DimensionProperty;
57                      rdfs:label "Anio de la publicacion"@es;
58                      rdfs:subPropertyOf sdmxdimension:refPeriod;
59                      rdfs:range skos:Concept;
60                      qb:concept sdmxconcept:refPeriod .
61
62  #Definicion de Medidas
63  pub:numPublicaciones a rdf:Property, qb:MeasureProperty;
64                      rdfs:label "Total de publicaciones Autores"@es;
65                      rdfs:subPropertyOf sdmxmeasure:obsValue;
66                      rdfs:range xsd:integer .
67
```

Tabla 5.13.- Inserción de la estructura del cubo de datos

Adicionalmente, se realiza la inserción de las observaciones con la que se van a trabajar en el cubo, para esta visualización se realiza con información de dos publicaciones del autor “Víctor Saquicela” (Tabla 5.14).

```
pub:extraccion-de-preferencias-televisivas-desde-los-perfiles-de-redes-sociales
pub:numPublicaciones "1";
 qb:measureType pub:numPublicaciones ;
 pub:refAreaConocimiento "Digital" ;
 pub:refFuente "UCUENCA" ;
 pub:refPeriod "2014" ;
 pub:refAutor <http://ucuenca.edu.ec/resource/author/victor-saquicela>;
 qb:dataSet pub:dataset-np1 ;
 a qb:Observation .
```

```
pub:adding-semantic-annotations-into-geospatial-restful-services
pub:numPublicaciones "1" ;
 qb:measureType pub:numPublicaciones ;
 pub:refAreaConocimiento "Semantic";
 pub:refFuente "UCUENCA" ;
 pub:refPeriod "2012" ;
 pub:refAutor <http://ucuenca.edu.ec/resource/author/victor-saquicela>;
 qb:dataSet pub:dataset-np1 ;
 a qb:Observation .
```

Tabla 5.14.- Inserción de 2 observaciones en el cubo de datos multidimensionales basado en Tecnología Semántica

El proceso presentado en este trabajo de tesis se realizó de manera manual (Ad-Hoc) para demostrar lo analizado en literatura o estado del arte, e indicar que mediante un proceso detallado es posible realizar la transformación de RDF a QB. La automatización de este trabajo, se realizó con la ayuda de software que procesa toda la información *(Jena*[24]*)*, y se encuentra implementado en el prototipo del REDI en http://redi.cedia.org.ec/#/es/data/datacube.

3.1. Políticas de actualización de la información

La actualización de la información en el SDW se puede realizar de manera total e incremental. La actualización total consiste en detectar un cambio en el repositorio RDF mediante una revisión periódica de la información, esto se puede realizar mediante consultas SPARQL la primera opción es realizar un conteo de las tripletas almacenadas, y la segunda opción para detectar cambios realizados en el repositorio. Una vez detectado el cambio se elimina toda la información almacenada y se carga nuevamente toda la información, esto se puede realizar en la plataforma *Apache Marmotta*, en el menú "*Context Manager*", se identifica el grafo y se procede a la eliminación del mismo.

La actualización incremental consiste en trabajar con los resultados obtenidos previos al cambio y continuar con el proceso de actualización de los nuevos datos. De acuerdo a (Reyes Álvarez, Hidalgo Delgado, Martínez Rojas, Roldán García, & Aldana-Montes, 2014), para realizar una actualización incremental se debe realizar este proceso en tres fases, la primera consiste en detectar un cambio en la BD, la segunda fase es generar la tripletas con los nuevos cambios y la tercera fase es la actualización del grafo con las nuevas tripletas (Figura 5.6).

[24] *https://jena.apache.org/tutorials/rdf_api.html*

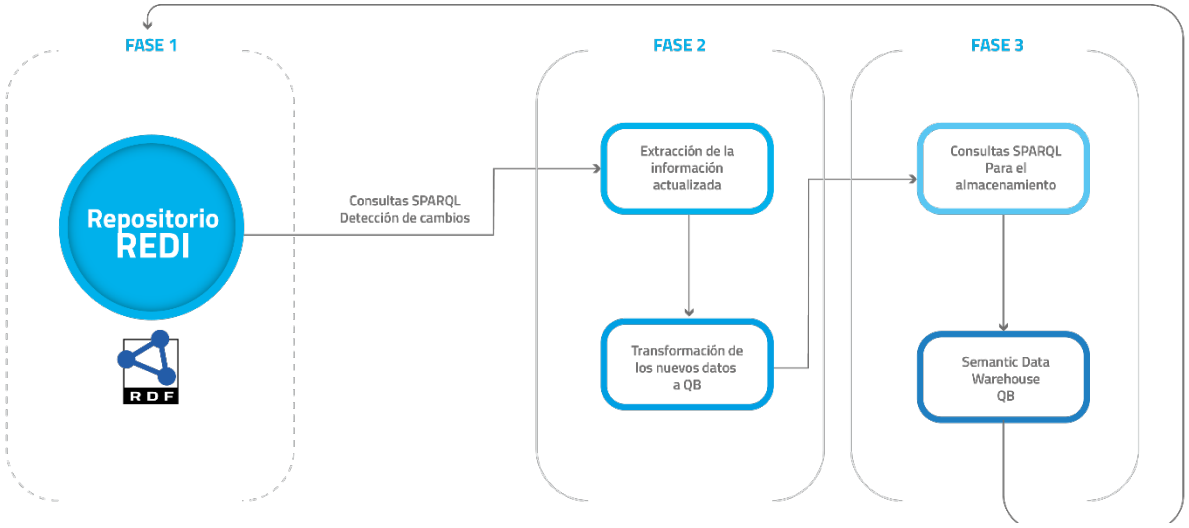


Figura 5.6.- Proceso de detección de la información actualizada

Mediante consultas SPARQL, se puede detectar cambios en el repositorio REDI como por ejemplo, la incorporación de nuevos autores, nuevas áreas de conocimiento, publicaciones, o si un autor en particular incrementó o modificó publicaciones o áreas de conocimiento (Consulta SPARQL 5.7).

```
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
SELECT (count(?Nombre) as ?NumAutores)
WHERE {?algo foaf:firstName ?Nombre}
```

```
PREFIX foaf:<http://xmlns.com/foaf/0.1/>
SELECT (count(?Publicaciones) as ?NumPublicaciones)
WHERE {?algo foaf:publications  ?Publicaciones}
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT Distinct (count(?AreaConocimiento) as
?NumAreaConocimiento)
WHERE {GRAPH
<http://ucuenca.edu.ec/wkhuska/clusters>
    {?grupo rdfs:label  ?AreaConocimiento}}
```

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT (count(?Publicaciones) as
?NumPublicaciones)
```

```
WHERE
{ GRAPH <http://ucuenca.edu.ec/wkhuska>
      { <http://ucuenca.edu.ec/resource/author/víctor-
saquicela>
    foaf:publications  ?Publicaciones.
    }
}
```

Consulta SPARQL 5.7.- Recolección de información almacenada en el repositorio REDI

De acuerdo a la ejecución de estas consultas se obtuvo los resultados del número de autores, número de publicaciones, número de áreas de conocimiento que se encuentran registrados en el repositorio REDI (Tabla 5.15). Adicionalmente, se obtuvo datos específicos sobre el autor "Víctor Saquicela" (Tabla 5.16).

NumAutores	NumPublicaciones	NumAreaConocimiento
147.946	14.111	166

Tabla 5.15.- Resultado de la Consulta SPARQL 5.7

NumPublicaciones
30

Tabla 5.16.- Resultado de la Consulta SPARQL 5.7 sobre el autor "Víctor Saquicela"

Cuando se realice el primer conteo de la información esta se almacena y se verifica cada sábado a las 23h00, cuando se detecte algún cambio en la información almacenada esta ser identifica para continuar con el proceso, para el ejemplo, se ha detectado que el número de publicaciones ha incrementado en cinco publicaciones del autor "Víctor Saquicela" y las áreas de conocimiento se mantienen igual (Tabla 5.17 y Tabla 5.18).

NumAutores	NumPublicaciones	NumAreaConocimiento
147.946	14.116	166

Tabla 5.17.- Detección de aumento de publicaciones en el repositorio REDI

NumPublicaciones
35

Tabla 5.18.- Detección de aumento de publicaciones del autor "Víctor Saquicela" en el repositorio REDI

Mediante la (Consulta SPARQL 5.8), se puede listar las publicaciones del autor “Víctor Saquicela” y se confirma la incorporación de nuevas cinco publicaciones incorporadas en el repositorio REDI (Tabla 5.19).

