



Fundamentación teórica del uso de algoritmos basados en grafos para el cálculo del indicador IRRH

Theoretical basis of the use of graph-based algorithms for calculating IRRH indicator

MSc. Eliober Cleger Despaigne ¹

MSc. Carlos R. Rodríguez Rodríguez ²

DrC. Surayne Torres López ³

¹ Universidad de las Ciencias Informáticas. Cuba.

² Universidad de las Ciencias Informáticas. Cuba.

³ Universidad de las Ciencias Informáticas. Cuba.

Resumen

La herramienta de dirección integrada de proyectos (Xedro-GESPRO) propone reportes basados en una serie de indicadores para medir el avance de los proyectos gestionados. Sobresalen aquellos vinculados al rendimiento y desempeño de los recursos humanos. El sistema calcula sus indicadores mediante funciones SQL en el gestor de base de datos PostgreSQL. Se evidenciaron demoras en la recuperación de datos en aquellas tablas cuyo crecimiento aumenta exponencialmente respecto al resto. Este comportamiento provocó un aumento del costo computacional del cálculo del indicador Índice de Rendimiento de los Recursos Humanos (IRRH). La presente investigación tiene como objetivo fundamentar el uso de algoritmos sobre grafos para disminuir el costo computacional del cálculo del indicador IRRH. Se propuso la aplicación de un modelo basado en grafos para el cálculo de indicadores asociados a la evaluación de recursos humanos en proyectos de software. Se propuso además la característica grafo de propiedad de la base de datos de grafos Neo4J, que se caracteriza por el uso de atributos tanto en los nodos como en las relaciones entre ellos. El grafo de propiedad posibilita además la búsqueda de nodos adyacentes sin necesidad de usar índices, ya que el recorrido se realiza a través de las relaciones y sus atributos. El resultado de la investigación constituye una alternativa para el perfeccionamiento de Xedro-GESPRO.

Palabras clave: Grafos, algoritmos, Neo4j, costo computacional, indicador IRRH



Abstract

The integrated management tool Xedro-GESPRO proposes reports based on several benchmarks to measure the progress of the managed projects. Standing out there are some linked to performance of human resources. The system calculates the benchmarks using SQL functions executed in the Database Management System PostgreSQL. When running the functions some delays are being noticed on retrieving information, especially in those tables which growth increases exponentially respect to the rest. This behavior provoked an increase in the computational cost of calculating IRRH benchmark. What is new in this research is that we proposed a graph based model to calculate the benchmarks related to evaluation of human resources in software projects. We incorporate the use of the graph model proposed by the graph database Neo4J, that uses attributes either in the nodes and in the relations between them as well. The property graph also enables us for adjacent node search without using indexes because the path is made through the relations and their attributes. The result of this research is an alternative for improving Xedro-GESPRO.

Keywords: Graph, algorithms, Neo4j, computational cost, IRRH benchmark

Introducción

Desde el surgimiento de la Universidad de las Ciencias Informáticas (UCI), existieron diversas herramientas para la Gestión de Proyectos de Software (GPSW), las cuales fueron utilizadas hasta el año 2010, fecha en la que se desarrolla la Herramienta de Dirección Integrada de Proyectos (Xedro-GESPRO) (Piñero Pérez, 2010). La herramienta ha incorporado una serie de indicadores para el control de la ejecución de los proyectos (Lugo García, 2012), determinados mediante lógica borrosa. Uno de los indicadores utilizados en la gestión de los RRHH es el relacionado con su Índice de Rendimiento (IRRH), el cual muestra el estado o avance de los integrantes de un proyecto de software ante la realización de las tareas asignadas.

El IRRH es calculado en el sistema mediante cuatro sub-indicadores: índice de rendimiento con respecto al trabajo (IRHT), índice de rendimiento con respecto a la eficacia (IRHE), índice de rendimiento con respecto a su aprovechamiento (IRHA) e índice de rendimiento con respecto a la eficiencia (IRHF). Los sub-indicadores son determinados por diferentes mecanismos de consulta sobre los datos. Su estructuración se basa en el uso del modelo relacional, utilizando para ello el sistema gestor de base de datos (SGBD) PostgreSQL.

Xedro-GESPRO utiliza actualmente la versión 13.05, incorpora un conjunto de reportes necesarios para la toma de decisiones de los directivos, elemento que ha mejorado el nivel de explotación de la herramienta. De igual forma gestiona proyectos con equipos de desarrollo de gran tamaño, con mayor número de tareas dado el nivel de complejidad de los mismos, y ha crecido el volumen de datos que persiste en la base de datos relacional (BDR) diseñada para la herramienta.

