**­­­­­Instituto Politécnico Nacional**

***Unidad Profesional Interdisciplinaria en***

***Ingeniería y Tecnologías Avanzadas***

Módulo de consultas federadas geoespaciales en el contexto de la Web de Linked Data para el triple store Apache Marmotta

***Alumno:*** Oswaldo Emmanuel Páez Ortega

***Asesores:***

Nombre: Luis Manuel Vilches Blázquez

Procedencia: Centro de Investigación en Computación (CIC)

Nombre: Cyntia Eugenia Enríquez Ortiz

Procedencia: Unidad Profesional en Interdisciplinaria en Ingeniería y Tecnologías Avanzadas (UPIITA)

**Noviembre de 2019**

ÍNDICE GENERAL

[Resumen 5](#_Toc30864654)

[Palabras clave 5](#_Toc30864655)

[Capítulo I: Panorama general 6](#_Toc30864656)

[Introducción 6](#_Toc30864657)

[Planteamiento del problema 6](#_Toc30864658)

[Justificación 8](#_Toc30864659)

[Propuesta de solución 9](#_Toc30864660)

[Alcances (Resultados esperados) 12](#_Toc30864661)

[Metodología 13](#_Toc30864662)

[Objetivo general 14](#_Toc30864663)

[Objetivos específicos 14](#_Toc30864664)

[Capítulo II: Estado del arte 15](#_Toc30864665)

[Trabajos a nivel Internacional 15](#_Toc30864666)

[Querying Geospatial Data over the Web: a GeoSPARQL Interface 15](#_Toc30864667)

[Enabling the Geospatial Semantic Web with Parliament and GeoSPARQL 15](#_Toc30864668)

[Strategies for Executing Federated Queries in SPARQL1.1 15](#_Toc30864669)

[Linking UK Gorvernment Data 15](#_Toc30864670)

[A parallel approach for improving Geo-SPARQL query performance 15](#_Toc30864671)

[DBpedia SPARQL Benchmark – Performance Assessment with Real Queries on Real Data 16](#_Toc30864672)

[Explotación de información en el dominio geo-hídrico ecuatoriano utilizando tecnología semántica 16](#_Toc30864673)

[Sextant: Browsing and Mapping the Ocean of Linked Geospatial Data 16](#_Toc30864674)

[Answering geospatial queries over relational data 16](#_Toc30864675)

[Benchmarking Commercial RDF stores with Publications Office Dataset 17](#_Toc30864676)

[Geographica: A Benchmark for Geospatial RDF Stores 17](#_Toc30864677)

[Geoyasgui: The GeoSPARQL query editor and result visualizer 17](#_Toc30864678)

[Trabajos a nivel Nacional 17](#_Toc30864679)

[Linked Open Data en la Biblioteca Digital Semántica Académica 17](#_Toc30864680)

[Enfoque semántico para el descubrimiento de recursos sensible al contexto sobre contenidos académicos estructurados con OAI-PMH 18](#_Toc30864681)

[Facilitador de contenido móvil para el viajero basado en servicios de localización y Web Semántica 18](#_Toc30864682)

[Trabajos desarrollados en UPIITA 18](#_Toc30864683)

[Recuperación de información geográfica utilizando similitud semántica 18](#_Toc30864684)

[Software similar 19](#_Toc30864685)

Índice de figuras

[Figura 1 Analogía entre MySQL y Apache Marmotta. 8](#_Toc30864622)

[Figura 2 Elementos que contiene Apache Marmotta a considerar en el proyecto. 8](#_Toc30864623)

[Figura 3 Apache Marmotta no soporta consultas federadas. 8](#_Toc30864624)

[Figura 4 Diagrama de Apache Marmotta haciendo una consulta individual (línea sólida) y consultas federadas (líneas punteadas). 10](#_Toc30864625)

[Figura 5 Consulta geoespacial simple. 10](#_Toc30864626)

[Figura 6 Diagrama a bloques de una consulta geoespacial federada. 11](#_Toc30864627)

[Figura 7 Diagrama a bloques de la implementación del módulo de consultas federadas. 11](#_Toc30864628)

[Figura 8 Propuesta de visualización. 13](#_Toc30864629)

Índice de tablas

[Tabla 1. Relación entre variables dependientes e independientes en el *Benchmarking*. 11](#_Toc30864630)

[Tabla 2 Tabla comparativa de software similar a Apache Marmotta. 19](#_Toc30864631)

# Resumen

El presente proyecto terminal se propone diseñar, implementar y caracterizar un módulo de consultas federadas de datos geoespaciales para un *triple store* que no tenga implementado dicho módulo, en específico, a la plataforma Apache Marmotta cuya arquitectura y funcionamiento están basados en los estándares SPARQL y GeoSPARQL. Además, con el propósito de que los usuarios finales visualicen e interactúen con los resultados desplegados por el módulo de consultas federadas geoespaciales, se usó una aplicación Web para que los resultados recuperados de la Web de *Linked Data* pudieran ser visualizados y explorados.

# Palabras clave

Módulo, consultas federadas, *triple store*, datos geoespaciales, Apache Marmotta, *Linked Data*, SPARQL, GeoSPARQL, Web Semántica, aplicación.