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT ?Publicaciones
WHERE { GRAPH <http://ucuenca.edu.ec/wkhuska>{
  <http://ucuenca.edu.ec/resource/author/víctor-saquicela>
 foaf:publications  ?Publicaciones.
}}
```

Consulta SPARQL 5.8.- Detectar cambios en las publicaciones del autor "Víctor Saquicela"

#	Publicaciones
1	http://ucuenca.edu.ec/wkhuska/publication/enriching-electronic-program-guides-using-semantic-technologies-and-external
2	http://ucuenca.edu.ec/wkhuska/publication/an-automatic-method-for-the-enrichment-of-dicom-metadata-using-biomedical
3	http://ucuenca.edu.ec/wkhuska/publication/interlinking-geospatial-information-in-the-web-of
4	http://ucuenca.edu.ec/wkhuska/publication/lightweight-semantic-annotation-of-geospatial-restful
5	http://ucuenca.edu.ec/wkhuska/publication/geolinked-data-and-inspire-through-an-application
6	http://ucuenca.edu.ec/wkhuska/publication/geographical-linked-spanish-use
7	http://ucuenca.edu.ec/wkhuska/publication/semantic-annotation-of-restful-services-using-external
8	http://ucuenca.edu.ec/wkhuska/publication/adding-semantic-annotations-into-restful
9	http://ucuenca.edu.ec/wkhuska/publication/marco-de-trabajo-para-la-integracion-de-recursos-digitales-basado-en-un-enfoque-de-web-semantica
10	http://ucuenca.edu.ec/wkhuska/publication/explotacion-de-informacion-en-el-dominio-geo-hidrico-ecuatoriano-utilizando-tecnologia-semantica
11	http://ucuenca.edu.ec/wkhuska/publication/webmedsa-web-based-framework-for-segmenting-and-annotating-medical-images-using-biomedical-ontologies
12	http://ucuenca.edu.ec/wkhuska/publication/integration-and-massive-storage-of-hydro-meteorological-data-combining-big-data-semantic-web-technologies
13	http://ucuenca.edu.ec/wkhuska/publication/hacia-la-creacion-de-un-repositorio-semantico-de-modelos-de-contexto-basados-en-el-metodo-dharma
14	http://ucuenca.edu.ec/wkhuska/publication/plataforma-para-la-busqueda-por-contenido-visual-semantico-de-imagenes-medicas
15	http://ucuenca.edu.ec/wkhuska/publication/rdf-ization-of-dicom-medical-images-towards-linked-health-data-cloud
16	http://ucuenca.edu.ec/wkhuska/publication/towards-the-creation-of-semantic-repository-of-istar-based-context-models
17	http://ucuenca.edu.ec/wkhuska/publication/lightweight-semantic-annotation-of-geospatial-rest-ful-services

18	http://ucuenca.edu.ec/wkhuska/publication/reconocimiento-de-caracteres-del-alfabeto-dactilologico-mediante-redes-neuronales-artificiales-un-enfoque-experimental
19	http://ucuenca.edu.ec/wkhuska/publication/semantic-annotation-of-restful-services-using-external-resources
20	http://ucuenca.edu.ec/wkhuska/publication/extraccion-de-preferencias-televisivas-desde-los-perfiles-de-redes-sociales
21	http://ucuenca.edu.ec/wkhuska/publication/design-of-an-integrated-decision-support-system-for-library-holistic-evaluation
22	http://ucuenca.edu.ec/wkhuska/publication/adding-semantic-annotations-into-geospatial-restful-services
23	http://ucuenca.edu.ec/wkhuska/publication/interlinking-geospatial-information-in-the-web-of-data
24	http://ucuenca.edu.ec/wkhuska/publication/anotacion-semantica-de-web-feature-services
25	http://ucuenca.edu.ec/wkhuska/publication/integracion-de-repositorios-de-acceso-abierto-del-ecuador-traves-de-un-enfoque-de-web-semantica
26	http://ucuenca.edu.ec/wkhuska/publication/semantic-annotation-of-geospatial-restful-services-using-external-resources
27	http://ucuenca.edu.ec/wkhuska/publication/integrated-decision-support-system-idss-for-library-holistic-evaluation
28	http://ucuenca.edu.ec/wkhuska/publication/enriching-electronic-program-guides-using-semantic-technologies-and-external-resources
29	http://ucuenca.edu.ec/wkhuska/publication/geographical-linked-data-spanish-use-case
30	http://ucuenca.edu.ec/wkhuska/publication/sistema-de-recomendacion-de-contenidos-audiovisuales-algoritmo-de-inferencia-semantica
31	*http://ucuenca.edu.ec/wkhuska/publication/semantic-recommender-systems-for-digital-tv-from-demographic-stereotyping-to-personalized-recommendations*
32	*http://ucuenca.edu.ec/wkhuska/publication/literature-review-of-data-mining-applications-in-academic-libraries*
33	*http://ucuenca.edu.ec/wkhuska/publication/geolinked-data-and-inspire-through-an-application-case*
34	*http://ucuenca.edu.ec/wkhuska/publication/analisis-de-la-influencia-de-las-propiedades-semanticas-en-los-sistemas-de-recomendacion*
35	*http://ucuenca.edu.ec/wkhuska/publication/an-automatic-method-for-the-enrichment-of-dicom-metadata-using-biomedical-ontologies*

Tabla 5.19.- Detección de nueva información del autor "Víctor Saquicela"

Con la información detectada se procede a incorporar las nuevas observaciones en el grafo creado para este trabajo (Consulta SPARQL 5.9).

```
insert data {
  graph <http://lapproy01.cedia.org.ec:8080/marmotta/context/Ejemplo1QB>
```

```
{
pub:semantic-recommender-systems-for-digital-tv-from-demographic-
stereotyping-to-personalized-recommendations  pub:numPublicaciones "1" ;
 qb:measureType pub:numPublicaciones ;
 pub:refAreaConocimiento "Recognition";
 pub:refFuente "UCUENCA" ;
 pub:refPeriod "2015" ;
 pub:refAutor <http://ucuenca.edu.ec/resource/author/victor-saquicela>;
 qb:dataSet pub:dataset-np1 ;
 a qb:Observation .
```

Consulta SPARQL 5.9.- Inserción de nueva publicación del autor "Víctor Saquicela"

De esta manera se incorpora en el grafo la nueva información hasta completar todos los cambios detectados. Este proceso se realizó de manera manual (Ad-Hoc), para comprobar lo estudiado en el estado del arte, el mismo se presenta automatizado y se puede apreciar el prototipo en http://redi.cedia.org.ec/#/es/data/datacube.

5) Extracción de datos QB

Mediante la ejecución de consultas SPARQL, se puede acceder a la información almacenada en el SDW como por ejemplo: dimensiones, medidas y observaciones, (Consulta SPARQL 5.10) y el resultado se puede apreciar en la (Tabla 5.20)

```
PREFIX qb: <http://purl.org/linked-data/cube#>
PREFIX pub: <http://190.15.141.66:8899/ucuenca/>
SELECT DISTINCT *
WHERE {?dim a qb:DimensionProperty}
```

```
PREFIX qb: <http://purl.org/linked-data/cube#>
PREFIX pub: <http://190.15.141.66:8899/ucuenca/>
SELECT DISTINCT *
WHERE {?dim a qb:MeasureProperty}
```