El comportamiento antes descrito ha generado un aumento del orden de los segundos a minutos en los tiempos de respuesta para calcular los sub-indicadores del IRHH, influyendo de forma directa en el cálculo general de la evaluación de un proyecto. Se realizó un diagnóstico al cálculo del indicador IRRH y sus sub-indicadores para el proyecto Nova 6.0 del Centro de Soluciones Libre (CESOL) y los proyectos Sistema de Gestión de los Tribunales Populares Cubanos II, Sistema de Gestión de Bufetes Colectivos y



Sistema de Decisiones del Fiscal del Centro de Gobierno Electrónico (CEGEL). Entre los resultados del tiempo de ejecución del cálculo del indicador IRRH se pudo apreciar más de dos minutos en un proyecto de 26 personas y 2103 tareas. Se determinó además que el tiempo de respuesta del cálculo del indicador IRHT es el que mayor incidencia tiene en el resultado final del IRRH. Se analizó el comportamiento del tiempo de repuesta del cálculo del indicador IRHT para los proyectos del Centro CESIM con más de 1000 tareas. A continuación se muestra cómo el aumento del número de tareas del proyecto afecta de forma negativa los tiempos de respuesta asociados al cálculo realizado.

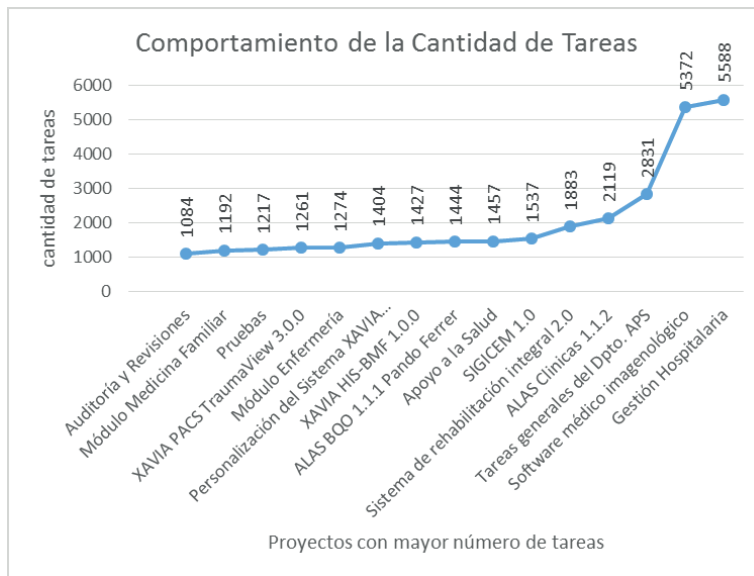


Figura 1: Proyectos con más de 1000 tareas del Centro CESIM. Fuente: elaboración propia

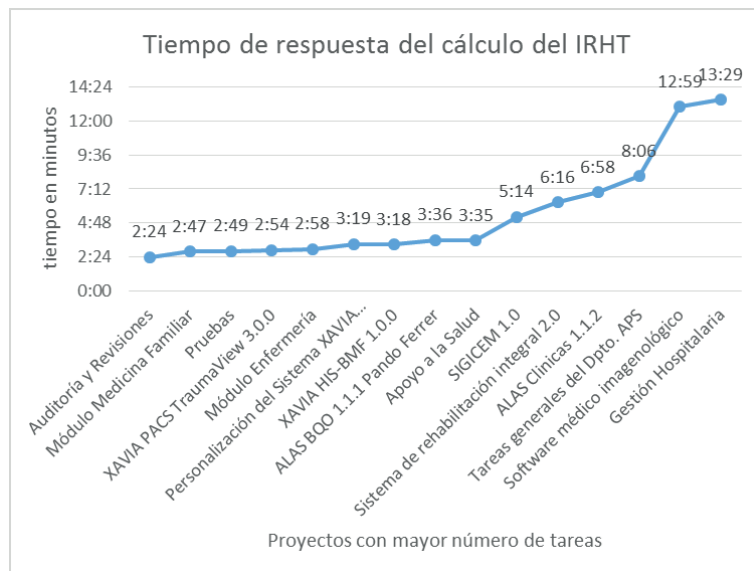


Figura 2: Aumento de los tiempos de respuesta ante el crecimiento del número de tareas. Fuente: elaboración propia

Todas las pruebas fueron realizadas con la asesoría de especialistas del Dpto. de Gestión de Proyectos del Centro CDAE. Las figuras 1 y 2 evidencian que los tiempos de respuestas pueden aumentar más allá de los 2 minutos. Si bien el uso del script GESPROStatBash aun permite la obtención de la línea base de los datos del indicador el lunes de cada semana, se evidencia en el diagnóstico realizado que el costo de dichas operaciones aumenta a medida que aumenta el número de tareas registradas.

Objetivo general: Fundamentar el uso de algoritmos basados en grafos para el cálculo del indicador IRRH definido en la herramienta Xedro-GESPRO, de manera que contribuya a la disminución del costo computacional de la operación. La propuesta está sustentada en la realización de un prototipo funcional que evidenció mejoras en cuanto a la complejidad temporal del cálculo del indicador (Cleger Despaigne, 2017).