Capítulo I: Panorama general

## Introducción

La creación de la Web, llevada a cabo por Tim Berners Lee, y la popularidad que alcanzó, provocó que los usuarios se interesaran en aportar contenido de toda índole en poco tiempo, sin prestar atención a desarrollar un conjunto de buenas prácticas, la cuales sirvieran como referencia para los usuarios al momento de crear y subir contenido a la Web. Debido a esta omisión, la posibilidad de tener una web inteligente se volvería difícil de lograr, esto a consecuencia de que las computadoras no son capaces de interpretar ni de hacer inferencias en el contenido de la Web [1]. Sin embargo, se propuso una evolución que le permitiría a la Web tener un contexto y significado en el contenido que alberga en ella; es aquí donde surgió la Web Semántica. Con esta propuesta se pretendió que el contenido en la Web pudiera ser interpretado por las computadoras a nivel semántico [2]. A partir de este acontecimiento surgieron servidores de *triple store* cuya información que almacenan son tripletas en documentos del tipo Marco de Descripción de Recursos (RDF, por sus siglas en inglés) que describen entidades y relaciones en la Web Semántica, a través de grafos y a su vez, surgieron las plataformas de *Linked Data* (LDP, por sus siglas en inglés) las cuales son herramientas que son capaces de manipular dichas entidades y las relaciones existentes entre ellas.

Las *triple stores* están basadas en el protocolo y lenguaje de consultas para RDF [3], *SPARQL* por sus siglas en inglés, que es el lenguaje estandarizado de consultas para bases de datos de tipo RDF. Este concepto también es utilizado en el dominio de la Web Semántica para datos geoespaciales, donde para la realización de consultas se utiliza el estándar *GeoSPARQL* [4]. Este estándar, en conjunto con tecnologías propias de la Web Semántica, ha sido aplicado en problemas de logística, hidrología, turismo, entre otros [5]. Estos ejemplos, con frecuencia, presentan propuestas donde se realizan consultas únicamente a un *triple store* y éste se encarga de devolver la información almacenada con características geoespaciales. No obstante, existen algunas propuestas donde se han realizado algunos ejemplos *ad hoc* de consultas federadas en el ámbito de los datos geoespaciales [6]. Sin embargo, el estado del arte actual presenta una importante limitación, ya que existen *triple stores* que no permiten la realización de consultas federadas a través de múltiples *triple stores* que presenten de información geoespacial (conforme a *GeoSPARQL*) en el contexto de la nube de *Linked Data*.

## Planteamiento del problema

Se ha demostrado que aproximadamente el 80% de los datos tienen relación con una ubicación geográfica [7]. Esta es una de las razones por la que las herramientas que traten con datos geoespaciales están en constante actualización. El caso de la implementación del *Linked Data*, no es la excepción. Investigadores, empresas y organizaciones gubernamentales usan la nube del *Linked Data* para el estudio y administración de su información [8]. Actualmente existen diversos motores de *triple store* que no son capaces de hacer consultas a diversas fuentes de información geográfica a la vez, es decir, consultas federadas a *triple stores* de datos geoespaciales conforme a los estándares *SPARQL* 1.1 y *GeoSPARQL*.

Existen diversas empresas y organizaciones que se encargan de desarrollar herramientas para la Web Semántica y *Linked Data* para la manipulación y almacenamiento de datos semánticos. Una de estas organizaciones, es la organización sin fines de lucro *Apache Software Foundation* (ASF) la cual ofrece diversas herramientas para diferentes necesidades en cuanto a software se refiere. Ejemplos de estas herramientas son servidores Web, *frameworks*, bases de datos, entre otras. Para el mundo del *Linked Data* y Web Semántica, ASF también tiene su plataforma y es conocida como Apache Marmotta. Para contextualizar a Apache Marmotta, una analogía con bases de datos SQL se muestra en la figura 1.



Figura 1 Analogía entre MySQL y Apache Marmotta.

La plataforma Apache Marmotta cuenta con diversas características, entre las cuales, fueron 3 las relevantes para este proyecto: es una LDP, es un *SPARQL* *endpoint* y también es una base de datos para tripletas RDF, *triple store*. En el caso particular de este proyecto, el trabajo está completamente desarrollado sobre Apache Marmotta. Tal y como se indica en la figura 2, entre los elementos que contiene este *triple store,* no se encuentra un módulo de consultas federadas de datos geoespaciales.



y SPARQL *endpoint.*

Figura 2 Elementos que contiene Apache Marmotta considerados en el proyecto.

La plataforma funciona con los estándares *SPARQL* y *GeoSPARQL* los cuales establecen la manipulación correcta de datos en la nube del *Linked Data* y datos geoespaciales respectivamente. Esto quiere decir que tales características ya existen, pero lo que aún no se había implementado es la capacidad de hacer consultas federadas. Como prueba, la figura 3 muestra una captura de pantalla del sitio oficial de Apache Marmotta[[1]](#footnote-1) encerrado en un rectángulo rojo, indicando que la plataforma aún no era capaz de hacer consultas federadas.



Figura 3 Apache Marmotta no soportaba consultas federadas.

Sin embargo, hubo aspectos que se debieron cumplir para que el proyecto fuera realidad. El primer punto que se abordó es que Apache Marmotta [9] está escrito en Java por lo que se tuvo que comprender cómo es que está diseñado y construido. Las librerías, objetos y el paradigma de programación implementados son ejemplos de los retos que se enfrentaron. Igualmente, se tuvieron que atender los protocolos que fueron indispensables para que Apache Marmotta funcionara junto al módulo desarrollado.