```
PREFIX qb: <http://purl.org/linked-data/cube#>
PREFIX pub: <http://190.15.141.66:8899/ucuenca/>
SELECT DISTINCT *
WHERE {?dim a qb:Observation}
```

Consulta SPARQL 5.10.- Extracción de información almacenada en el SDW

Medida	Dimensión	Observaciones
NumPublicacion	rdfFuente	rdf-ization-of-dicom-medical-images-towards-linked-health-data-cloud
	refPeriodo	plataforma-para-la-busqueda-por-contenido-visual-semantico-de-imagenes-medicas
	refAutor	wearable-biomedical-measurement-systems-for-assessment-of-mental-stress-of-combatants-in-real-time
	refAreaConocimiento	estudio-seleccion-de-una-arquitectura-orientada-servicios-soa-que-permita-la-integracion-de-sistemas-informaticos-legados

Tabla 5.20.- Resultado de la Consulta SPARQL 5.10

Este tipo de consultas es posible realizar únicamente sobre en el SDW, debido a que ésta información se encuentra actualmente en modelos multidimensionales basados en Tecnología Semántica. La estructura de los datos y los datos para entender de una mejor manera se puede apreciar en forma de grafo (Figura 5.7 y Figura 5.8).

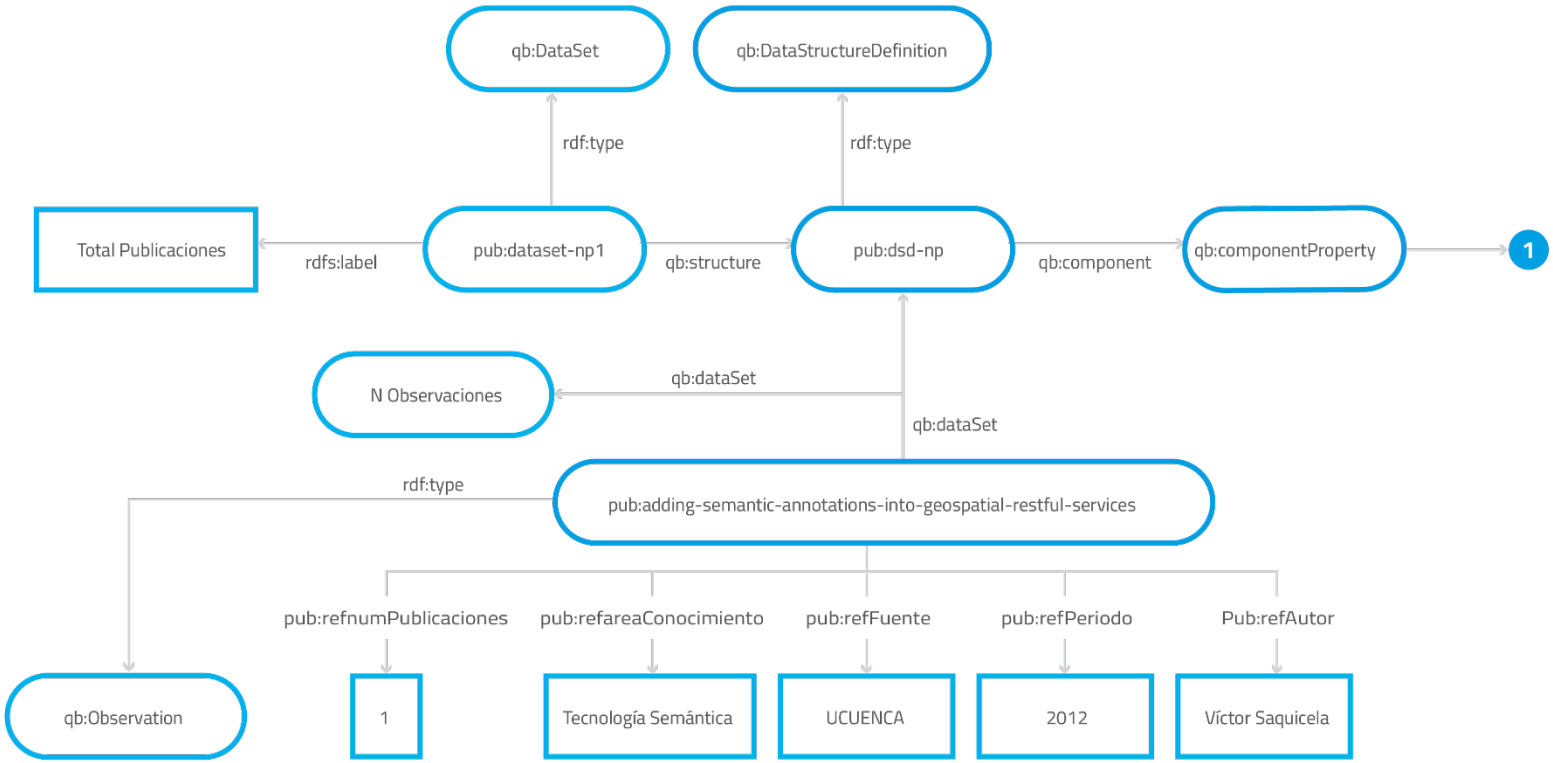


Figura 5.7.- Grafo de la estructura y observaciones basados en el vocabulario simplificado QB 1/2

Figura 5.8.- Grafo de la estructura y observaciones basados en el vocabulario simplificado QB 2/2

5.1.3 Capa de presentación

La capa de presentación es la encargada de visualizar la información que se encuentra almacenada en el SDW, para esto, es necesario utilizar una herramienta que permita la visualización de datos multidimensionales semánticos. De acuerdo a la literatura estudiada, y al análisis realizado en el apartado 4.5, para el presente trabajo de tesis se utilizó la herramienta "OpenCube Toolkit[25]".

La primera vez que se accede a esta herramienta es necesario configurar algunos widgets para enlazar con la información que se encuentra en el repositorio *Marmotta*. Por única vez, en el menú superior dirigirse a "*Admin*", y escoger la opción "*Data Provider Setup*" y crear uno, para conectar las dos herramientas (Figura 5.9), donde:

a. Se escoge el tipo de "Data Provider" que se desea crear, en este caso se escoge "SPARQLEndpointProvider"
b. Se coloca un nombre que lo identifique
c. Se establece el tiempo de actualización de datos "60 minutos".
d. Se coloca el Endponit con el que se está trabajando http://lapproy01.cedia.org.ec:8080/marmotta/sparql/select
e. Se define el query para realizar la conexión

