Exploración de datos de Pandora Papers con Neo4j. Caso de estudio: Panamá

Jimmy Fernando Castillo Crespín
Maestría en software, cohorte II
Universidad Técnica de Machala
Machala, Ecuador
jfcastilloc_est@utmachala.edu.ec

***Abstract*—**La extracción de valor en la información es el objetivo principal de la big data debido a que muchas empresas poseen demasiado información de la cual por si solas no generan ningún valor para la empresa., pero estructurándola y relacionándolas entre sí, se obtiene lo que se conoce como la sabiduría de la información **[**1**]**, dicha sabiduría es aprovecha por muchas base de datos relacionales y NoSql para el consumo de aplicaciones pero actualmente las base de datos orientadas a grafos (BDOG) es la que mayormente está en auge de las tecnologías **[**2**]**.

Las BDOG son consideradas también como bases de datos NoSql y proporcionan beneficios como resiliencia, escalabilidad y alta disponibilidad y la más sobresaliente de estas es Neo4j, que es opensource por estar implementando en Java y utiliza Json para estructurar la información en forma de grafos en lugar de tablas [3]. Neo4j ha demostrado grandes hallazgos al momento de analizar terabytes de información filtrada por organizaciones como Wikileaks o Anonimous, uno de los casos más famosos fue el caso de Panamá Papers [4].

Por tal motivo, en este artículo se realizará una exploración de datos del caso de Pandora Papers, utilizando el software Neo4j tomando como caso de estudio el país Panamá para analizar que información relevante se puede obtener de la filtración de este país.

Keywords—Pandora Papers, Neo4j, big data, grafos, BDOG.

I. Introducción

Las empresas viven constantemente en un ambiente competitivo, de los cuales los altos mandos deben tomar decisiones estratégicas si desean permanecer en el mercado y para tomar decisiones acertadas se es necesario aprovechar la información proveniente de sus usuarios y convertirlas en conocimiento, dicha información debe ser oportunas, rápidas y certeras para que la empresa pueda reducir las incertidumbres y riesgos en las decisiones [5].

En la actualidad, muchas empresas siguen utilizando base de datos relacionales o en el peor de los casos, información no procesada de fuentes no estructuradas, como hojas de Excel, cuadernos etc dificultando enormemente el proceso de obtener conocimientos de estas fuentes, agregando labores extra como el diseño e implementación de procesos ETL [6], la obtención de información es exponencial, lo que quiere decir que se llegará a un punto en que cada empresa tendrá demasiada información, naciendo aquí el término datalake donde mucha información se perderá o muchas bases de datos relacionales no tendrán la capacidad ni la eficacia para gestionarlas [7].

Hay que hacer énfasis en el término datalake, que no es más que un repositorio donde se almacena una gran cantidad de datos en bruto y se mantendrá allí el tiempo que sea necesario [8]. Tener datalakes sin aprovecharse es una pérdida económica debido a que está debe estar centralizada en algún servidor consumiendo recursos y también desaprovechando la oportunidad de trabajar con estos datos y obtener información para ser utilizada como estrategias de marketing dentro de los negocios.

Una propuesta de solución para la problemática de tener un datalake sin aprovecharse son las bases de datos orientadas a grafos, dado a su gran capacidad de almacenar cualquier tipo de información en formato JSON, flexibilidad y potencia. Gracias al grafo, se puede realizar consultas más sencillas para obtener información más completa sin mucho esfuerzo y en fracciones de segundos. La información obtenida es mucho más completa para ser utilizada en un ambiente de machine learning y así obtener información valiosa y no erróneas [9].

El objetivo principal de esta investigación es la de realizar una exploración de la información filtrada del país Panamá en Pandora Papers utilizado Neo4j y así poder recopilar información relevante sobre este país.

