Islas Canarias
Del 15 al 19 de noviembre de 2021

Diseño de una Base de Datos Orientada a Grafos para la gestión de datos en iDatos

Jesús Alberto González Yanes
Instituto Canario de Estadística, ISTAC
jgonyanp@gobiernodecanarias.org

Rafael Betancor Villalba
Instituto Canario de Estadística, ISTAC
rbetvilb@gobiernodecanarias.org

Luz Marina Moreno de Antonio
Departamento de Ingeniería Informática y de Sistemas, Universidad de La Laguna
lmmoreno@ull.edu.es

Isabel Sánchez Berriel
Departamento de Ingeniería Informática y de Sistemas, Universidad de La Laguna
isanchez@ull.edu.es

PALABRAS CLAVE
BB.DD. orientada a Grafos, iDatos, estadísticas multifuente

Introducción

En la actualidad, en el marco del Plan Estadístico 2018-2022 se pretende impulsar el Sistema de Datos Integrados (iDatos) con el fin de producir estadísticas multifuente, apoyándose en una gestión eficiente de datos maestros compartidos en múltiples registros, de forma que faciliten el enlazamiento de los diferentes orígenes de datos. Dentro de este plan se han marcado objetivos que potencien el uso de registros administrativos y diversas fuentes de datos. El gran volumen de datos manejado y su continuo crecimiento exige el uso de tecnologías Big Data que garanticen la eficacia y el rendimiento de la solución que se proponga.

En este contexto encontramos que a una misma unidad de análisis le corresponderá datos que están dispersos en distintas fuentes ya sean administrativas u otras fuentes complementarias Big Data. El uso de datos maestros permite determinar un único elemento de referencia y facilita la construcción de las tablas que registran las relaciones entre los diferentes registros. El conjunto de datos maestros que se contemplan son: direcciones, edificios, viviendas y locales, población y hogares, y por

último empresas y establecimientos. La solución actual almacena las relaciones en una base de datos relacional, sin embargo, en los últimos años han proliferado en diferentes contextos las bases de datos orientadas a grafos en las que las relaciones constituyen el elemento crucial en el modelo de datos. El beneficio del almacenamiento nativo de grafos viene dado por la infraestructura de distribución de los datos que se diseña y construye especialmente para tener un buen rendimiento y una alta escalabilidad en el tratamiento de los modelos de grafos, idóneos para la representación de las relaciones. Frente a las bases de datos relacionales y otras soluciones NoSQL, cuando se pretende explotar las relaciones entre datos masivos relacionados hay un aumento evidente de rendimiento.

Este trabajo se ha ocupado del análisis del problema y diseño del esquema de una Base de Datos Orientada a Grafos que de soporte al sistema iDatos. Para obtener conclusiones respecto a la viabilidad e idoneidad del uso de esta tecnología Big Data se ha trabajado con un conjunto de datos de prueba que permite implementar el grafo resultante y diseñar consultas para comparar el rendimiento en una base de datos relacional similar a la utilizada actualmente por el ISTAC y la solución propuesta.

Datos maestros en iDatos

La producción de estadísticas multifuente en el ISTAC se apoya en la tecnología de gestión de datos maestros con objeto de garantizar la calidad de los datos y facilitar el enlazamiento de las diversas fuentes que intervienen en el proceso. Las fuentes a las que se hace referencia corresponden a registros administrativos (padrón municipal, afiliados seguridad social, demandantes de empleo, DIRCE,…) y otras fuentes externas expuestas mediante APIs a través de Internet que complementan y ayudan a mejorar la calidad de los datos. En este apartado se describe la infraestructura de datos maestros que da soporte a la elaboración de dichas estadísticas.