Así mismo, puesto que se escribieron algoritmos y código, dicho código debía ser eficiente y mantenible. El desarrollo fue optimizado a través de la múltiples pruebas llevadas a cabo y que están descritas en el escenario de pruebas.

Otros aspectos que se consideraron fue la funcionalidad del módulo, la cual permitió que las consultas federadas se desplegaran en la nube de *Linked Data* y que el usuario final pudiera recuperar información geoespacial de forma federada. Lo anterior se logró mediante la combinación de respuestas de las diversas fuentes consultadas y el procesamiento de la información para que los resultados no fueran repetidos u omitidos, y la correcta manipulación de datos geoespaciales para poderlos desplegar sobre una aplicación Web. En este sentido, existen diversas alternativas para lograr esta meta, un ejemplo es la herramienta *Map4RDF* [10] cuya función es visualizar y explorar *datasets* RDF cuya información sea geométrica o *GeoYASGUI* [11]que es un editor y visualizador de consultas basadas en *GeoSPARQL*. Por tanto, cabe aclarar que se reutilizará una herramienta de este estilo solo para visualizar los datos recuperados por el módulo de consultas federadas geoespaciales.

De esta forma, la intención del presente proyecto es contribuir a las herramientas *open source* usadas en la Web Semántica. La pregunta que estuvo presente en el desarrollo del proyecto fue: ¿Cuánta información revelará una consulta geoespacial federada frente a una consulta a un único repositorio? Con la intención de cumplir tal propósito la pregunta que surge es: ¿Cuánta información adicional revelará una consulta geoespacial federada frente a una consulta a un único repositorio?

## Justificación

Ante este escenario, el desarrollo e implementación del módulo que solucione dicha problemática fue clave para que el software Apache Marmotta pudiera ser usados de una óptimamente y obtener mejores resultados en las búsquedas.

Un aspecto que se consideró para justificar el desarrollo del módulo propuesto en este proyecto es que las bases de datos no solo son del tipo distribuidas, sino que también existen aquellas que están bajo el esquema de bases de datos federadas. En el escenario de la Web Semántica, a este tipo de bases de datos se les denomina *federated* *triple store*. Con esta característica también se involucra el hecho de que los servidores estén en diferentes localizaciones geográficas, por lo que si no se realizan consultas federada a una *triple store* federada, se estaría obteniendo una porción de información. Por el contrario, si la consulta realizada es federada, se obtendría una respuesta completa que agrupa los resultados de las múltiples *triple store*. Además, considerando la característica de que la información proviene de distintos proveedores de información, el desarrollo de este módulo permitirá que los expertos en las múltiples disciplinas que tiene asociada la información geográfica [6] puedan abordar sus problemáticas con una nueva herramienta que permita integrar y enriquecer sus análisis y estudio.

A pesar de que Apache Marmotta es un software desarrollado por Apache, aún tiene características por incorporar al sistema. Ejemplo de estas características es que al estar basado en el estándar *SPARQL* 1.1, este documento estipula que hay 11 características por cumplir para que un *triple store* se considere completo conforme a esta versión del estándar (*SPARQL* 1.1). Apache Marmotta carece de 2 características: consultas federadas y regímenes de vinculación [12]; las consultas federadas es una característica que se implementó en el presente proyecto terminal con el fin de ofrecer una herramienta *open source* a desarrolladores e investigadores que usen la Web Semántica en sus trabajos e investigaciones sin invertir dinero.

## Propuesta de solución

En la actualidad, existen diversos motores *triple store,* aunque gran parte de ellos [13] solo pueden hacer consultas de forma individual sobre aquellos conjuntos de datos con características geoespaciales que tienen almacenados. Así, la herramienta que se desarrolló en el proyecto terminal propone una solución a este problema brindando la posibilidad de realizar consultas distribuidas a conjuntos de datos geoespaciales presentes en la nube de *Linked Data* para el *triple store* Apache Marmotta. De esta manera, se pretende aprovechar los beneficios que dicha plataforma ofrece para avanzar en el estado del arte y contribuir con el desarrollo y progreso de la Web Semántica geoespacial.

En la figura 4 se muestra con una línea sólida una consulta simple a un solo *triple store*, mientras que, en las líneas punteadas representan la capacidad de consultar diversos *triple store* a la vez, funcionalidad que se le proveyó al software Apache Marmotta en este proyecto.



Figura 4 Diagrama de Apache Marmotta haciendo una consulta individual (línea sólida) y consultas federadas (líneas punteadas).

Descomponiendo la figura 4, en la figura 5 se muestra un diagrama a bloques que detalla el proceso de una consulta geoespacial simple, es decir, una consulta a un solo *triple store*. Con base a los estándares *SPARQL* y *GeoSPARQL*, la consulta en función de los operadores usados, retornará una geometría o un booleano.



Figura 5 Consulta geoespacial simple.

La intención del módulo desarrollado es que las consultas en Apache Marmotta no se centren en un solo *triple store,* sino que consulten a todas las *triple store* que estén disponibles y que presenten información con características geoespaciales. Una vez que se consulten a todas las bases de datos geoespaciales, las respuestas tendrán que ser procesadas de tal forma que no exista datos repetidos. Este resultado será íntegro, ya que contendrá información no solo de un *triple store,* sino que la consulta será respondida con información de diferentes fuentes. La figura 6 muestra un diagrama a bloques de lo mencionado.