El siguiente trabajo está estructurado primeramente por el resumen y la introducción que dan las pautas iniciales sobre que se tratará la investigación, seguido del aparto II denominado marco teórico donde se detalla la teoría con la cual se sustentará la investigación, en el apartado III denominado desarrollo y resultados se presentará la propuesta de solución, finalizando con las conclusiones y bibliografía.

II. Marco Teórico

A. *Cypher Neo4j*

Cypher es un lenguaje declarativo, intuitivo y fácil de entender utilizado para realizar consultas en Neo4j [10].

Para representar alguna relación en Cypher se utilizan flechas y las entidades (nodos) se utilizan paréntesis logrando así que las consultas en cypher seán más gráficas, logrando así realizar consultas de tipo where, order by etc tal y como si estuviera utilizando sql [11].

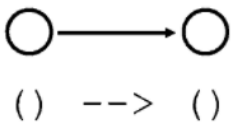

Ilustración 1: Representación de grafos en cypher
Fuente: elaboración propia

B. *Lenguaje Gremlin*

Es la alternativa de cypher, con gremlin se interactúa directamente con el lenguaje Java por lo que brinda la ventaja de que las consultas sean más rápidas que utilizando cypher [12].

C. *Pandora Papers*

Fue una filtración de archivos donde varios proveedores de servicios de empresas y fideicomisos se encontraban registradas en paraísos fiscales, donde los impuestos son muy

bajos y la identidad de los propietarios de dichas empresas se encuentran en anonimato. Son un total de 11’903.676 documentos que llegan a 2.94 teraBytes desde el año 1996 hasta el 2020 [13].

D. Neo4j

Es opensource por estar implementando en Java y utiliza Json para estructurar la información en forma de grafos en lugar de tablas [3].

Entre los comandos más comunes en Neo4j se encuentran:

Crear un nodo

```
CREATE (n)
```

Crear un nodo con label

```
CREATE (n:Product)
```

Crear un nodo con datos y etiqueta

```
CREATE (p:Product {name:'Ipad
Air',price:450}) RETURN p
```

Crear relación simple

```
CREATE
(user1)-[:BUY]->(ipadAir),
(user2)-[:BUY]->(ipadAir)
RETURN ipadAir,user1,user2
```

III. Desarrollo y resultados

A continuación, se detallará el desarrollo de la investigación y los resultados del mismo.

Para este ejemplo práctico, partimos de la creación de un “blank sandbox”

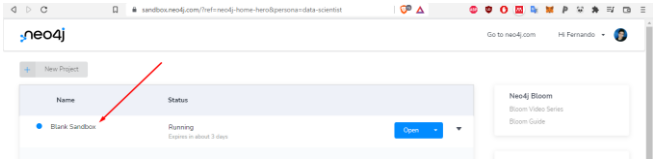

Ilustración 2: Blank Sandbox en Neo4j
Fuente: elaboración propia

Una vez dentro del proyecto recién creado, ejecutamos los dos siguientes comandos para realizar la carga inicial de los datos de Pandora Papers.

Comando #1:

```
call
apoc.load.jsonArray("https://gist.githubusercontent.com/jexp/8afb65325bf99014c8de68a9511a549b/raw/pandora.json")
yield value
with collect(value) as values
unwind range(0, size(values)-1) as set
with apoc.convert.toMap(values[set]) as value, set
unwind value.nodes as n
call apoc.merge.node(n.data.categories, {node_id:
n.data.properties.node_id},n.data.properties) yield node
set node.id=set+"-"+n.id
return count(*);
```