```
construct {?subject ?property ?object}
from <http://lapproy01.cedia.org.ec:8080/marmotta/context/Redi1pQB>
where {?subject ?property ?object}
```

[25] *http://opencube-toolkit.eu/*

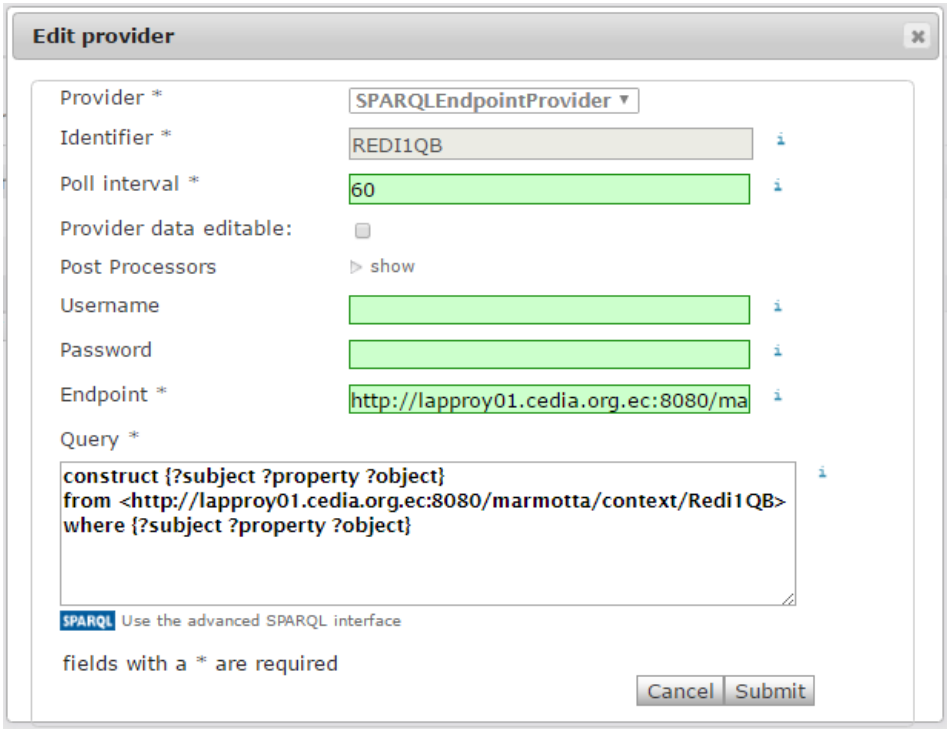


Figura 5.9.- Configuración de un "Data Provider"

Mediante el widget “OpenCube Compatibily Explorer”, se comprueba la compatibilidad de cubo creado para realizar los cálculos con las medidas y dimensiones establecidas para iniciar el manejo de los datos (Figura 5.10)

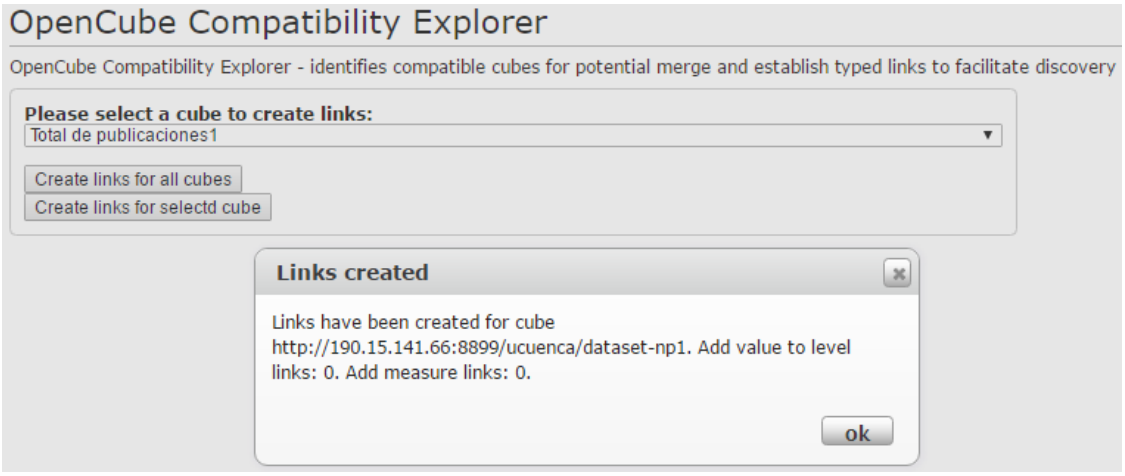


Figura 5.10.- Comprobación de compatibilidad del cubo creado

Con el widget “OpenCube Aggregator” se genera de acuerdo a los datos, las agregaciones de suma y promedio, una vez escogido el tipo de comportamiento para la medida, se procede a generar el cubo (Figura 5.11)

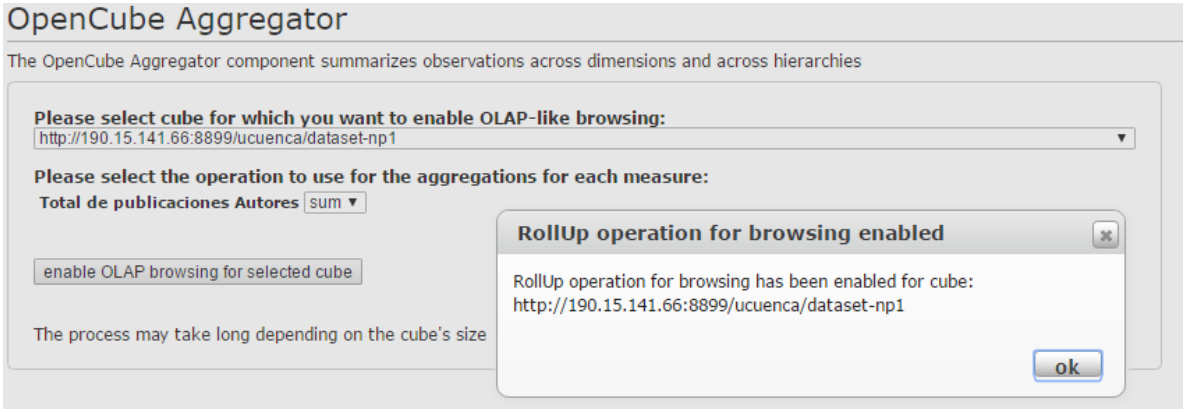


Figura 5.11.- Generación de las agregaciones para las dimensiones propuestas.

Finalmente, con el widget "OpenCube Browser" se procede a la visualización del modelo multidimensional basados en Tecnología Semántica.

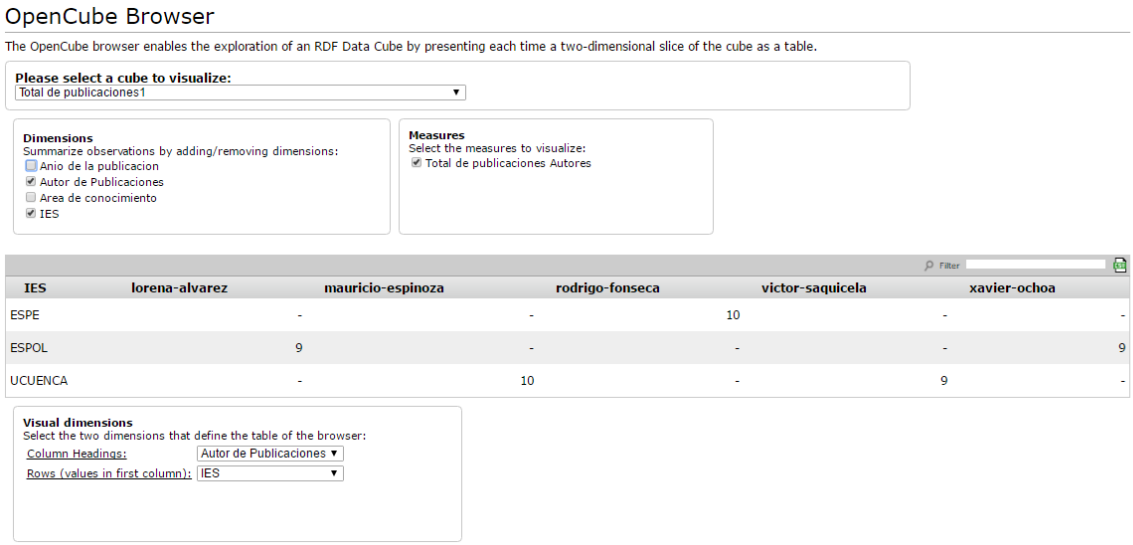


Figura 5.12.- Visualización de la información mediante modelos multidimensionales basados en Tecnología Semántica, Número de publicaciones por autor por Institución de Educación Superior.

Estas visualizaciones aun no son accesibles para el usuario, puesto que estas configuraciones se han realizado en el *"back end"* de la plataforma, comprensible solo para los administradores de la plataforma.

5.1.4 Capa Cliente

En esta capa el usuario cliente puede acceder a la información almacenada en el cubo multidimensional basado en Tecnologías Semánticas, desde un navegador ingresando a la dirección http://redi.cedia.org.ec/#/es/data/datacube

Basado en la arquitectura propuesta, los componentes definidos en la capa de aplicación fueron automatizados con el fin de procesar todos los datos del REDI utilizando las consultas definidas en este trabajo.

6 PROTOTIPO

En este capítulo se generan dos ejemplos basados en la arquitectura propuesta, para comprobar su validez, presentando los resultados obtenidos.

6.1 IMPLEMENTACIÓN DE LA ARQUITECTURA CON DOS EJEMPLOS

Para la ejecución de los dos ejemplos se trabaja con la información descrita en este trabajo de tesis en el capítulo 0, para continuar con el proceso de visualización.

Ejemplo 1

Para demostrar la utilidad del proceso descrito anteriormente, en este ejemplo se presenta un caso de uso, que a través de la ejecución de una consulta SPARQL se detecta lo siguiente: autores que dispongan 10 publicaciones generadas en el período 2006 -2016, que los autores pertenezcan a tres Instituciones de Educación Superior (Tabla 6.1) acorde a esta información se responden a las preguntas planteadas en la sección 5.1.2

Para publicar en el cubo de datos multidimensionales basados en Tecnología Semántica, es necesario seguir los pasos indicados en el apartado Capa de presentación, en la configuración de la herramienta "*OpenCube Toolkit*".

Autor	Institución	Publicaciones	Período de publicación
Víctor Saquicela	UCUENCA	10	2012 - 2016
Mauricio Espinoza	UCUENCA	10	2006 - 2016
Rodrigo Fonseca	ESPE	10	2006 - 2013
Xavier Ochoa	ESPOL	10	2008 - 2016
Lorena Alvarez	ESPOL	10	2007 - 2016

Tabla 6.1.- Ejemplo1: Datos para la ejecución del cubo multidimensional basados en TS.

1. *¿Cuál es el número de publicaciones realizadas por autor, en un período determinado?* En esta gráfica se aprecia el número total de publicaciones de cada uno de los cinco autores ingresados al cubo por año de acuerdo al período de tiempo indicado en la (Tabla 6.1) (Figura 6.1).

1 - 13 / 13 Show 13 rows (max. 1000) Filter

Anio de Publicacion.	lorena-alvarez	mauricio-espinoza	rodrigo-fonseca	victor-saquicela	xavier-ochoa
2004	-	-	1	-	-
2005	-	-	1	-	-
2006	-	1	-	-	-
2007	1	1	4	-	-
2008	1	-	1	-	1
2009	-	2	2	-	-
2010	-	1	-	-	-
2011	-	1	-	-	2
2012	1	-	1	2	-
2013	3	-	2	-	1
2014	1	2	-	3	3
2015	1	1	-	5	-
2016	1	1	-	-	3

Figura 6.1.- Visualización del número de publicaciones por autor por año

Una vez se encuentre el grafo con toda la información, esta consulta se visualiza el número total de publicaciones que genera cada autor ingresado al grafo, a nivel nacional de esta manera las autoridades y el mismo autor podrá llevar un registro de sus publicaciones y analizar su avance con respecto a los años anteriores.

2. *¿Cuál es el número de publicaciones generadas por cada Institución de Educación Superior, por año?* En esta gráfica se aprecia el número total de publicaciones que cada IES ha generado por año (Figura 6.2).

1 - 10 / 11 Show 10 rows (max. 1000) Filter

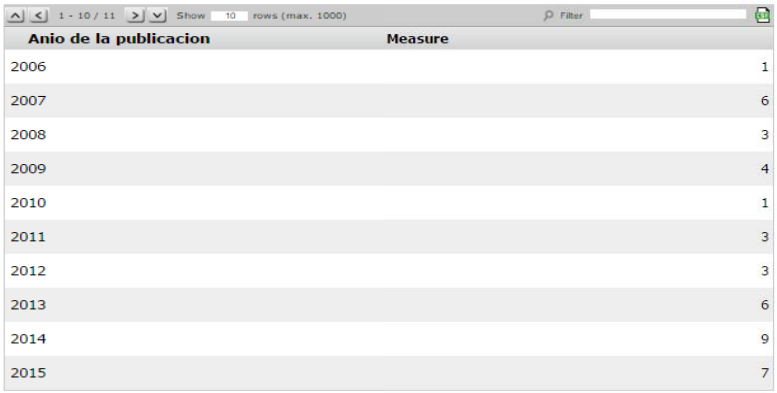
Anio de la publicacion	ESPE	ESPOL	UCUENCA
2006	-	-	1
2007	4	1	1
2008	1	2	-
2009	2	-	2
2010	-	-	1
2011	-	2	1
2012	1	1	1
2013	2	4	-
2014	-	4	5
2015	-	1	6

Figura 6.2.- Publicaciones por IES por año

Con esta consulta, permite a que todas las IES a nivel nacional puedan visibilizar el número de publicaciones generadas por sus investigadores, manteniendo un registro anual de avances del mismo.

Adicionalmente, permitirá generar la información necesaria para procesos de evaluación de las IES ecuatorianas ante los organismos reguladores.

3. ¿Cuál es el número de publicaciones generadas por año? En esta gráfica se puede apreciar el número total de publicaciones almacenadas en el repositorio que han sido generadas por todos los autores escogidos para el ejemplo, por año de acuerdo al período indicado (Figura 6.3)



Anio de la publicacion	Measure
2006	1
2007	6
2008	3
2009	4
2010	1
2011	3
2012	3
2013	6
2014	9
2015	7

Figura 6.3.- Número de publicaciones generadas por año

En esta consulta permite disponer de una visión general del número de publicaciones generadas por autores a nivel nacional y registrar el avance anualmente.