Se utilizan 4 dominios correspondientes a población-hogares, empresas-establecimientos, lugares y edificios-viviendas-locales. Estos dominios se organizan en:

- Directorios: Colección de registros conceptualmente relacionados
- Registros: Colección de ficheros relacionados con una unidad de análisis
- Capas: Colección de ficheros de un registro según funcionalidad de integración
- Variables nucleares: Variables clave de un registro
- Variables normalizadas: Variables que no siendo claves se quieren normalizar en iDatos
- Variables de entidades relacionadas: Identificadores que vinculan unidades de un registro con las de otro registro.

La estrategia de construcción de los registros es siempre similar: se construyen registros de referencia (IDT) a partir de una fuente básica que se enriquece con otras fuentes auxiliares (IDF). La infraestructura que da soporte se encuentra en una base de datos relacional PostgresSQL, con tres clases de tablas para cada registro:

- Tablas DAT: información en registros administrativos
- Tablas IDT: representante único en un registro . El conjunto de tablas IDT representa los datos maestros en la infraestructura iDatos y por lo tanto la verdad en la colección de datos que se manejan en un registro. Estos datos se pueden obtener en registros administrativos y garantizan la consistencia de la información de cada entidad que interviene en una estadística.

- Tablas IDF: complementan un registro a partir de fuentes auxiliares. Recogen cada una de las versiones de la entidad en los registros administrativos o fuentes Big Data que intervienen en una estadística. El diseño de las tablas IDF incluye tanto las variables nucleares como variables estandarizadas que resultarán útiles en la explotación de las fuentes para elaborar las estadísticas. Estos ficheros pueden ir creciendo a medida que se incorporan diferentes registros administrativos o fuentes Big Data en las operaciones.

Las relaciones con los datos maestros se almacenan en tablas URD para una explotación más eficiente en el sistema de bases de datos relacional sobre el que opera actualmente iDatos. Estas pueden ser:

- IDT a IDF: Relaciones del representante único de la entidad con sus diferentes versiones en el directorio.
- IDTD1 con IDTD2 : De elementos de un directorio con los de otro directorio: personas con lugares, personas con empresas, … Constituyen relaciones entre elementos representando la verdad de un directorio con la de otro.
- IDT a las fuentes de datos o datos crudos en cada periodo o instante de tiempo del que se reciben.

De las relaciones se recoge información adicional como la técnica usada para vincular los datos, el tipo y calidad del enlace que genera la relación, etc.

Los directorios contemplados en iDatos son: población y hogares, calles y direcciones y por último, unidades económicas. Los ficheros que intervienen en cada uno de ellos:

- Directorio de Población y Hogares
    - Registro Integrado de Población [de Canarias]
    - Padrón Municipal de Habitantes (PMDH) (Desde 2003 hasta 2021)
    - Demandantes de empleo según SISPE (Sistema de Información de los Servicios Públicos de Empleo) (Desde 2011 hasta 2021)
    - Afiliaciones a la Seguridad Social (Desde 2011 hasta 2021)
    - Afiliaciones a MUFACE (Desde 2011 hasta 2021)
    - Contratos Laborales Registrados (Desde 2011 hasta 2021)
    - Llamamientos Laborales (Desde 2011 hasta 2021)
    - Egresados universitarios (Desde 2015 hasta 2018)
- Directorio de Calles y Direcciones
    - Registro de Portales
    - Fichero de vías y portales del Sistema de Información Territorial de Canarias (SITCAN)
    - Registros administrativos cuyas descripciones de direcciones han sido georreferenciadas con SITCAN y APIs (Google, Komoot, Cartociudad, etc):
    - Padrón Municipal de Habitantes, demandantes de empleo, afiliaciones a la Seguridad Social, afiliaciones a MUFACE, contratos laborales, llamamientos laborales y cuentas de cotización
- Directorio de Unidades Económicas (DUE)
    - Registro Integrado de Empresas [de Canarias]
    - Trabajadores autónomos de afiliaciones en la Seguridad Social
    - Cuentas de Cotización a la Seguridad Social
    - DIRCE (Directorio Central de Empresas)

- Base de datos SABI
- Inventario de Entes del Sector Público (Inventario de Entidades del Sector Público Estatal, Autonómico y Local de la Intervención General de la Administración del Estado)
- Estadística Estructural de Empresas
- Encuesta Industrial Anual de Productos
- Empresas Inscritas en la Seguridad Social

Por simplicidad a lo largo del documento también nos referiremos a estos directorios como Población, Portales y Empresas respectivamente. La estructura de iDatos se representa en la Figura 1, se puede observar como un registro interviene en varios directorios.