Figura 6 Diagrama a bloques de una consulta geoespacial federada.

El desarrollo e implementación del módulo de consultas federadas geoespaciales están basados en los estándares *SPARQL* y *GeoSPARQL*, es decir, el funcionamiento, desarrollo y resultados cumplen con las características estipuladas en los documentos *SPARQL 1.1 Federated Query* [3] y *A Geographic Query Language for RDF Data* [14].

Apache Marmotta está basado en el lenguaje de consultas de la Web Semántica *SPARQL* y es capaz de hacer consultas geográficas *SPARQL* cuya tecnología es mejor conocida como *GeoSPARQL*. El módulo que ha sido desarrollado está integrado al código fuente del motor para lograr que éste sea capaz de hacer múltiples consultas en vez de una sola además de funcionar con datos geoespaciales según el estándar *GeoSPARQL*. La figura 7 muestra una representación de los módulos de Apache Marmotta; en verde se encuentra el lenguaje y protocolo *SPARQL*, en amarillo la extensión *GeoSPARQL* y en rojo se muestra el módulo de consultas federadas implementado.



Figura 7 Diagrama a bloques de la implementación del módulo de consultas federadas.

Para la caracterización del módulo desarrollado, se propuso llevar a cabo un conjunto de pruebas descritas en el escenario de pruebas. Esto a su vez permitió realizar un *benchmarking* contra otras plataformas similares para determinar sus fortalezas y debilidades respecto a dichas plataformas.

Se analizaron 3 artículos que abordan métodos de evaluación en sistemas de consultas federadas. En ellos se hablan de variables independientes y dependientes. Las variables independientes son aquellas características que debe de ser especificadas con el fin de asegurar que sean replicables en los escenarios de evaluación mientras que las variables dependientes son aquellas que serán medidas en la evaluación. La tabla 1 muestra la relación existente entre ellas

Tabla 1. Relación entre variables dependientes e independientes en el *Benchmarking*.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variables independientes | | Variables dependientes | | |
| Tiempo de selección del *SPARQL endpoint* | Tiempo de ejecución | Completitud de la respuesta |
| Consulta | Forma del plan de ejecución | Si | Si | Si |
| Número de patrones de tripletas básicas en la consulta | Si | Si | Si |
| Instancias y posición en las tripletas | Si | Si | No |
| Datos | Tamaño del RDF *dataset* | No | Si | No |
| Características estructurales del *dataset* | No | Si | No |
| Tipo de partición | Si | Si | Si |
| Distribución de datos | Si | Si | Si |
| Plataforma | Memoria RAM | Si | Si | No |
| Número de procesadores | Si | Si | No |
| Administración de memoria caché | Si | Si | No |
| SPARQL endpoint | Número de *SPARQL endpoints* | Si | Si | Si |
| Tipo de *SPARQL endpoints* | Si | Si | No |
| Distribución de las transferencias | Si | Si | Si |
| Latencia de red | Si | Si | Si |
| Retraso inicial del *SPARQL endpoint* | Si | Si | No |

La tabla 1 muestra cuáles son las variables fueron debidamente evaluadas en el *benchmarking* (variables dependientes), y su impacto en las características específicas (variables independientes).Los parámetros considerados en la caracterización son los mismos que se utilizaron para efectuar el *benchmarking*:

* Tiempo de selección del *SPARQL endpoint*
* Tiempo de ejecución.
* Completitud de respuesta.

Los parámetros pueden ser leído con más detalle en los siguientes documentos:

* *FedBench: A Benchmark Suite for Federated Semantic Data Query Processing* [15]
* *SP2Bench: A SPARQL Performance Benchmark* [16]
* *The berlin sparql benchmark* [17]

Con los datos recabados y una vez desarrollado el módulo, se pudo redactar la documentación de la herramienta que permita su correcto uso, así como su compartición con la comunidad relacionada a la Web Semántica y *Linked Data*.

Además, el módulo se complementó con una herramienta para que los usuarios finales pudieran interactuar con los resultados de las consultas geoespaciales federadas. Actualmente ya existen herramientas que permiten visualizar datos geoespaciales, entonces con el único fin de demostrar que el módulo funciona correctamente, por ende, como otra alternativa de comprobación de funcionamiento, este trabajo presenta los resultados obtenidos usando una de esas herramientas que permite desplegar dicha información y analizarla de forma amigable sobre una aplicación Web. Una propuesta de visualización se muestra en la figura 8 donde se aprecian puntos en color azul que representan la información geoespacial recuperada de múltiples *triple store*.



Figura 8 Propuesta de visualización.

## Alcances (Resultados esperados)

Al no haber existido un módulo de consultas federadas geoespaciales para Apache Marmotta, ningunos de los alcances propuestos habían sido elaborados hasta el momento. Los alcances planteados para este trabajo fueron:

* El *triple store* Apache Marmotta en conjunto con el módulo desarrollado es capaz de hacer consultas federadas geoespaciales con base a los estándares *SPARQL* y *GeoSPARQL*.
* Se realizó un *benchmarking* basado en las propuestas descritas en [15], [16] y [17].
* La caracterización está delimitada por los resultados arrojados en el *benchmarking*.
* La comprobación y explotación de datos geoespaciales devueltos por el módulo de consultas federadas en Apache Marmotta se efectuó mediante una aplicación Web donde el usuario puede escoger *datasets* precargados o escribir una consulta federada geoespacial para poder visualizar los datos solicitados.
* Ya que el módulo está actualmente desarrollado, implementado y haer sido puesto a prueba con las pruebas descritas en el escenario de pruebas, se propuso a la organización *Apache Software Foundation* que el módulo sea incorporado en el sistema Apache Marmotta de forma oficial mediante un *pull request* en su repositorio oficial.