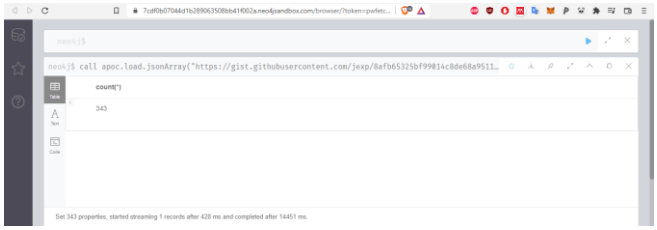

Ilustración 3: Resultado de ejecutar el comando #1
Fuente: elaboración propia

Comando #2:

```
call
apoc.load.jsonArray("https://gist.githubusercontent.com/jexp/8afb65325bf99014c8de68a9511a549b/raw/pandora.json")
yield value
with collect(value) as values
unwind range(0, size(values)-1) as set
with apoc.convert.toMap(values[set]) as value, set
unwind value.edges as e
match (n) where n.id=set+"-"+e.source
match (m) where m.id=set+"-"+e.target
call apoc.create.relationship(n, e.data.type,
apoc.map.clean(e.data.properties,["edge_id","power_player_profile_id"],[]) ,m) yield rel
return count(*);
```

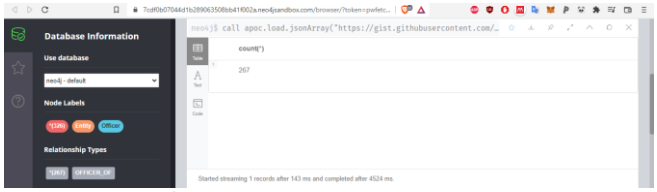

Ilustración 4: Resultado de ejecutar el comando #2
Fuente: elaboración propia

El modelo de datos del pandora papers esta estructura de la siguiente manera:

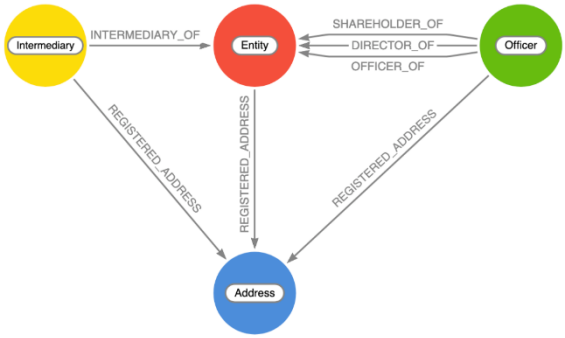

Ilustración 5: Modelo de datos de Pandora Papers
Fuente: [14]

Donde cada nodo representa lo siguiente:

- **Entity**: es la empresa fantasma o construcción offshore.

- **Intermediary**: Grupo de abogados o bancos que ayudaron a las empresas fantasmas.
- **Officer**: propietarios de las empresas fantasmas.
- **Address**: direcciones registradas para los nodos anteriores.

Mostrar todas las empresas fantasmas cuya jurisdicción radique en Panamá

MATCH (e:Entity) WHERE e.jurisdiction CONTAINS "Panama" RETURN e

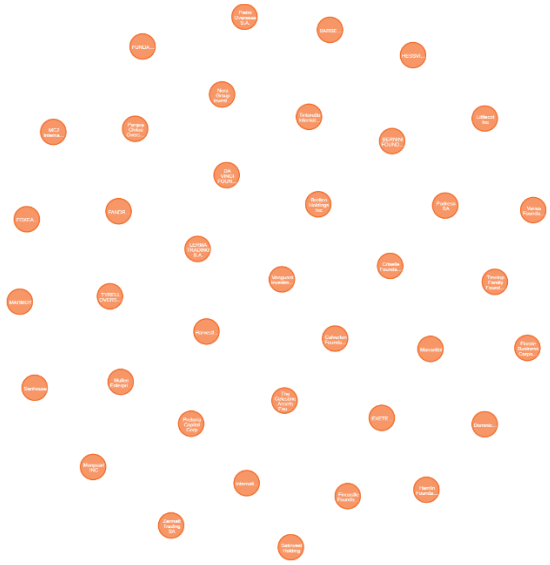

Ilustración 6: Mostrar todas las empresas fantasmas cuya jurisdicción radique en Panamá
Fuente: elaboración propia

Mostrar los propietarios cuya jurisdicción radique en Panamá y que estén relacionados con alguna empresa fantasma.