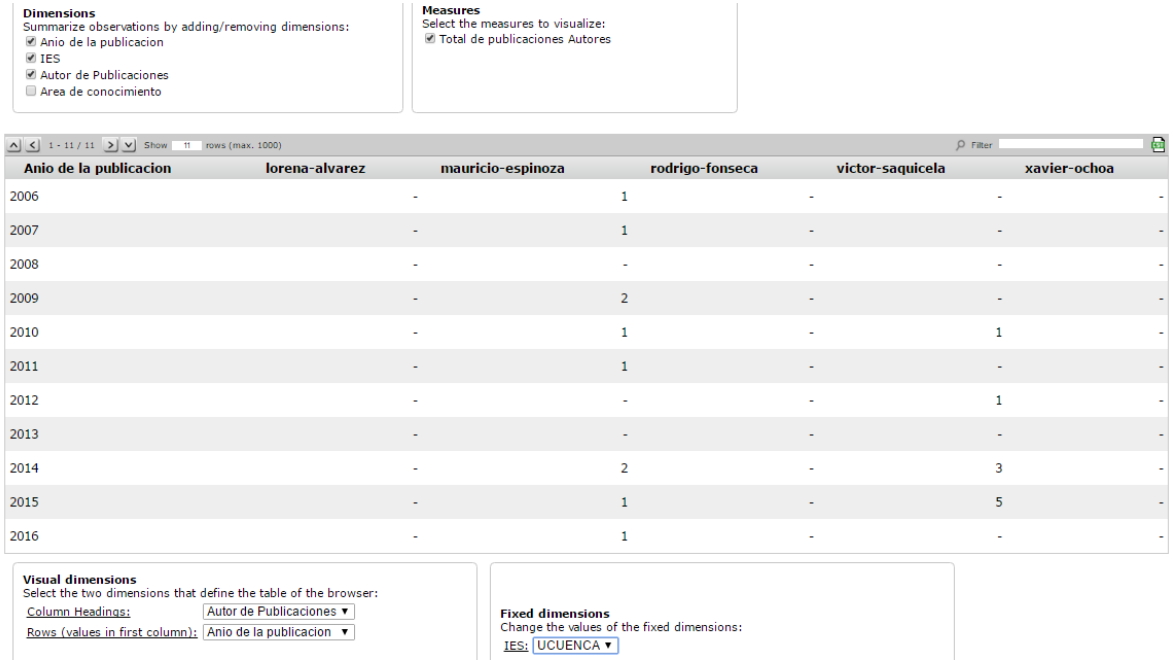
4. ¿Cuál es el número de publicaciones generadas por áreas de conocimiento por autor? En esta gráfica se aprecia el número total publicaciones generadas por los cinco autores ingresados al cubo por área de conocimiento en el período de 2006 - 2016 (Figura 6.4).

Area de conocimiento	lorena-alvarez	mauricio-espinoza	rodrigo-fonseca	victor-saquicela	xavier-ochoa
Digital	-	-	-	1	-
Interlinking	-	1	-	-	-
Ontology localization	-	2	-	-	-
Recognition	-	-	-	1	-
Semantic	-	7	-	5	1
Sensor	-	-	1	-	-
Warehousing	-	-	-	2	-
analysis	1	-	-	-	2
assessment	2	-	-	-	-
audio visual	-	-	-	-	1
ciencia	-	-	-	-	3
computational	-	-	1	-	3
educational	3	-	-	-	-
endocrinology	1	-	-	-	-
genetic	2	-	-	-	-
interlinking	-	-	-	1	-
knowledge	-	-	2	-	-
network	-	-	6	-	-
recognition	1	-	-	-	-

Figura 6.4.- Numero de publicaciones por áreas de conocimiento por Autor

En esta consulta se puede apreciar las áreas de conocimiento en las que se encuentran trabajando los investigadores ecuatorianos y el número de publicaciones generadas en cada una de ellas, permitiendo detectar que áreas de conocimiento predominan en el Ecuador y brindar el apoyo necesario para un área en específico que se desee fortalecer.

5. ¿Cuáles es el número de publicaciones generadas por año, por autor de la Universidad de Cuenca en un período determinado? en esta gráfica se aprecia el número total publicaciones generadas por año, por los autores de la Universidad de Cuenca en un periodo de 2006 – 2016 (Figura 6.5)



Anio de la publicacion	lorena-alvarez	mauricio-espinoza	rodrigo-fonseca	victor-saquicela	xavier-ochoa
2006	-	1	-	-	-
2007	-	1	-	-	-
2008	-	-	-	-	-
2009	-	2	-	-	-
2010	-	1	-	1	-
2011	-	1	-	-	-
2012	-	-	-	1	-
2013	-	-	-	-	-
2014	-	2	-	3	-
2015	-	1	-	5	-
2016	-	1	-	-	-

Figura 6.5.- Numero de publicaciones por institución por autores que pertenecen a la Universidad de Cuenca en el período de tiempo de 2006 – 2016.

Esta consulta permite trabajar con tres dimensiones, generando un nivel de especificidad sobre una IES en concreto, permitiendo disponer de información más detalle de sus autores.

La plataforma *OpenCube Toolkit*, brinda la posibilidad de visualizar la estructura creada del cubo de datos multidimensionales basados en Tecnología Semántica (Figura 6.6 y Figura 6.7)

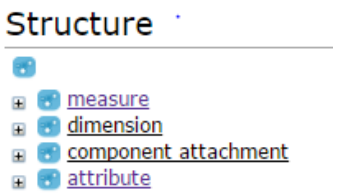


Figura 6.6.- Estructura del cubo de datos multidimensionales basados en TS, Ejemplo 1, 1/2

Figura 6.7.- Estructura del cubo de datos multidimensionales basados en TS, Ejemplo 1, 2/2

Adicionalmente, se puede apreciar un grafo de la información referente a una publicación registrada en el cubo de datos, con información del autor, año de publicación, la institución que se registró dicha publicación, y el área de conocimiento de la publicación, etc.

En la (Figura 6.8) se presentan las dimensiones, medidas y propiedades propia de la publicación denominada "*RDF-ization of dicom medical images towards linkend health data cloud*", como: Autor "*Víctor Saquicela*"; IES a la que pertenece "*UCUENCA*"; Área de conocimiento "Computer Science"; Año de la publicación "2015"; indica que tiene una medida que describe el número total de publicaciones, e indica que es de tipo "Observación". Adicionalmente, indica a que conjunto de datos pertenece.

Figura 6.8.- Grafo con información de una publicación

Ejemplo 2

Para este ejemplo se presenta un segundo caso de uso, reutilizando información del ejemplo anterior, e incorporando dos dimensiones adicionales como "*Ciudad*" y "*Provincia*" que a través de la ejecución de una consulta SPARQL permite visualizar la información propuesta por área geográfica y se agrega un nuevo autor de la Universidad del Azuay. De igual manera, se incorpora

un nuevo vocabulario denominado "*ontology: <http://ucuenca.edu.ec/ontology#>*" para incorporar la información geográfica (Tabla 6.2).

Este ejemplo se realizó de manera local para ser posteriormente implementado en el REDI, en la sección de Cubo de datos en RDF.

Autor	Institución	Ciudad	Provincia	Publicaciones	Período de publicación
Víctor Saquicela	UCUENCA	Cuenca	Azuay	10	2012 - 2016
Mauricio Espinoza	UCUENCA	Cuenca	Azuay	10	2006 - 2016
Verónica Abad	UDA	Cuenca	Azuay	10	1999 - 2016
Rodrigo Fonseca	ESPE	Sangolqui	Pichincha	10	2004 - 2013
Xavier Ochoa	ESPOL	Guayaquil	Guayas	10	2008 - 2016
Lorena Alvarez	ESPOL	Guayaquil	Guayas	10	2007 - 2016

Tabla 6.2.- Ejemplo2: Datos para la ejecución del cubo multidimensional basados en TS

Para realizar el ejemplo 2 se solicita visualizar la siguiente información:

- ¿Cuál es el número de publicaciones que han realizado las Instituciones de educación superior a nivel provincial, desde el año 1999?
- ¿Cuál es el número de publicaciones generadas por área de conocimiento a nivel provincial desde 1999?
- ¿Cuál es el número de publicaciones generadas por los autores de las Instituciones de Educación Superior de la ciudad de Cuenca?

Para visualizar la información es necesario seguir los pasos de configuración de la herramienta "*OpenCube Toolkit*", como se indicó en la Capa de presentación.

1. ¿Cuál es el número de publicaciones que han realizado las Instituciones de educación superior a nivel provincial, desde el año 1999? (Figura 6.9).

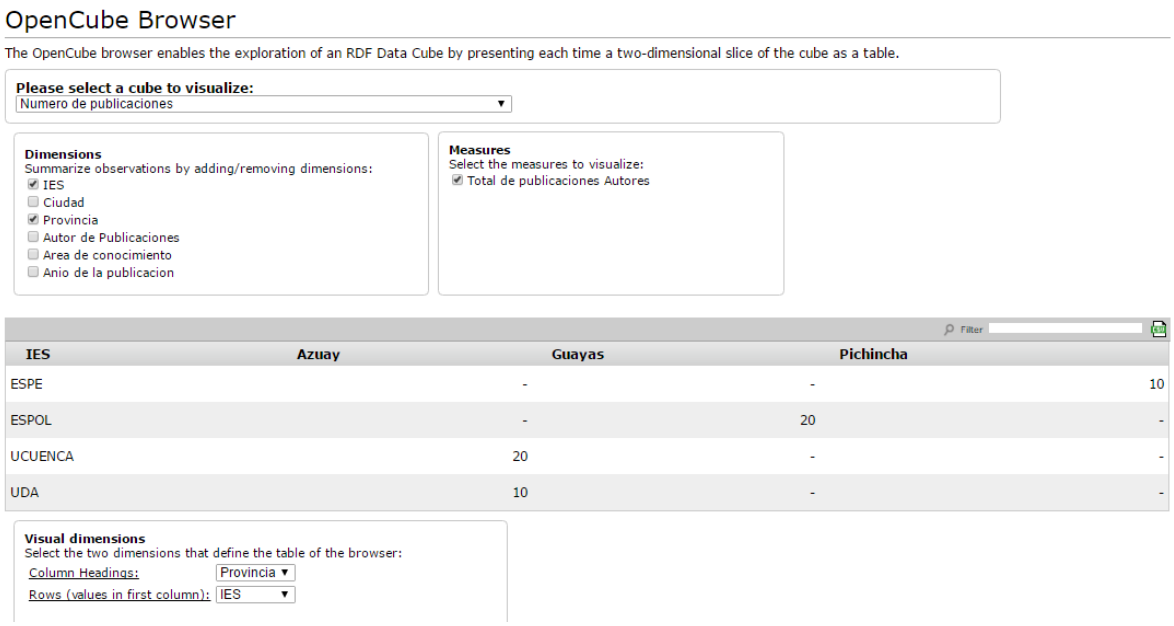


Figura 6.9.- Número de publicaciones realizadas por las IES a nivel provincial desde el año 1999

En la (Figura 6.9) se puede apreciar las publicaciones que han generado los autores pertenecientes a las IES que se encuentran a nivel provincial. De esta manera con todos los datos cargados a la plataforma se puede distinguir que provincia ha realizado más publicaciones.

- ¿Cuál es el número de publicaciones generadas por área de conocimiento a nivel provincial desde 1999? (Figura 6.10).

1 - 18 / 18 Show 18 rows (max. 1000) Filter