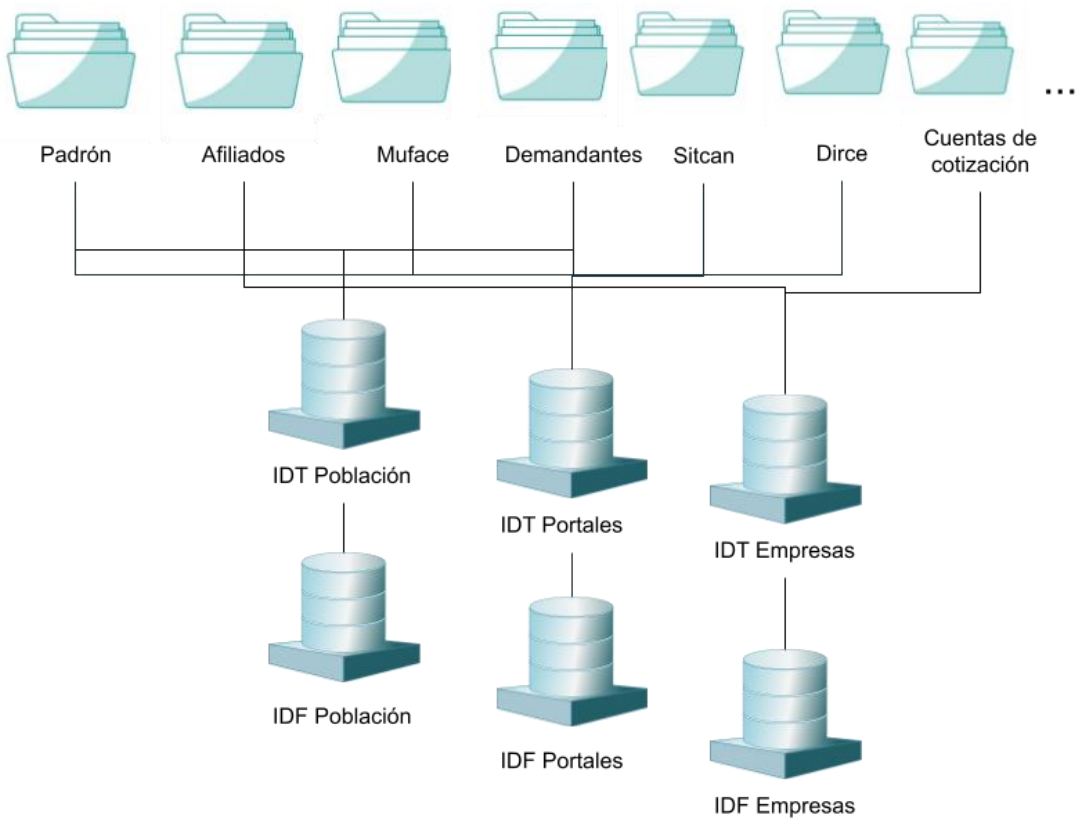

Figura 1. **Arquitectura de iDatos para la elaboración de estadísticas multifuente.**

En el presente trabajo se expone el análisis llevado a cabo sobre la viabilidad y beneficios del uso de un modelo de datos orientado a grafos para la gestión de los datos maestros en iDatos como alternativa a las tablas URD.

Como ejemplo para las pruebas de rendimiento se recurre a la Estadística de Población Activa Registrada (EPA-Reg) que ya usa el sistema para ofrecer datos de la población activa a escala municipal y submunicipal. Esto se logra gracias a la referenciación geográfica e integración de datos administrativos del Padrón Municipal de Habitantes, las afiliaciones a la Seguridad Social o Muface, y las demandas en el Servicio Canario de Empleo.