**Limitaciones**

* Se necesita una computadora con conexión a Internet, sistema operativo Unix-Linux Ubuntu para instalar Apache Marmotta y el módulo y un navegador web para visualizar la aplicación web.
* El módulo está integrado al código fuente del motor Marmotta por lo que no es un *framework* ni una API. Su uso está completamente dentro del entorno Apache Marmotta.
* ~~Una de las 2 herramientas por usar para la exploración de datos geoespaciales desplegados por el módulo de consultas federadas no será desarrollada, serán~~ *~~Map4RDF o GeoYASGUI~~* ~~las que se utilizarán.~~ La herramienta de visualización, *GeoYASGUI,* no forma parte del desarrollo si no que es un recurso para la visualización de los resultados de las consultas.
* No se busca competir contra los otros *triple store* similares ya que solo se busca incorporar una nueva funcionalidad a Apache Marmotta.
* El proyecto no trata sobre efectuar una búsqueda semántica inter espacial.

## Metodología

La toma de decisión sobre qué metodología se usaría en el proyecto, la decisión fue basada en el artículo de Alfonso Fuggetta, quien propone que el desarrollo de software *open source* debe de llevarse a cabo con metodologías que sean de rápido prototipado cuyo desarrollo sea evolutivo e incremental. Fuggetta propone que las metodologías como espiral y las ágiles son las adecuadas [18].

En la metodología tipo espiral se consideran 4 fases: Determinación de objetivos, análisis de riesgo, desarrollo y prueba y planificación. La ventajas que ofrece esta metodología es que es muy flexible en cuanto a los cambios que requiera el cliente, implementación, reutilización de software e incorporación de objetivos de calidad en el desarrollo del proyecto. Sin embargo, las desventajas que existen en la metodología es que el tiempo de desarrollo es ambiguo ya que al estar haciendo los cambios que el cliente requiera puede nunca terminar y, por ende, si no se realizó un análisis y diseño correcto del sistema, puede negativamente afectar por completo al proyecto.

Las metodologías que Fuggetta también propone son las ágiles. Si bien la más popular es *SCRUM*, se optó por considerar la metodología *Extreme Programmig (XP)* ya que se adaptó al desarrollo del proyecto y al contexto de este. La metodología *XP* se adapta para equipos que cuentan con pocas personas y en proyectos donde el desarrollo del sistema sea de forma incremental.

En *XP* se consideran roles dentro del equipo: Clientes, programadores, *testers*, *coach* y mánager. En esta metodología, al igual que espiral, también se manejan 4 prácticas: Planeación, diseño, codificación y pruebas. Una de las ventajas que la metodología en espiral ofrece, *XP* también lo proporciona y es la característica de ser una metodología recursiva. Cada vez que se terminen las 4 fases, se planea la siguiente etapa del proyecto para que la siguiente etapa se concluyan las historias pendientes y así poder avanzar con las siguientes

*XP* está basada en historias que el cliente propone con ayuda de los *testers*, las cuales son pequeñas tareas que el sistema debe de hacer. Estas historias son desarrolladas por el equipo de programación mediante pruebas unitarias y los *testers* se encargan de validar el correcto desarrollo y funcionamiento de estas. Esta característica de *XP* fuerza al equipo a terminar todas las historias antes de continuar con las siguientes. Cada vez que se terminan las historias correspondientes a la actual fase del proyecto, se procede a integrarlo al proyecto maestro. La práctica de integración evita que los problemas de comunicación e implementación entre interfaces se presenten al implementar cada prueba unitaria aprobada por los *testers* de forma individual en vez de integrar una característica al proyecto por completo.

Ya que *XP* es la metodología que mejor se adaptó a las condiciones y contexto del desarrollo del proyecto, se determinó usar esta metodología.

## Objetivo general

Desarrollar un módulo de consultas geoespaciales federadas para el *triple store* Apache Marmotta, con el propósito de contribuir al avance de las tecnologías usadas en la Web Semántica y proveer una alternativa *open source* diferente a los *triple store* existentes.

### Objetivos específicos

* Implementar, con base en los estándares *SPARQL*, *GeoSPARQL*, así como auxiliándose de otras tecnologías involucradas en la Web Semántica y *Linked Data,* un módulo de consultas federadas para el *triple store* Apache Marmotta*.*
* Evaluar el rendimiento de las consultas hechas por el módulo desarrollado (*benchmarking*).
* Comparar el *triple store* Apache Marmotta con otros *triple store* auxiliándose de la caracterización y *benchmarking* del módulo.
* Caracterizar el módulo de consultas federadas.
* Explotar lar características geoespaciales de los datos obtenidos del despliegue de consultas federadas mediante una aplicación Web para poder visualizarlos y explorarlos.