Métodos teóricos utilizados

Analítico-sintético: durante el proceso investigativo se realizó un estudio de la bibliografía relacionada con el manejo y la representación de datos asociados a la gestión de proyectos, el cálculo de indicadores para la evaluación de RRHH, la teoría de grafos, el costo computacional, así como de las herramientas que dan soporte a la solución. A partir del análisis realizado se seleccionó una síntesis de lo estudiado.

Entrevista en profundidad. Objetivo: Conocer los indicadores cuyos tiempos de respuestas superan los establecidos a nivel de base de datos.

Experimento: Busca establecer relaciones entre el modelo utilizado para el manejo y representación de los datos y su costo computacional. Se analiza además si la cantidad de tareas del proyecto y las asociadas a la persona influyen en los tiempos de respuesta del cálculo de indicadores relacionados con los recursos humanos en proyectos de software

Representación y manejo de datos asociados a la Gestión de Proyectos

Las bases de datos NoSQL ofrecen soluciones a los problemas que surgen en los sistemas gestores de bases de datos relacionales en escenarios donde radican las redes y el alto crecimiento de información (Camacho, 2010). Las principales características de estas nuevas formas de concebir una base de datos se basan específicamente en su alto rendimiento, su escalabilidad, replicación, y la desestructuración de los altos volúmenes de datos que gestionan. Como aplicación específica al modelado de redes y su posterior análisis, una de las categorías de las bases de datos NoSQL ha cobrado mayor atención por parte de los especialistas: las bases de datos orientadas a grafos (Gracia del Busto & Yanes Enríquez, 2012). Estas representan su información en forma de nodos y relaciones entre ellos, lo que posibilita la realización de diferentes análisis aplicando la teoría de grafos. La principal característica que aporta esta categoría es la conceptualización de los elementos de un grafo (dígase nodos y relaciones) como un dato en sí. Dada esta posibilidad, tanto en los nodos, como en las relaciones, pueden almacenarse diferentes atributos con sus respectivos valores (Robinson, Webber, & Eifrem, 2013). En la siguiente sección se realiza una comparación entre las bases de datos relacionales y las basadas en grafo, resaltando el uso del modelo matemático grafo.



Bases de Datos en Grafo vs BDR

El ejemplo mostrado en las figuras 3 y 4 ilustra un escenario en el que el rendimiento de una BDG es superior a la de una BDR. En ambos casos se tiene la misma información y se pretende extraer todos los trabajadores de una determinada empresa. En el caso de la BDR se muestran las referencias a extraer para poder recorrer las relaciones que hay entre los datos reales. Esta diferencia de rendimientos puede parecer despreciable cuando se trabaja con pocos datos, pero a medida que el volumen se incrementa, esta diferencia se hace notable. A pesar de que probablemente en la primera parte de la consulta la BDG también deba mirar un índice para encontrar el nodo de origen, en el resto de los pasos se hace por uso directo de los punteros físicos que relacionan los nodos, mientras que en la BDR es necesario buscar en al menos un índice (a menudo dos) cada una de las referencias buscadas (lo que habitualmente se hace en $O(\log_2 N)$ si la indexación está correctamente hecha) (Sancho Caparrini, 2014).

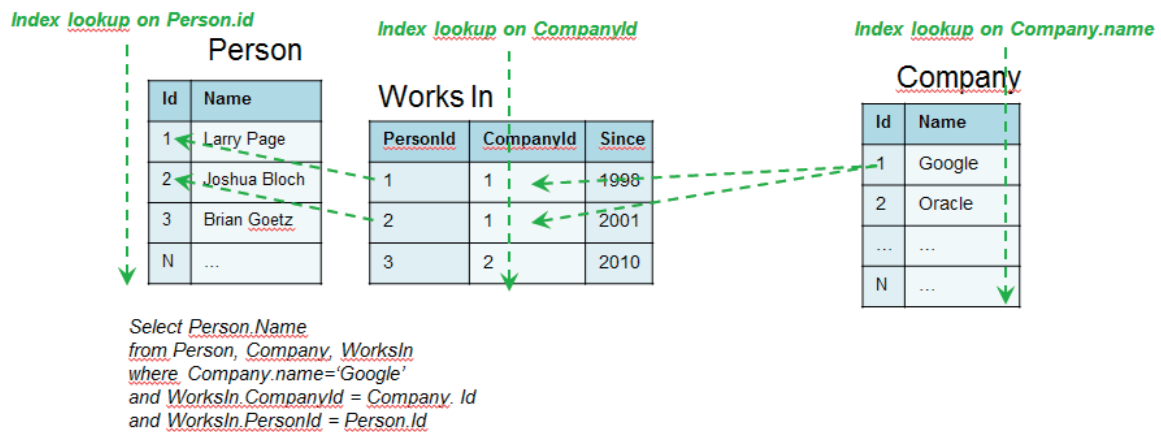


Figura 3: Búsqueda de los trabajadores de Google en la BDR. Fuente: (Sancho Caparrini, 2014)