MATCH (o:Officer)-[rel]->(e:Entity) WHERE e.jurisdiction CONTAINS "Panama" RETURN o, rel, e

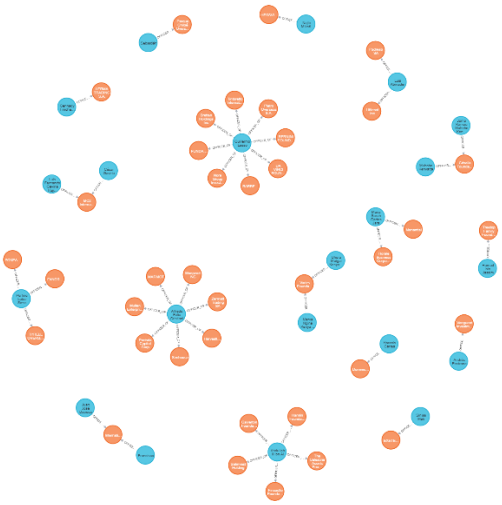

Ilustración 7: Propietarios cuya jurisdicción radique en Panamá y que estén relacionados con alguna empresa fantasma.
Fuente: elaboración propia

Mostrar los más frecuentes proveedores de Panamá

match (e:Entity) WHERE e.jurisdiction CONTAINS "Panama" return e.provider, count(*) as c order by c desc;

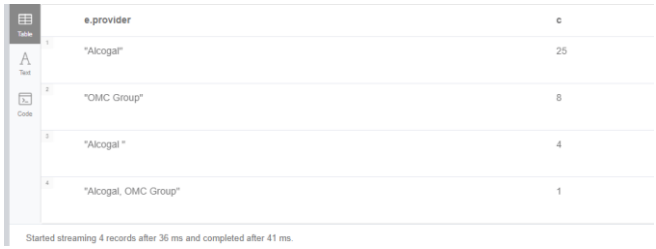

Ilustración 8: Mostrar los más frecuentes proveedores de Panamá
Fuente: elaboración propia

Mostrar cuantas jurisdicciones tiene Panamá

match (e:Entity) WHERE e.jurisdiction CONTAINS "Panama" return e.jurisdiction, count(*) as c order by c desc;

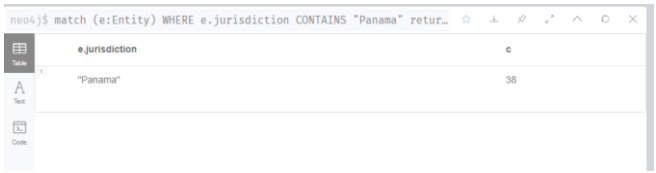

Ilustración 9: Mostrar cuantas jurisdicciones tiene Panamá
Fuente: elaboración propia

Mostrar todos los propietarios cuyo país sea Panamá

MATCH (o:Officer) WHERE o.country CONTAINS "Panama" RETURN o

Ilustración 10: Mostrar todos los propietarios cuyo país sea Panamá
Fuente: elaboración propia

Mostrar todas las empresas fantasmas del presidente de Panamá Juan Carlos Varela.

MATCH (o:Officer)-->(e:Entity) WHERE toLower(o.name) CONTAINS 'juan carlos varela' RETURN *

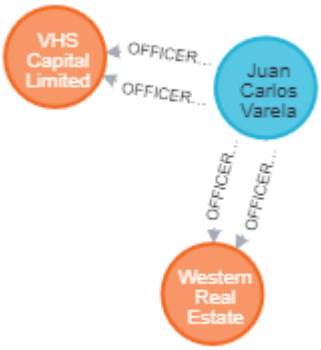

Ilustración 11: Mostrar todas las empresas fantasmas del presidente de Panamá Juan Carlos Varela.
Fuente: elaboración propia

IV. Conclusiones

En esta investigación se realizó una aproximación muy básica sobre la exploración de datos de Pandora Papers con Neo4j, partiendo desde como conectarnos con una fuente externa en este caso un json hasta construir querys para consumir estos datos a través de comandos cypher, obtenido resultados interesantes como cuantas empresas fantasmas tiene el presidente de Panamá, el total de empresas fantasmas que tiene Panamá, todos los propietarios de cada una de estas empresas fantasmas en Panamá entre otras búsquedas relevantes.