Area de conocimiento	Azuay	Guayas	Pichincha
Endocrinology	1	-	-
Ontology localization	2	-	-
Sensor	-	-	1
analysis	-	3	-
assessment	-	2	-
audio visual	-	1	-
ciencia	7	3	-
computational	-	3	1
digital	1	-	-
educational	-	3	-
endocrinology	-	1	-
genetic	-	2	-
interlinking	2	-	-
knowledge	2	-	2
network	-	-	6
recognition	1	1	-
semantic	12	1	-
warehousing	2	-	-

Figura 6.10.- Publicaciones generadas por área de conocimiento a nivel provincial

En la (Figura 6.10) se puede apreciar las publicaciones que han generado los autores pertenecientes a las IES, de acuerdo a las áreas de conocimiento registradas en el modelo multidimensional. De esta manera con todos los datos cargados a la plataforma se podrá distinguir en que área de conocimiento los investigadores ecuatorianos investigan más y generan más publicaciones. De acuerdo a los datos ingresados para el ejemplo, se puede apreciar que en la provincia del Azuay y la provincia del Pichincha existe más trabajo en el sector técnico con el área "*semantic*" y "*network*" respectivamente; en la provincia del Guayas, de acuerdo a las áreas de conocimiento se encuentran trabajando más en el área de ciencias sociales.

- ¿Cuál es el número de publicaciones generadas por los autores de las Instituciones de Educación Superior de la ciudad de Cuenca? (Figura 6.11)

1 - 18 / 18 Show 18 rows (max. 1000) Filter

Area de conocimiento	Veronica-Abad	lorena-alvarez	mauricio-espinoza	rodrigo-fonseca	victor-saquicela	xavier-ochoa
Endocrinology	1	-	-	-	-	-
Ontology localization	-	-	2	-	-	-
Sensor	-	-	-	-	-	-
analysis	-	-	-	-	-	-
assessment	-	-	-	-	-	-
audio visual	-	-	-	-	-	-
ciencia	7	-	-	-	-	-
computational	-	-	-	-	-	-
digital	-	-	-	-	1	-
educational	-	-	-	-	-	-
endocrinology	-	-	-	-	-	-
genetic	-	-	-	-	-	-
interlinking	-	-	1	-	1	-
knowledge	2	-	-	-	-	-
network	-	-	-	-	-	-
recognition	-	-	-	-	1	-
semantic	-	-	7	-	5	-
warehousing	-	-	-	-	2	-

Figura 6.11.- Número de publicaciones por autores de las IES de la ciudad de Cuenca

También, es posible visualizar información especializada de las IES, en este caso se puede apreciar en la (Figura 6.11), el número publicaciones generadas por cada autor perteneciente a la ciudad de Cuenca, de acuerdo a las áreas de conocimiento registradas para este ejemplo.

Adicionalmente, la plataforma *OpenCube Toolkit*, brinda la posibilidad de visualizar la estructura creada del cubo de datos multidimensionales basados en Tecnología Semántica (Figura 6.12 y Figura 6.13), esta estructura es posible revisarla en la plataforma del REDI http://redi.cedia.org.ec/#/es/data/datacube.

Figura 6.12.- Estructura consolidad del cubo de datos basado en TS, Ejemplo 2; 1/2

Figura 6.13.- Estructura consolidad del cubo de datos basado en TS, Ejemplo 2. 2/2

7 CONCLUSIONES Y TRABAJOS FUTUROS

En este capítulo se indican las conclusiones y resultados obtenidos durante el desarrollo del trabajo de tesis, además se plantean nuevas mejoras a la plataforma para un mejor acceso a los usuarios finales.

7.1 CONCLUSIONES

Después de realizar la investigación planteada en el marco del presente trabajo de tesis se puede responder las preguntas de investigación plateadas en el apartado 1.4.

a) De acuerdo a la literatura estudiada, y a la ejecución de los dos ejemplos planteados en el capítulo 6, se puede indicar que si es posible realizar el proceso de transformación de información basada en RDF a Cubo de datos en RDF debido a que se cuenta con la tecnología y conocimiento necesario para realizarlo, continuando el trabajo con Tecnología Semántica y modelos multidimensionales que permiten el desarrollo de una arquitectura de software semántica que determina el proceso de transformación, almacenamiento, extracción de la información del Data Warehouse que a través del uso de herramientas de visualización la información puede ser accesible por parte de los usuarios.
b) La implementación de modelos multidimensionales basados en Tecnologías Semánticas dio como resultado consultas dinámicas, las mismas que permiten usuario obtener mayor información sobre los datos buscados para que estos sean analizados a fondo, en el caso del ejemplo 1 con cuatro dimensiones (autor, área de conocimiento, año de publicación, IES) y con el ejemplo 2 con seis dimensiones (+ ciudad y provincia) y en ambos casos con una medida, obteniendo más información que puede ser manipulada de acuerdo a la necesidad del análisis que realiza el usuario.

Previo al proceso de transformación de RDF a QB es necesario identificar claramente que información se desea visualizar e identificar las dimensiones y medidas a utilizar, con esta información lista, es necesario establecer la estructura de los datos con los que se va a trabajar en el momento de la visualización para proceder a publicarla en el SDW en conjunto con las observaciones.

Basado en este trabajo, actualmente, el proyecto REDI dispone de un *Semantic Data Warehouse*, que a través de la automatización del proceso de transformación de RDF a QB, puede trabajar con modelos multidimensionales basados en Tecnologías Semánticas.

La herramienta de visualización escogida OpenCube Tollkit, permite crear una conexión con la plataforma *Apache Marmotta*, para la extracción de la información almacenada en los grafos generados que permitirán la visualización de los datos en QB. La herramienta permite extraer datos desde diferentes fuentes para este trabajo de tesis se utilizó la opción "SPARQLEndpoint Provider". Adicionalmente, permite exportar los datos almacenados en diferentes formatos (N3, XML, RDF/XML, etc.) e importar datos de igual manera de diferentes formatos. Dispone de diferentes plugins, para trabajar con datos provenientes de archivos como CSV/TSV, bases de datos relacionales, jason, etc. Permite trabajar con la herramienta "R" para estudio estadístico de los resultados. Dispone de un menú para la administración de la herramienta que permite desde crear usuarios, editar la visualización de los resultados, idiomas, realizar búsquedas avanzadas, herramientas de ayuda y solución de problemas.