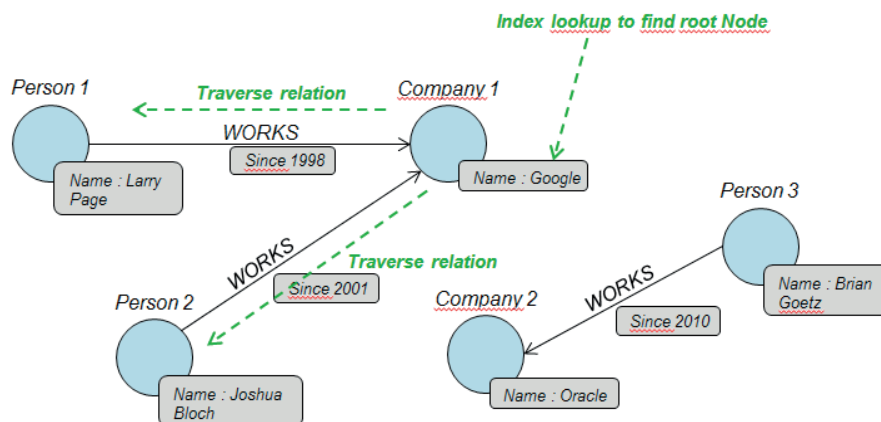


Figura 4: Búsqueda de los trabajadores de Google en la BDG. Fuente: (Sancho Caparrini, 2014)

Modelo basado en grafos

En un proyecto de desarrollo de software generalmente las relaciones se establecen entre personas y los diferentes conceptos de la GPSW tales como hito, rol, tarea, proyecto, metodología, entre otros; y a su vez, entre los mismos RRHH. Para representar las relaciones se propone un modelo teórico conceptual en (Rodríguez Puente & Ril Valentin, 2014). Mediante el modelo propuesto (Ver Figura 5), se pueden realizar diversos análisis relevantes para la toma de decisiones en el ámbito de los RRHH.

- Balance de carga de tareas entre los integrantes del proyecto, propiciando información sobre qué personas tienen una sobrecarga y cuáles pueden asimilar una mayor carga de trabajo.
- Búsqueda de posibles líderes en los equipos de desarrollo.
- Cálculo del desempeño de las personas en diferentes roles, lo cual puede propiciar una eficaz asignación de personas ante un nuevo proyecto y los roles requeridos.
- Análisis de intermediación, exponiendo aquellas personas que jugaron un papel decisivo en hitos importantes del proyecto, entre otros; basados todos en los diferentes análisis que se pueden realizar sobre el modelo matemático grafo.

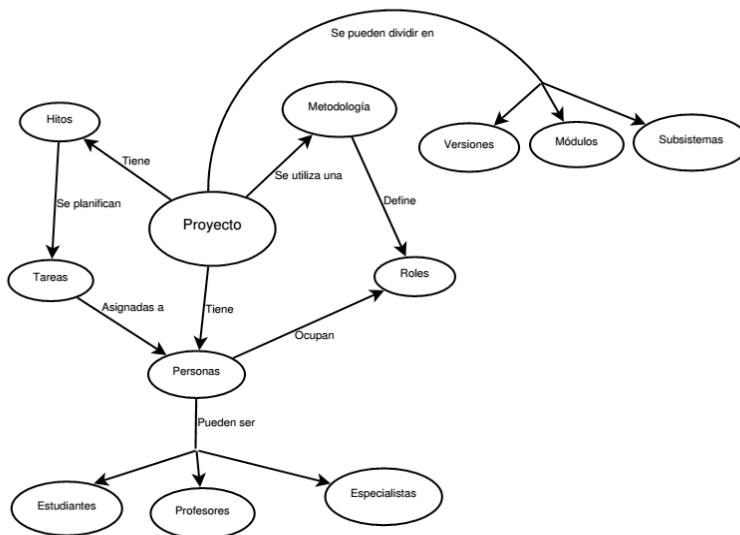


Figura 5: Fragmento del Modelo teórico conceptual para la representación de las relaciones en un equipo de proyecto de software. Fuente: (Rodríguez Puente & Ril Valentin, 2014)

Aunque este modelo recoge los principales conceptos de la gestión de proyectos, representa una abstracción que puede ser modificada en escenarios específicos. Debido a la flexibilidad ante el cambio y a la posibilidad de identificar en el modelo de datos del sistema Xedro – GESPRO gran parte de los elementos aquí expuestos, la propuesta será utilizada como referencia para la propuesta de solución.