En los siguientes epígrafes se detalla la pertinencia del uso de las bases de datos orientadas a grafos, los objetivos que se marcaron, el conjunto de datos que ha servido

para desarrollar la prueba de concepto, el diseño del grafo que dará soporte a los datos maestros, el diseño de las pruebas realizadas para comparar ambos sistemas y los resultados obtenidos.

Bases de datos orientadas a grafo vs bases de datos relacionales en la gestión de datos maestros.

Una base de datos orientada a grafos es un sistema que usa como modelo de datos un grafo y cuenta con métodos CRUD (Create, Read, Update y Delete). El modelo de datos que se sigue en este tipo de bases de datos son los grafos, en los que los nodos cuentan con propiedades y las aristas representan relaciones, que también pueden tener propiedades. En caso de que se necesite categorizar los nodos pueden estar etiquetados por una o más etiquetas. De este modo se recoge la semántica según algún rol que jueguen en el conjunto de datos. Las relaciones definen la estructura del grafo, siempre empiezan en un nodo y terminan en un nodo. Este conjunto de primitivas permite establecer el contexto semántico de forma similar a la concreción en una instancia en la base de datos del modelo que se define.

Este tipo de bases de datos son especialmente útiles en grandes conjuntos de datos conectados, en los que las consultas que requieren el uso de joins bajan el rendimiento a medida que el conjunto crece. El uso intensivo del operador JOIN genera problemas de rendimiento, consultas que se deban resolver con más de 3 saltos requieren mucho tiempo de cómputo, que puede llevar incluso al bloqueo del sistema. En estos casos, el rendimiento en una base de datos orientada a grafos permanece constante, ya que el tiempo de ejecución es proporcional al tamaño del subgrafo que da respuesta a la consulta, y no al tamaño del grafo completo.

Las bases de datos orientadas a grafos que usan procesamiento nativo de grafos se basan en índices libres de adyacencia, donde los nodos conectados apuntan unos a otros físicamente, lo que se traduce en ventajas significativas en el rendimiento (Robinson, I., Webber, J., & Eifrem, E., 2015) .

La mayoría de los sistemas de gestión de datos maestros heredados se basan en una base de datos relacional que no está optimizada para consultar datos enlazados con una capacidad de respuesta rápida. Las bases de datos orientadas a grafos representan una solución idónea para modelar, almacenar y consultar las relaciones entre los datos maestros, los metadatos y las jerarquías. Por otra parte, las relaciones del grafo conectan fácilmente los datos en los diferentes silos de la organización proporcionando consistencia a los datos de la organización.

La ventaja de utilizar bases de datos orientadas a grafos para la gestión de datos maestros viene de su capacidad para resolver consultas sobre relaciones que son muy costosas para los sistemas de bases de datos relacionales, especialmente cuando conllevan varios saltos. Otras ventajas adicionales son la capacidad de complementar los sistemas existentes, sin necesidad de remplazarlos, la simplicidad del lenguaje de consultas, la posibilidad de aplicar algoritmos de aprendizaje basados en grafos y la facilidad de visualizar y comunicar la estructura (Mathur, N., 2021).

Neo4j

Neo4j es un sistema gestor de base de datos open source que implementa como modelo de datos un grafo con una arquitectura de clusters distribuidos, altamente escalable. Es

una base de datos que permite transacciones ACID implementada en Java y Scala y puede ser utilizada tanto en una instalación propia, en la nube con Neo4j Aura, como en cualquier plataforma nativa de nube. Además proporciona herramientas visuales que facilitan la consulta y administración de los datos, APIs RESTful y Java entre otros. Dispone del lenguaje de consulta Cypher similar a las consultas SQL pero con expresividad adaptada a las estructuras en el grafo. Neo4j tiene soporte para los lenguajes de programación: .Net, Clojure, Elixir, Go, Groovy, Haskell, Java, JavaScript, Perl, PHP, Python, Ruby y Scala. Neo4j cuenta con una amplia comunidad y documentación disponible.