La tecnología utilizada para el desarrollo del prototipo de una plataforma que permite la visualización de datos multidimensionales basados en Tecnología Semántica, permitió cumplir con los objetivos planteados en este trabajo de tesis.

7.2 TRABAJOS FUTUROS

- En base al trabajo realizado se ha detectado nuevas actividades para mejorar la herramienta, como generar la visualización de la información descrita en este trabajo de tesis de manera geográfica, es decir, visualización la información en un mapa utilizando ontologías como "*geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>*" que permiten utilizar propiedades como "*geo:lat*" y "*geo:long*" para la visualización de la información de manera gráfica.
- Adicionalmente, se encuentra en estado borrador el desarrollo del vocabulario "*QB4OLAP*" descrito en el apartado 3.5, que permite realizar operaciones agregadas como "*sum, min, avg, cout, max*" directamente desde la estructura del vocabulario. Se puede realizar pruebas sobre éste, pero con la limitante de que aún no es un estándar de la W3C.
- La visualización de la información desde la capa cliente, es posible mejorar la presentación haciéndola más intuitiva para el usuario, se puede trabajar con más de un conjunto de datos, generación de *slices* incorporando el número de dimensiones y medidas como co-autores, palabras clave, incorporar más áreas de conocimiento, para que el usuario pueda realizar un detalle minucioso sobre las publicaciones generadas, para que en el momento de buscar investigadores ecuatorianos que trabajen sobre un área de conocimiento en específico disponga de toda la información necesaria.

8 TRABAJOS CITADOS

Acosta Gonzaga, E., Alvarez Cedillo, J. A., & Gordillo Mejía, A. (2006). Arquitecturas en n-Capas: Un Sistema Adaptivo. (I. P. Nacional, Ed.) *Polibits*(34), 34-37. Obtenido de http://www.redalyc.org/articulo.oa?id=402640447007

Aggoume, A., Bouramoul, A., & Kholladi, M. K. (2016). Big Data Integration: A Semantic Mediation Architecture Using Summary. *2nd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2016*, (págs. 21-25). Monastir.

Aghaei, S., Ali Nematbakhsh, M., & Khosravi Farsani, H. (2012). EVOLUTION OF THE WORLD WIDE WEB: FROM. *International Journal of Web & Semantic Technology (IJWesT), 3*(1), 1 - 10. Obtenido de http://search.proquest.com/openview/2a5feb19fa27f1487d772432595f84e5/1?pq-origsite=gscholar

Bayerl, S., & Granitzer, M. (2015). Data-transformation on historical data using the RDF data cube vocabulary. En *Proceedings of the 15th International Conference on Knowledge Technologies and Data-driven Business* (pág. 15). ACM.

Bellatreche, L., Selma, K., & Berkani, N. (2013). Semantic Data Warehouse Design: From ETL to Deployment `a la Carte. En W. Meng, L. Feng, S. Bressan, W. Winiwarter, & W. Song, *Database Systems for Advanced Applications, Proceedings, Part II* (págs. 64-83). Wuhan: Springer.

Bosch, J. (2004). Software architecture: The next step. En *European Workshop on Software Architecture* (págs. 194 - 199). Springer.

Cardacci, D. (2015). *Arquitectura de software académica para la comprensión del desarrollo de software en capas.* Buenos Aires: Universidad del CEMA. Recuperado el ene de 2017, de https://www.econstor.eu/bitstream/10419/130825/1/837816424.pdf

Castells, P. (2003). *La web semántica. Sistemas interactivos y colaborativos en la web.*

CEDIA, R. (2015). Repositorio Ecuatoriano de Investigadores. Cuenca, Ecuador. Recuperado el julio de 2016, de https://www.cedia.org.ec/dmdocuments/FOLLETO%20REDI_digital.pdf

Chaudhuri, S. a. (1997). An overview of data warehousing and OLAP technology. *ACM Sigmod record, 26*(1), 65-74.

Clements, P., Garlan, D., Reed, L., Nord, R., & Stafford, J. (2003). Documenting software architectures: views and beyond. En *ICSE* (Vol. 3, págs. 740 - 741).

Codina, L., & Rovira, C. (2006). La Web Semántica. En J. Tramullas, *Tendencias en documentación digital* (págs. 9-54). Guijón: TREA.

Corchuelo, R. (2007). *Introducción a la Web Semántica.* Obtenido de http://www2.tdg-seville.info/projects/Integraweb/Seminars/semi-19-01-07-material.pdf

Cyganiak, R., Dave , R., & Tennison, J. (2013). *The RDF data cube vocabulary*. Obtenido de http://www.w3.org/TR/vocab-data-cube/

De la Torre Llorente, C., Zorrilla Castro, U., Ramos, M. A., & Calvarro, N. X. (2010). *Guía de Arquitectura de N-Capas orientada al Dominio con .Net (Beta).* España, Microsoft Iberica .

Dourado, A. (2014). An Approach To Publish a Data Warehouse Content as Linked Data. (I. S. Porto, Recopilador)

Espinoza, R. (19 de abril de 2010). *El Rinco del BI, Descubriendo el Business Intelligence…*. Obtenido de Kimball vs Inmon. Ampliación de conceptos del Modelado Dimensional: https://churriwifi.wordpress.com/2010/04/19/15-2-ampliacion-conceptos-del-modelado-dimensional/

Etcheverry, L., & Vaisman, A. (2012). QB4OLAP: a new vocabulary for OLAP cubes on the semantic web., *905*, págs. 27-38. Boston.

Etcheverry, L., Vaisman, A., & Zimányi, E. (2014). Modeling and querying data warehouses on the semantic web using QB4OLAP. En Springer, *International Conference on Data Warehousing and Knowledge Discovery* (págs. 45-56). Springer. doi:10.1007/978-3-319-10160-6_5

Ghasemi, S. (2014). *M2RML: Mapping Multidimensional Data to RDF.* Thesis of Master of Science, Simmon Fraiser University, School of Computing Science, Faculty of Applied Sciences.

Helmich, J. (2013). *Analysing and Visualizing Statistical Linked Data.* Tesis de Maestría, Charles University in Prague, Department of Software Engineering, Faculty of Mathematics and Physics. Recuperado el 23 de octubre de 2016, de https://is.cuni.cz/webapps/zzp/detail/130250/?lang=en