Modelo matemático grafo

Como sustento teórico de las bases de datos orientadas a grafos se encuentra el modelo matemático grafo. El mismo define en su composición un conjunto de vértices y aristas, los que representan los nodos y vértices del grafo respectivamente, así como una serie de teoremas para su comprensión y análisis (Wasserman & Faust, 1994). El modelo propuesto tiene particularidades que optimizan el trabajo con dos estructuras de grafos (Van Bruggen, 2014):

- Grafos dirigidos: Los links entre nodos (comúnmente conocidos como relaciones) tienen dirección.
- Grafos multirelacionales: Puede haber múltiples relaciones entre dos nodos iguales. Estas relaciones serán claramente distintas y de un tipo diferente.

Dicho modelo recibe el nombre “Grafo de Propiedad” y será fundamentado en la sección *Herramientas para la gestión de un modelo basado en grafos*.

Análisis de la función SQL para el cálculo del indicador IRHT

Se realiza un resumen del análisis de la función sobre PostgreSQL 9.4 que permite calcular este indicador crítico en el sistema Xedro-GESPRO para un rol de proyecto determinado. El concepto tareas asociadas a la persona constituye el centro de atención para la evaluación del indicador. En la investigación se parte del hecho de que la función o algoritmo de mayor complejidad es el utilizado para calcular el sub-indicador IRHT. Ello se debe a que en el mismo se calcula la correlación de Pearson. Ello permite establecer la correlación entre variables estadísticas almacenadas en columnas de igual tamaño y la desviación estándar de los tiempos estimados y dedicados de las tareas. Se aprecia el uso de varias operaciones de unión (JOIN) cuyo costo aumenta con el crecimiento de las tablas involucradas, con énfasis en las tareas (issues). Las operaciones Join se realizan sobre las mismas dos tablas (issues y work_team_members). Se realizan además llamadas a otras funciones (get_td_petición) y varias subconsultas para dar respuesta a la sentencia inicial. Teniendo en cuenta los aspectos anteriores el tiempo de ejecución de la función es elevado. Se requiere del estudio de otras alternativas que permitan disminuir el costo computacional de la operación.

Análisis de soluciones existentes para la representación de redes basadas en grafos

Se analizan dos herramientas para el análisis y representación de grafos. Cytoscape (Shannon, y otros, 2003) y Gephi (Gephi.org, 2014) son herramientas libres y de código abierto. Cuentan con un conjunto de características que pueden contribuir al logro del resultado esperado y añadir valor agregado al presente trabajo.

Luego del análisis realizado, se determina el desarrollo de un prototipo funcional empleando para ello características importantes de las herramientas antes mencionadas, enunciadas a continuación:



- La infraestructura de carga de los datos del grafo en memoria (caché de datos) utilizada por Gephi. Este mecanismo optimiza la búsqueda de nodos y relaciones en tiempo real, influyendo en el factor temporal.
- Los algoritmos de visualización de un grafo existentes en Gephi, para la organización de los elementos del grafo en las imágenes.
- El proceso de exportación de imágenes utilizado por Cytoscape para la generación de vistas en el grafo.

Una vez definido el modelo teórico conceptual basado en grafos y los componentes reutilizables de las soluciones existentes para el manejo de grafos, se debe seleccionar un gestor de bases de datos adecuado. En los siguientes epígrafes se definen un conjunto de criterios que posibilitarán la selección entre el conjunto de alternativas enunciadas.

Resultados y discusión

Selección de herramientas para la gestión de un modelo basado en grafos

Teniendo en cuenta que se debe desarrollar un modelo basado en grafos con nodos y relaciones, y atributos en ambos, la herramienta para la gestión del grafo debe cumplir con los criterios que se exponen a continuación:

- *Soporte del modelo de grafo de propiedad:* en la Figura 6 puede apreciarse su representación. El modelo expresa el uso de nodos y relaciones para la persistencia de los datos en forma de grafo, incorporando atributos tanto en los nodos como en las relaciones entre ellos. Por esta razón permite:
 - › Representar los datos de manera más natural.
 - › Aplicar múltiples algoritmos de la teoría de grafos.
 - › Encontrar nodos adyacentes sin necesidad de búsquedas indexadas. Esta característica recibe el nombre de adyacencia libre de índice (index free adjacency).
 - › Almacenar pares clave-valor como propiedades de nodos y relaciones.