Según el ranking de DB-Engines Neo4j es la base de datos orientada a grafos más utilizada actualmente y la número 19 del total analizado en el ranking (DB-Engines, 2021)

Los elementos que se usan en Neo4j para modelar los datos son: nodos, relaciones, propiedades y etiquetas. Los nodos representan entidades, elementos de un dominio. Las relaciones indican una conexión entre un nodo origen y uno destino, son las aristas del grafo. Se pueden utilizar propiedades tanto en relaciones como en nodos que permiten especificar en mayor detalle el elemento del grafo mediante pares clave-valor. Las etiquetas se aplican a los nodos para clasificarlos, un nodo admite más de una etiqueta.

La herramienta de visualización del grafo de Neo4j representa los nodos mediante círculos, las etiquetas se identifican por colores, las relaciones mediante aristas. Por otra parte, en el lenguaje de consultas Cypher la sintaxis sobre consultas que implican saltos permanece más limpia que en SQL, se enfoca sobre las conexiones entre nodos y de una manera más visual.

Objetivos

El objetivo se ha seguido para la realización de este trabajo consistió en diseñar el grafo que sirva de esquema en la base de datos orientada a grafos Neo4j a los datos maestros en iDatos, así como la implementación del mismo en una prueba de concepto con un conjunto de datos reales o simulados en los casos en que estos no sean publicables. La propuesta incluye la extracción de una muestra de datos en los ficheros estadísticos que sea representativa del sistema iDatos, que permitirá replicar la complejidad de las consultas que se resuelven actualmente a través de esta infraestructura. La instancia obtenida servirá para analizar el rendimiento de consultas relacionadas con la operación estadística EPA-Reg. También se estudiarán relaciones que no estén explícitas en las tablas URD pero que se puedan inferir gracias al uso de la base de datos orientada a grafos.

Metodología

El trabajo se ha desarrollado utilizando una muestra de datos anonimizados en los registros de población y hogares, calles y direcciones y unidades económicas en iDatos en los meses de junio y septiembre del año 2017. La muestra se ha generado seleccionando una semilla aleatoria en las tablas IDT de cada uno de los registros. Se ha implementado un algoritmo que recorre el conjunto de tablas IDF y ficheros en el registro de forma que se agregan nuevos casos a la muestra que permitan encontrar casos enlazados. Estos enlaces provienen de cada una de las relaciones URD en las que interviene el caso semilla. De esta forma se cuenta con casos de relaciones:

$IDT_{DIRECTORIO\_A}$-$IDF_{DIRECTORIO\_A}$, $IDT_{DIRECTORIO\_A}$-Datos $_{DIRECTORIO\ A}$. En una segunda fase se amplía la muestra para garantizar la existencia de instancias de relaciones complejas provenientes de enlaces transitivos. Para este trabajo se incluyen casos:

Datos Raw $_{DIRECTORIO\ A}$-**$IDT_{DIRECTORIO\_A}$**
$IDT_{DIRECTORIO\_A}$- $Datos_{DIRECTORIO\_B}$
$Datos_{DIRECTORIO\_B}$ - $IDT_{DIRECTORIO\_B}$

De esta forma se facilita la realización de pruebas en las que se infieren relaciones entre instancias de una misma unidad de análisis en diferentes registros. Los datos extraídos de iDatos permiten montar una réplica del sistema pero con un volumen más reducido y con información anonimizada para poder ser explotada por los investigadores externos al Instituto Canario de Estadística.

La distribución de los datos en la muestra generada se pueden consultar en la Tabla 1. Se puede comprobar que en todos los directorios el volumen de datos en las tablas URD, que representan relaciones es considerablemente mayor que el resto de tipos.