# Capítulo II: Estado del arte

Los trabajos que se presentan en este capítulo o bien usan las herramientas del *Linked Data* o dan un enfoque de cómo se usan las tecnologías, pero ninguna aborda una propuesta similar a la considerada en este trabajo.

## Trabajos a nivel Internacional

### *Querying Geospatial Data over the Web: a GeoSPARQL Interface*

En este artículo [19] se describe cómo es que Nancy, Ralhp y Dave crearon e implementaron una interfaz para datos *GeoSPARQL* llamada *GeoQuery TOOL*. Esta interfaz intuitiva pretende hacer que las consultas geoespaciales sean más fáciles de hacer al implementar listas en su interfaz para poder escoger atributos y operadores espaciales. Con base en los datos de entrada que haya ingresado el usuario, *GeoQuery* genera código *GeoSPARQL* automáticamente, realiza la consulta usando el *triple store* Parliament y es desplegado en una aplicación web en vez de utilizar el Sistema de Información Geográfica (GIS, por sus siglas en inglés).

### *Enabling the Geospatial Semantic Web with Parliament and GeoSPARQL*

Este trabajo de Robert y Dave [20] se presentan razones por las cuáles hay que usar *GeoSPARQL*, su estado del arte en la industria y en la investigación, y su implementación de *GeoSPARQL* en el *triple store* Parliament. Explican conceptos geoespaciales tales como las diferencia entre una característica y una geometría, qué es un sistema de referencia de coordenadas (CRS, por sus siglas en inglés) y las relaciones topológicas que existen. En esta última mencionan 8 operaciones básicas y sus 2 variantes Egenhofer y RCC8. Ambas expresan las mismas operaciones al ser equivalentes. En general, este documento da las herramientas para comprender *GeoSPARQL*, así como su uso e implementación con la intención de que empresas u organizaciones consideren adoptar esta tecnología.

### *Strategies for Executing Federated Queries in SPARQL1.1*

En esta propuesta [21] se analizan diferentes estrategias para implementar consultas federadas con la intención de evitar los límites que un *endpoint* presenta. Las estrategias que proponen están basadas en la versión de *SPARQL* 1.1 mediante descomposición de consultas federadas. En este artículo se describe la sintaxis de *SPARQL* mediante teoría de conjuntos, así como la evaluación de las estrategias propuestas probando los teoremas propuestos. Por último, los autores muestran la mejora de resultados habiendo implementado sus estrategias.

### *Linking UK Gorvernment Data*

En este trabajo [22] se establecen los casos de uso para la adopción de los principios de *Linked* Data para la publicación de datos públicos del gobierno de Reino Unido. Además, los autores plantean los beneficios de usar *Linked Data*. En sí, el trabajo pretende convencer a empresas, centros de estudio y a desarrolladores a empezar a usar *Linked Data*. En el documento se abordan los temas de datos públicos del gobierno y la responsabilidad que deben existir para su publicación, patrones de diseño, tópicos imprescindibles para abordar la publicación de datos relacionados con información estadística y geoespacial. Por último, presentan las tecnologías disponibles en el contexto *Linked Data* para que desarrolladores de software puedan crear nuevas herramientas.

### *A parallel approach for improving Geo-SPARQL query performance*

Esta investigación [23] expone el problema actual que existe en las consultas geoespaciales que involucran complejas relaciones topológicas y que en conjunto a las bases de conocimiento las cuales no están indexadas, generan ineficiencia en consultas de sus datos. Entonces, en el documento aparte de exponer el problema, proponen una estrategia para disminuirlo, que consiste en el uso de cómputo en paralelo y la creación virtual de índices de la base de conocimiento. En el desarrollo del documento, se muestran el antes y el después de los resultados obtenidos al usar su estrategia sobre consultas en la ciudad de Connecticut, EUA.

### *DBpedia SPARQL Benchmark – Performance Assessment with Real Queries on Real Data*

El documento [24] propone una nueva forma de hacer un *benchmarking* con el propósito de demostrar que los *triples stores* existentes no son tan homogéneos como los otros *benchmarking* lo muestran, es decir, el rendimiento depende de los parámetros usados por lo que una consulta no puede mostrar la misma eficiencia respecto a otra consulta.

Se ponen a prueba 4 *triple store* populares Virtuoso, Sesame, Jena-TDB y BigOWLIM. A lo largo del documento se muestra el desarrollo del *benchmarking*, desde la generación del conjunto de datos de DBpedia, hasta los resultados obtenidos. En este trabajo los autores miden la cantidad de consultas por segundo (QpS, por sus siglas en inglés) en 4 casos: 10%, 50%, 100% y 200% del conjunto de datos generados previamente.

A parte del *benchmarking*, se presenta una discusión donde se habla sobre las fortalezas y debilidades de cada *triple store* al hacer múltiples y simples consultas con ellos. También se presentan trabajos relacionados y los retos por vencer en el campo de los *benchmarking* en *triple stores*.

### Explotación de información en el dominio geo-hídrico ecuatoriano utilizando tecnología semántica

El documento [6] presenta una propuesta de cómo aprovechar los datos geográficos en el dominio geo hídrico. Se apoyan en la única herramienta *open source,* Parliament, para llevar a cabo las consultas geográficas de forma federada.

Presentan el escenario del dominio geo hídrico ecuatoriano y la manera de cómo explotan los datos mediante 3 repositorios de *triple store*.