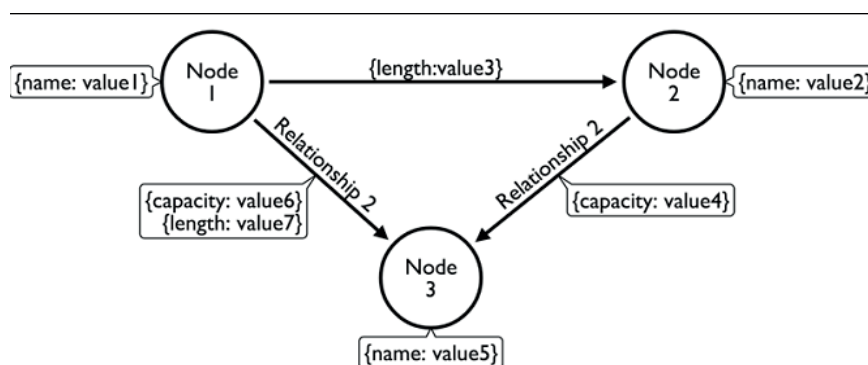


Figura 6 Representación del modelo de grafo de propiedad. Fuente: (Van Bruggen, 2014)

- *Soberanía tecnológica.*
- *Respaldo científico.*
- *Diversas fuentes bibliográficas.*
- *Acreditada por terceros.*
- *Soporte de propiedades ACID por sus siglas en inglés (atomicity, consistency, isolation, durability).*
- *Comunidades y redes sociales.*
- *Uso embebido.*

Teniendo en cuenta los criterios antes mencionados, fueron analizados los gestores de bases de datos orientadas a grafos: AllegroGraph¹, CloudGraph, OrientDB², VertexDB³, Sparksee⁴, Blazegraph⁵, Graph-Base y Neo4j.

En la *Tabla 1* puede apreciarse un resumen del análisis de los criterios expuestos por cada una de las herramientas analizadas. Se concluye que las que mayor número de criterios cumplen son OrientDB, Blazegraph y Neo4j. Se optó por Neo4J para el desarrollo del prototipo, basado además en su integración con el lenguaje de desarrollo seleccionado (Java), su amplia documentación, su sencilla curva de aprendizaje y la utilización de un lenguaje de consulta basado en grafos: Cypher, así como la amplia comunidad de desarrollo.

1 <http://allegrograph.com/allegrograph/>
 2 <http://orientdb.com/orientdb-vs-neo4j/>
 3 <http://www.dekorte.com/projects/opensource/vertexdb/>
 4 <http://www.sparsity-technologies.com/#sparksee>
 5 <http://www.blazegraph.com/>



Tabla 1: Aplicación de criterios de medidas sobre gestores de bases de datos (Cleger Despaigne, 2017). Fuentes: Elaboración propia a partir de (Neo Community, 2015), (Van Bruggen, 2014), (Redmond & Wilson, 2012), (Webling, 2012), (Tesoriero, 2013), (Angles, 2012), (Sikos, 2015), (Jouili & Vansteenbergh, 2013), (Aloci y otros, 2015) y (FactNexus Pty, 2015)

Criterio	Bases de datos orientadas a grafos							
	Neo4j	CloudGraph	OrientDB	VertexDB	AllegroGraph	Sparksee	Blazegraph	GraphBase
Libre								
Código abierto								
Diversas fuentes bibliográficas								
Respaldo científico								
Comunidad y redes sociales								
Acreditación por terceros								
Soporte de propiedades ACID								
Uso embebido								
Soporte del modelo grafo de propiedad								

Neo4J fue creada por la organización Neo Technology, la que actualmente proporciona múltiples servicios orientados al trabajo con dicho producto. Es libre, de código abierto, y tiene una amplia comunidad y aceptación, tanto por sectores académicos como empresariales. Se encuentra desarrollada en Java, y actualmente cuenta con múltiples complementos para el trabajo con otros lenguajes como .NET, PHP, Javascript, Ruby, Python y otros que continúan desarrollando por múltiples colaboradores. Posee implementaciones para los sistemas de Windows, Linux y Macintosh, y las arquitecturas de procesadores de 32 bits y 64 bits (Van Bruggen, 2014).

Almacenamiento físico del grafo de propiedad Neo4J:

Al igual que la mayoría de los archivos de Neo4j, el almacenamiento de nodos se realiza a través de registros de tamaño fijo, donde cada uno tiene 9 bytes de longitud. Los registros de tamaños fijos permiten buscar nodos rápidamente dentro de los mismos. Si se tiene un nodo con id 100 entonces se conoce que su registro empieza a 900 bytes del inicio y por lo tanto la base de datos puede calcular directamente la ubicación del registro con costo $O(1)$ sin realizar una búsqueda de costo $O(\log n)$.



Al igual que el archivo de almacenamiento de nodos, el de relaciones contiene registros de tamaño fijo (en este caso 33 bytes) y cada registro contiene el id de los nodos inicial y final de cada relación, un puntero al tipo de relación y punteros a las cadenas de relación del nodo inicial y final (como una lista doblemente enlazada) ya que una relación pertenece lógicamente a ambos nodos y por lo tanto debe aparecer en la lista de las relaciones de ambos nodos (Robinson, Webber, & Eifrem, 2013).