Algo a destacar de Neo4j es que no importa si son grafos dirigidos, no dirigidos, sin peso o con peso, la estructura de grafos que maneja esta tecnología lo hace realmente muy sencillo de comprender, en especial para aquellos millones de datos en bruto proveniente de diferentes fuentes como lo fue en el caso de Pandora Papers.

Para esta investigación se llevó a cabo una simple exploración, pero con las bases de datos orientados a grafos, en este caso Neo4j, se pueden realizar mucho más cosas como detección de fraudes, recomendaciones en tiempo real, analizar las interacciones en redes sociales entre otros.

References

[1] Y. Duan, L. Shao, G. Hu, Z. Zhou, Q. Zou and Z. Lin, "Specifying architecture of knowledge graph with data graph, information graph, knowledge graph and wisdom graph," *EEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, pp. 327-332, 2017.

[2] C. C. P. Álvarez, C. Pinilla and M. Bello, "Bases de datos orientadas a grafos," *Tecnología, investigación y academia TIA*, vol. 5, no. 2, 2017.

[3] D. Fernandes and J. Bernardino, "Graph Databases Comparison: AllegroGraph, ArangoDB, InfiniteGraph, Neo4J, and OrientDB," *Proceedings of the 7th International Conference on Data Science, Technology and Applications*, pp. 373-380, 2018.

[4] S. E. McGregor, E. A. Watkins, M. N. Al-Ameen, K. Caine and F. Roesner, When the Weakest Link is Strong: Secure Collaboration in the Case of the Panama Papers, Vancouver: USENIX Association, 2017, pp. 505-522.

[5] Y. RODRÍGUEZ-CRUZ and M. PINTO, "Modelo de uso de información para la toma de decisiones estratégicas en organizaciones de información," *Transinformação*, vol. 30, no. 1, 2018.

[6] S. Vila Guerrero, "Diseño e implementación de procesos ETL con el fin de mejorar la toma de decisiones en una compañía," *Repositorio institucional UPV* , 2019.

[7] L. Alonso Varela and I. Saraiva Cruz, "Búsqueda y evaluación de información: dos competencias necesarias en el contexto de las fake news," *Sedici*, vol. 9, no. 2, 2020.

[8] P. P. Khine and Z. S. Wang, "Data lake: a new ideology in big data era," *4th Annual International Conference on Wireless Communication and Sensor Network* , vol. 17, 2018.

[9] M. A. P. Menvielle, "Búsqueda de patrones en un dominio representado en una base de datos de grafos dirigidos," *CIDS-Centro de Investigación, Transferencia y Desarrollo de Sistemas de Información*, 2018.

[10] N. G. Francis and A. Guagliardo, "Cypher: An Evolving Query Language for Property Graphs," *Association for Computing Machinery*, 2018.

[11] N. Francis, A. Green and P. Guagliardo, "Formal Semantics of the Language Cypher," *francis2018formal*, 2018.

[12] M. J. Núñez Carballo, "Estudio del estado del arte en bases de datos orientadas a grafos," *Universidad ORT Uruguay, Facultad de Ingeniería*, 2021.

[13] E. Universo, "Diario El Universo," 2021. [Online]. Available: https://www.eluniverso.com/temas/pandora-papers/.

[14] M. Hunger, "Exploring the Pandora Papers with Neo4j," 05 10 2021. [Online]. Available: https://neo4j.com/developer-blog/exploring-the-pandora-papers-with-neo4j/. [Accessed 02 12 2021].