Cuando terminan de obtener los resultados de las consultas, estos son visualizados con el apoyo de una herramienta llamada MAP4RDF.

Cabe decir que en este trabajo no se hace ninguna implementación o desarrollo sobre un *triple store*, solo utilizan ejemplos de consultas y obtienen datos.

### *Sextant: Browsing and Mapping the Ocean of Linked Geospatial Data*

Es este documento [25] presentan una herramienta Web llamada *Sextant* que permite la exploración de datos enlazados geoespaciales para la creación, compartición y edición colaborativa mediante la combinación de datos geoespaciales enlazados y otro tipo de información disponible en archivos de formato OGC.

### *Answering geospatial queries over relational data*

En este artículo [26] los autores parten del hecho de que los datos geoespaciales son comúnmente almacenados en sistemas manejadores de bases de datos, *DBMS* por sus siglas en inglés, del tipo geoespacial. Para llevar a cabo dicha tarea, se deben de convertir esos datos en datos RDF y almacenarlos en *triple store* cada vez que nuevos datos llegan. Esta labor resulta ser monótona, generando apatía en los administradores de dichas bases de datos para actualizarlas. Si bien existe el paradigma de administración de datos denominada Acceso a Datos Basado en Ontologías, *OBDA* por sus siglas en inglés, que se encargan de ofrecer consultas *SPARQL* en formato SQL, no existe soporte para datos geoespaciales; esto implica no hacer consultas actualizadas o no obtener resultados de consultas geoespaciales. La solución por parte de los autores es habilitar consultas del tipo *GeoSPARQL* *en el aire* al no convertir los datos consultados a RDF y después almacenarlos en un *triple store*. Como resultado, se pueden hacer consultas sin tener que hacer conversiones y almacenamiento, solo puras consultas.

### *Benchmarking Commercial RDF stores with Publications Office Dataset*

Los autores presentan un *benchmark* [27] para RDF *stores* usando *datasets* de la oficina de publicaciones, *PO* por sus siglas en inglés. En la comparación se miden 4 características: *bulkloading*, escalabilidad, estabilidad y ejecución de consultas. De la misma *PO* se utilizaron *datasets* normalizados y no normalizados. Usando lo mismos *datasets* se construyeron nuevos datos para medir la escalabilidad; en estas consultas de dividieron en 2 categorías: consultas instantáneas, consultas que involucran operadores primitivos, y consultas analíticas con el propósito de medir la calidad de servicio y no solo la velocidad. Se concluye que el rendimiento no es homogéneo entre sistemas y que la calidad y velocidad de resultados dependen de diversos parámetros como el tipo de consultas, características de base de datos o *hardware* utilizado.

### *Geographica: A Benchmark for Geospatial RDF Stores*

Se consideró por parte de los autores de este artículo [28], que no existía un *benchmark* que evaluara *triple store* geoespaciales que fuese usado de una manera conocida, por lo que el propósito de haber desarrollado el *benchmark* *Geographica* fue contribuir al estado de arte. En *Geographica* se usaron datos sintéticos y datos del mundo real para probar la funcionalidad ofrecida y el rendimiento de RDF *stores* geoespaciales; se usaron en específico Strabon, Parliament y uSeekM los cuales eran los más completos para la evaluación. El desarrollo de esta comparativa fue con el fin de ofrecer una metodología que permitiera evaluar RDF *stores* de una mejor manera que propuestas previas. Ellos usaron 2 *workloads* de datos: sintético y del mundo real. Con ellos evaluaron eficiencia de funciones primitivas espaciales y el rendimiento de los RDF *stores* en *reverse geocoding, map search y browsing.* Los autores buscan, como trabajo futuro, expandir el *benchmark* para cubrir el estándar *GeoSPARQL* de una manera completa y probar a *Geographica* en un ecosistema centralizado y distribuido.

### *Geoyasgui: The GeoSPARQL query editor and result visualizer*

Los autores abordan una problemática presente en los editores y evaluadores de consultas de *GeoSPARQL* [11]*,* al trabajar con el *Land Registry and Mapping Agency,* mejor conocido como *Kadaster,* quienes publican una gran cantidad de *datasets* entre los cuales son publicados de diversas maneras y en una cantidad considerable, muchos de esos datos son geoespaciales. Cabe decir que *Kadaster* publica sus datos basados el estándar *GeoSPARQL* como *Linked Open Data*. Básicamente lo que hace el sistema es evaluar las consultas al ser enviadas a un *SPARQL* *endpoint* donde se evalúa el álgebra respecto a la colección de datos almacenados en el *endpoint,* esto con el objetivo de optimizar la consulta y después el *endpoint* devuelve los resultados de la consulta en un formato estandarizado (XML, JSON, CSV/TSV). En cuanto a la edición de las consultas, los autores se basaron en trabajos previos: *YASQE, YASR, YASGUI* pero usando los componentes de *GeoSPARQL,* dando así una edición de consultas de datos geoespaciales ofreciendo un visualizador de consultas, un *feedback* directamente al usuario al autocompletar y resaltar sintaxis de la consulta, y un servicio Web que une los elementos anteriores.