Lenguaje de consulta propuesto

Definida como base de datos basada en grafos Neo4j, y teniendo en cuenta las múltiples formas de consultar sus datos, se requiere una fundamentación del lenguaje de consulta seleccionado. En (Holzschuher & Peinl, 2013) se realiza un análisis sobre cinco formas de consultar los datos en bases de datos Neo4j: Neo4j embebido, Neo4j REST, Neo4j embebido con Cypher, Neo4j Cypher REST, Neo4j Gremlin REST.

Para la realización de las pruebas fueron empleadas las siguientes herramientas y tecnologías: como base de datos orientada a grafo Neo4j en su versión 1.8, para el consumo de servicios web el servidor *Apache Shindig 2.5 beta* y *Gremlin* del proyecto *Tinkerpop*. Los resultados arrojan que el uso del marco de trabajo embebido (Neo4j embebido) mediante el conjunto de bibliotecas de Java, proporciona un mayor rendimiento en cuanto a tiempo. En la *Figura 7* se ilustra el comportamiento del tiempo de respuesta por cada tipo de petición realizada a la fuente de datos.

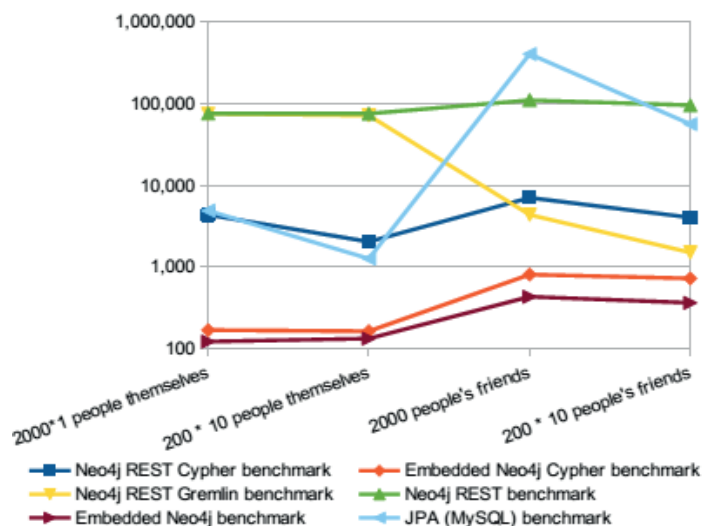


Figura 7 Resultados arrojados de las diferentes formas de consulta de información. Fuente: (Holzschuher & Peinl, 2013)

Por los resultados anteriores, se decide utilizar la forma de acceso embebida (*Embedded Neo4j benchmark*) en la base de datos a conformar, teniendo como principal criterio de selección la búsqueda de la eficiencia en cuanto al costo temporal de dicha actividad. A partir de las tecnologías seleccionadas se cuenta con los recursos necesarios para la realización de un prototipo o herramienta que calcule el indicador IRRH mediante un modelo orientado a grafos.

Conclusiones

Luego del planteamiento de las bases teóricas para el desarrollo de la propuesta, se concluye lo siguiente:

- El modelo grafo de propiedad brinda la posibilidad de recorrer sus nodos y relaciones con adyacencia libre de índices. El costo depende solo del número de pasos del recorrido, optimizando el análisis y la consulta de los datos. Por ello se considera el mecanismo idóneo para estructurar la información asociada al rendimiento de los recursos humanos.
- Se propone el uso del modelo basado en grafos propuesto por Rodríguez Puente y Ril Valentín en 2014 para la representación de información asociada a la Gestión de Proyectos (Cleger Despaigne y otros, 2015).
- Se plantea la necesidad de desarrollar un prototipo funcional que evidencie el correcto funcionamiento de la idea planteada en el presente trabajo.
- Se propone Neo4J como gestor de base de datos basado en grafos con soporte para el modelo grafo de propiedad, el lenguaje de consultas cypher y compatible con las tecnologías Java seleccionadas para el desarrollo del prototipo funcional.

Referencias

- Aloci D, Mariethoz J, Horlacher O, Bolleman JT, Campbell MP, Lisacek F. (diciembre 2015) Property Graph vs RDF Triple Store: A Comparison on Glycan Substructure Search. PLoS ONE, Vol.12. Recuperado de: <https://doi.org/10.1371/journal.pone.0144578>. eISSN: 1932-6203.
- Angles, Renzo. (Septiembre de 2012) A Comparison of Current Graph Database Models. Arlington, VA, USA: 2012 IEEE 28th International Conference on Data Engineering Workshops (ICDEW). Publicada en IEEE Xplore. ISBN: 978-0-7695-4748-0.
- Camacho, Erick. (2010) NoSQL la evolución de las bases de datos. SG Software Guru, Vol. I, No. 28. Mexico. Recuperado de: <https://sg.com.mx/revista/42/nosql-la-evolucion-las-bases-datos>
- Cleger Despaigne, Eliober, y otros. (2015) Modelo orientado a grafos de apoyo a la toma de decisiones sobre los RRHH en proyectos de software. Santo Domingo. República Dominicana. 13th LACCEI Annual International Conference. pág. 8. ISBN: 13-978-0-9822896-8-6.
- Cytoscape. (3 de diciembre de 2014) Cytoscape. Network Data Integration, Analysis, and Visualization in a Box. Recuperado de: http://www.cytoscape.org/what_is_cytoscape.html.
- Cleger Despaigne, Eliober, y otros. (10 de junio de 2017) Modelo Orientado a Grafos sobre Neo4j. Una Alternativa Libre para disminuir la Complejidad Temporal. Iberoamerican Journal of Project Management, Vol. 8, No. 1, págs 1-17. ISSN: 2346-9161.
- FactNexus Pty. (2015) GraphBase. GraphBase - the World's Most Powerful Graph DBMS. Recuperado de: <http://graphbase.net/>.