Kämpgen, B. (2015). *FlexiBle integration and eFFicient analysis oF MultidiMensional datasets FroM the WeB.* Karlsruhe, Alemania: KIT Scientific Publishing. doi:10.5445/KSP/1000047013

Kämpgen, B., & Harth, A. (2011). Transforming Statistical Linked Data for Use in OLAP. En ACM, *Proceedings of the 7th international conference on Semantic systems* (págs. 33-40). ACM.

Kruchten, P., Obbink, H., & Stafford, J. (2006). The past, present, and future for software architecture. *IEEE Sotfware, 23*(2), 22 - 30.

Lozano Tello, A. (2001). Ontologías en la Web Semántica. *I Jornadas de Ingeniería Web' 01.* Obtenido de www.anobium.es/docs/gc_fichas/doc/68ERfhjkmv.pdf

Martin, M., Abicht, K., Stadler, C., Auer, S., Ngibga Ngomo, A.-C., & Soru, T. (2015). Cubeviz: Exploration and visualization of statistical linked data. *WWW '15 Companion Proceedings of the 24th International Conference on World Wide* (págs. 2019 - 222). Florencia: ACM. doi:10.1145/2740908.2742848

McBride, B. (2004). The resource description framework (RDF) and its vocabulary description language RDFS. En S. Staab, & R. studer (Edits.), *Handbook on ontologies* (págs. 51-65). Springer Berlin Heidelberg.

Moquillaza, S. D., Vega Huerta, H., & Guerra Grados, L. (2010). Programación en N capas. *Revista de investigación de sistemas e informática - RISI, 7*(2), 57-67. Recuperado el 18 de enero de 2017, de http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3283/2741

Moreno, F. J., & Arango, F. (2007). Estado del Arte de los Modelos Multidimensionales Espacio Temporales. *Avances en Sistemas e Informática, 4*(1).

Nebot, V., Berlanga, R., Pérez, J. M., Aramburu, M. J., & Pedersen, T. B. (2009). Multidimensional integrated ontologies: A framework for designing semantic data

warehouses. En S. Spaccapietra, *Journal on Data Semantics XIII* (págs. 1-36). Springer.

OpenCubeProject. (11 de Noviembre de 2013). *OpenCube Toolkit*. Recuperado el 3 de Noviembre de 2016, de Publishing and Enriching Linked Open Statistical Data for the Development of Data Analytics and Enhanced Visualization Services: http://opencube-toolkit.eu/

Peis, E. H.-V.-M. (2003). Ontologías, metadatos y agentes: recuperación" semántica" de la información. *Actas de las II Jornadas de Tratamiento y Recuperación de la Información (JOTRI)* (págs. 157-165). ornadas de Tratamiento y Recuperación de la Información. Obtenido de http://digibug.ugr.es/bitstream/10481/1206/1/jotri2003.pdf

Perry, D., & Wolf, A. (1992). Foundations for the Study of Software Architecture. *Software Engineering notes. 17*, pág. 40. ACM SIGSOFT.

Posilio Gellida, I. (2014). *Consultas anal´ıticas y visualización para datos abiertos enlazados.* Trabajo Fin de Máster, Universidad Jaime I, Lenguajes y Sistemas Inform´aticos. Recuperado el 22 de 01 de 2017, de http://repositori.uji.es/xmlui/bitstream/handle/10234/138187/TFM_2013_posilioI.pdf?sequence=1&isAllowed=y

Pressman, R. S. (2010). *Ingeniería del Software, un enfoque práctico* (Séptima ed.). Ciudad de México, D.F., México: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.

Reyes Álvarez, L., Hidalgo Delgado, Y., Martínez Rojas, K., Roldán García, M. d., & Aldana-Montes, J. F. (2014). Actualización incremental de grafos RDF a partir de

bases de datos relacionales. En J. Tuya, M. Ruiz , & N. Hurtado (Ed.), *XIX Jornadas de Ingeniería del Software y Bases de Datos*, (págs. 21-26). Cadiz. Recuperado el 25 de Marzo de 2017, de http://sistedes2014.uca.es/Actas-JISBD-2014.pdf

Reynoso, C. B. (2004). Introducción a la Arquitectura de Software. *Universidad de Buenos Aires, 33*.

Rivera Salas, P. E., Martin, M., Da Mota, F. M., Auer, S., Breitman, K., & Casanova, M. A. (2012). Publishing Statistical Data on the Web. *IEEE Sixth International Conference on Semantic Computing. 6to*, págs. 285 - 292. IEEE Computer Society. doi:DOI 10.1109/ICSC.2012.16

Salazar Argonza, J. (11 de Noviembre de 2011). Estado actual de la Web 3.0 o Web Semántica. *Revista Digital Universitaria, 12*(11). Obtenido de http://www.revista.unam.mx/vol.12/num11/art108/art108.pdf

Samper Zapater, J. (2005). *Ontologías para servicios web semánticos de información de tráfico: descripción y herramientas de explotación.*

Senso, J. (2003). Herramientas para trabajar con rdf. *El profesional de la información, 12*(2), 132-139.

Souripriya , D., Seema , S., & Richard , C. (2012). *W3C Recomendation.* Obtenido de R2RML: RDB to RDF Mapping Language: https://www.w3.org/TR/r2rml/

Tamayo, M., & Moreno, F. J. (Diciembre de 2006). Análisis del modelo de almacenamiento MOLAP frente. *Ingeniería e investigación, 26*(3), 135-142. Obtenido de http://www.scielo.org.co/pdf/iei/v26n3/v26n3a16.pdf

Tello, A. L. (2001). Ontologías en la Web semántica. *Jornadas de Ingeniería*. Obtenido de www.anobium.es/docs/gc_fichas/doc/68ERfhjkmv.pdf

Vaisman, A., & Zimányi, E. (2014). *Data Warehouse Systems, Desing and Implementation.* Springer. doi:10.1007/978-3-642-54655-6

Vdovjak, R., & Houben, G.-J. (2001). RDF Based Architecture for Semantic Integration of Heterogeneous Information Sources. *Workshop on information integration on the Web*, (págs. 51 - 57).

W3C. (18 de Agosto de 2009). *SPARQL Lenguaje de consulta para RDF*. Obtenido de https://www.w3.org/TR/sparql11-query/

W3C. (21 de marzo de 2013). *SPARQL 1.1 Update*. Obtenido de https://www.w3.org/TR/2013/REC-sparql11-update-20130321/

W3C. (16 de enero de 2014). *The RDF Data Cube Vocabulary*. (R. Cyganiak, & D. Reynolds, Editores) Recuperado el 31 de octubre de 2016, de https://www.w3.org/TR/vocab-data-cube/

Williams, T. (2014). A primer on converting analysis results data to RDF Data Cubes using free and open source tools. En *Proceedings of 10th Annual PhUSE conference .* Londres.