## Trabajos a nivel Nacional

### *Linked Open Data* en la Biblioteca Digital Semántica Académica

En este trabajo [29] se describe y analiza cómo es que el *Linked Open Data* es aplicado en las bibliotecas digitales semánticas. En el documento se presenta un marco teórico explicando las tecnologías que son necesarias para el trabajo, tales como XML, RDF, *SPARQL*, *triple store.* Además, se describen las herramientas con las que cuentan en la UNAM para el desarrollo de la biblioteca digital. También presentan iniciativas de *Linked Open Data* en Alemania, Reino Unido y en México. Para concluir su documento, dan una justificación del porque su proyecto es innovador al usar nuevas alternativas de publicación, búsqueda y recuperación de información.

### Enfoque semántico para el descubrimiento de recursos sensible al contexto sobre contenidos académicos estructurados con OAI-PMH

En este trabajo [30], describen un enfoque que considera los recursos de información estructurados con el Protocolo para Cosecha de Metadatos de la Iniciativa de Archivos Abiertos, OAI-PMH por sus siglas en inglés, representación ontológica y su uso en aplicaciones de recuperación de información. Los autores presentan los conceptos a tomar en cuenta como son OAI-PMH, Dublin-Core, sensibilidad al contexto, ontologías. En el trabajo se usó Apache Jena, el cual es un *triple store* donde llevaron a cabo sus pruebas. Se presenta una metodología para consultar y obtener información. Después muestran sus resultados mostrando un grafo que describe la relación entre las instancias de ejemplo que propusieron. Para finalizar, abordan trabajos relacionados, conclusiones y el trabajo a futuro para mejorar su propuesta.

### Facilitador de contenido móvil para el viajero basado en servicios de localización y Web Semántica

En esta tesis [31] desarrollada en el CIC del IPN se propone el diseño e implementación de un facilitador de contenido móvil usando técnicas de la Web Semántica y datos geográficos con *GeoSPARQL* para recomendaciones de alimentación, hospedaje y sitios de interés a turistas. El proyecto se basó en lenguajes de la Web Semántica como RDF, OWL y *SPARQL*. Fue desarrollado sobre un *servlet* de Java en conjunto a servicios basado en geolocalización. Se basó en una metodología dividida en 5 etapas: conceptualización, recuperación de términos, recuperación de información turística y su presentación. Al ser un proyecto que pretende estar disponibles en cualquier dispositivo móvil, se hicieron pruebas experimentales de su servicio web, de la aplicación móvil, de la recuperación de información turística y la forma de presentar los datos. Por último, en la conclusión se hace una comparación del proyecto respecto a una popular aplicación llamada *TripAdvisor,* con la principal característica de que en el proyecto desarrollado muestra resultados precisos, limitados, pero correspondían a la zona donde el usuario se encuentra, mientras que la aplicación comercial en diversos casos devolvía resultados que no estaban relacionados a la consulta hecha.

## Trabajos desarrollados en UPIITA

### Recuperación de información geográfica utilizando similitud semántica

Trabajo hecho por 2 estudiantes [32] donde proponen el uso de un sistema de información geográfica (GIS) y un sistema de geoposicionamiento global (GPS) para identificar sitios de interés alrededor de la zona de donde se encuentre el usuario usando técnicas de similitud semántica, en específico la teoría de confusión. El proyecto fue probado en los alrededores de la IPN Zacatenco para encontrar banco, hospitales y lugares para comer y entretenerse.

## Software similar

Actualmente, no existen más que 4 *triple* *store* que se asemejan en condiciones a Apache Marmotta. Estos cuatro *triple store* implementan *GeoSPARQL* y están actualmente activos. A continuación, se describirán un poco cada uno de ellos

* Parliament
  + Licencia: *BSD License.*
  + Última actualización: mayo 2019.
  + Lenguajes: Java, C++.
* GraphDB
  + Licencia: Comercial.
  + Última actualización: agosto 2019.
  + Lenguaje: Java.
* Openlink Virtuoso
  + Licencia: Comercial.
  + Última actualización: octubre 2018
  + Lenguaje: C.
* Apache Jena
  + Licencia: Apache 2.
  + Última actualización: mayo 2019.
  + Lenguaje: Java.

La tabla 2, mostrada a continuación, se muestran 4 *triple store* que existen en la actualidad. Dichas plataformas son las que se usaron en el *benchmarking* contra *Apache Marmotta.*

Tabla 2 Tabla comparativa de software similar a Apache Marmotta.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre | Consultas Federadas | GeoSPARQL | Status | Libre o pago |
| *Parliament* | Si | Si, en versión 2.7.4 | Activo | Libre |
| *GraphDB* | Si | Si | Activo | Ambos |
| *OpenLink Virtuoso* | Si | Si | Activo | Ambos |
| *Apache Jena* | Si | Si | Activo | Libre |
| *Apache Marmotta* | No | Si | Activo | Libre |

Tal como se mencionó en la justificación del proyecto, desarrollar herramientas que contribuyan al desarrollo de la Web Semántica es primordial, ya que son pocas las plataformas libres para los usuarios que se apegan a los protocolos *SPARQL* y *GeoSPARQL*. Solamente 2 plataformas son completamente libres; los demás *triple store* en la tabla 2 y los que no fueron incluidos en la tabla, son de pago, no se apegan a los protocolos o no tienen las funcionalidades que las plataformas de la tabla 2 ofrecen.

1. <http://marmotta.apache.org/platform/sparql-module.html> [↑](#footnote-ref-1)