- Gracia del Busto, Hansel y Yanes Enríquez, Osmel. (septiembre de 2012) Bases de datos NoSQL. Telemática, Vol. 11 No.3, págs 21-33. Recuperado de: <http://revistatelematica.cujae.edu.cu/index.php/tele/article/download/74/74>. ISSN: 1729-3804.
- Gephi.org. (3 de diciembre de 2014) Gephi - The Open Graph Viz Platform. Recuperado de: <https://gephi.github.io/>.
- Holzschuher, Florian y Peinl, René. (2013) Performance of Graph Query Languages. Genoa, Italy: EDBT '13. Proceedings of the Joint EDBT/ICDT. págs. 195-204. ISBN: 978-1-4503-1599-9.
- Jouili, Salim, y Vansteenbergh, Valentin. (enero de 2014) An Empirical Comparison of Graph Databases. Alexandria, VA, USA: 2013 International Conference on Social Computing (SocialCom). Publicada en IEEE Xplore. ISBN: 978-0-7695-5137-1.
- Lugo García, José Alejandro. (2012) Modelo para el control de la ejecución de proyectos basados en indicadores y lógica borrosa. Laboratorio de Gestión de Proyectos, Universidad de las Ciencias Informáticas. La Habana. Cuba: s.n. pág. 79, Tesis de maestría.
- Martínez Cabezudo, Fernando. (diciembre de 2015) Soberanía Tecnológica y Gobierno Abierto. Profundizando en las Necesidades Democráticas de la Participación desde la Tecnopolítica. I Época, Revista Internacional de Pensamiento Político, Vol. 10, págs. 47-70 (54). ISSN: 1885-589X.
- Neo Community. (mayo de 2015) Neo4j.org. Neo4j, the World's Leading Graph Database. Recuperado de: <http://neo4j.com>.
- Piñero Pérez, Pedro Y. y otros. (2010) Herramienta de Dirección Integrada de Proyectos GESPRO. Patente: 1540-2010 Cuba, 29 de junio.
- Robinson, Ian, Webber, Jim y Eifrem, Emil. (2013) Graph Databases. Sebastopol, CA, USA : O'Reilly Media, Inc. pág. 224. ISBN: 978-1-4493-5626-2.
- Rodríguez Puente, Rafael y Ril Valentin, Eliana B. (2014) "Modelado de relaciones existentes en un equipo de proyecto". En actas del evento UCIENCIA 2014. Habana, Cuba.
- Redmond, Eric y Wilson, Jim. (2012) Seven databases in seven weeks. Dallas: Pragmatic Programmers. ISBN: 978-1-93435-692-0.
- Sancho Caparrini, Fernando. (28 de enero de 2014) Sitio oficial de la Universidad de Sevilla, España. Bases de Datos en Grafo. Recuperado de: <http://www.cs.us.es/~fsancho/?e=79>.
- Shannon, Paul, y otros. (22 de agosto de 2003) Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. Genome Research, págs. 2-3. Cold Spring Harbor Laboratory Press. Recuperado de: <http://genome.cshlp.org/content/13/11/2498.full>, ISSN: 2498-2504.
- Sikos, Leslie F. Ph.D. (2015) Graph Databases. Mastering Structured Data on the Semantic Web. Edición 1 págs 145-172, SA, Australia: Apress. ISBN: 978-1-4842-1049-9.
- Tesoriero, Claudio. (2013) Getting Started with OrientDB. Birmingham, UK: Packt Publishing Ltd. ISBN: 1782169962.
- Wasserman, Stanley y Faust, Katherine. (1994) Social Network Analysis: Methods and Applications. Illinois : Cambridge University Press. Vol. 8. ISBN: 0521387078.
- Van Bruggen, Rik. (2014) Learning Neo4j. Birmingham: Packt Publishing Ltd. ISBN: 978-1-84951-716-4.



Webling. (2012) CloudGraph .NET graph database. CloudGraph. Recuperado de: <http://www.cloudgraph.com/>.



Este contenido se publica bajo licencia CC-BY 4.0