IDT Empresas	154
IDF Empresas	982
URD	16.650
IDT Población y Hogares	2.472
IDF Población y Hogares	4.826
URD Población	169.926
IDT Portales	2.126
IDF Portales	3.896
URD Portales	102.624
CCotiza Junio	89.558
CCotiza Septiembre	89.193
Afiliados Seguridad Social Junio	2.216
Afiliados Seguridad Social Septiembre	2.586
Demandantes de empleo Junio	273
Demandantes de empleo Septiembre	153
Padrón 2017	2.465

Tabla 1. **Distribución de datos en la muestra.**

Diseño del Grafo

La representación de iDatos sobre un grafo ha llevado a considerar nodos que representan los registros en las tablas IDT, IDF y Datos crudos en los ficheros de cada registro. Las relaciones entre ellos se obtienen de las tablas URD. Las etiquetas de Neo4j permiten especificar a qué directorio pertenece un nodo IDT o uno IDF y también especificar el fichero del que proviene cada dato crudo. Los atributos de cada registro se

mapean a propiedades del nodo. A las relaciones se les asigna una etiqueta con el tipo de relación indicado en el atributo REL_TYPE de las tablas URD.

Por otra parte, el diseño de registros en algunos ficheros de datos en iDatos no respeta la segunda forma normal, lo que hace que determinadas características representen múltiples valores de algún atributo. En estos casos se ha optado por representar el atributo aplanado mediante un tipo de nodo adicional que se relaciona con la instancia que lo genera con una relación AUXILIAR y cada uno de los valores que toma para ese registro genera una instancia del tipo de nodo correspondiente. Por ejemplo, en el caso del fichero de Demandantes de empleo según SISPE, la formación ocupacional se repite cinco veces, cada uno como un atributo diferente con distintos códigos, que van desde el identificador único del registro de la formación ocupacional, hasta por ejemplo el tipo de formación. Lo mismo sucede con el idioma, con 8 valores posibles, contiene códigos con los idiomas que habla la persona física, y cada uno de ellos con atributos para el nivel de lectura, nivel de escritura, etc.

El grafo resultante tiene los siguientes tipos definidos:

- Nodos: $IDT_{POBLACIÓN}$, $IDT_{PORTALES}$, $IDT_{EMPRESAS}$, $IDF_{POBLACIÓN}$, $IDF_{PORTALES}$, $IDF_{EMPRESAS}$, DAT_PMDH, DAT_AfiliacionesSeguridadSocial, DAT_Muface, DAT_DemandantesEmpleoSISPE, DAT_Sitcan, DAT_Dirce, DAT_CuentasCotizacion
- Relaciones:
    - Directorio Población y Hogares:

      $IDF_{POBLACIÓN} \rightarrow IDT_{POBLACIÓN}$ etiqueta: SOURCE
      DAT_PMDH → $IDT_{POBLACIÓN}$ etiqueta: TRACE_STOCK
      DAT_AfiliacionesSeguridadSocial → $IDT_{POBLACIÓN}$ etiqueta: TRACE_STOCK
      DAT_DemandantesEmpleoSISPE → $IDT_{POBLACIÓN}$ etiqueta: TRACE_STOCK

    - Directorio Calles y Direcciones:

      $IDF_{PORTALES} \rightarrow IDT_{PORTALES}$ etiqueta: GEOREF
      DAT_PMDH → $IDT_{PORTALES}$ etiqueta: TRACE_STOCK
      DAT_Sitcanl → $IDT_{PORTALES}$ etiqueta: TRACE_STOCK

    - Directorio Unidades Económicas:

      $IDF_{EMPRESAS} \rightarrow IDT_{EMPRESAS}$ etiqueta: SOUCE
      DAT_CuentasCotizacion → $IDT_{EMPRESAS}$ etiqueta: TRACE_STOCK

El grafo resultante se puede consultar en la Figura 2, en el que se incluyen los tipos de nodos y los tipos de relaciones. Los casos en que las propiedades del nodo o la relación corresponden a un listado muy amplio se han truncado en la visualización.

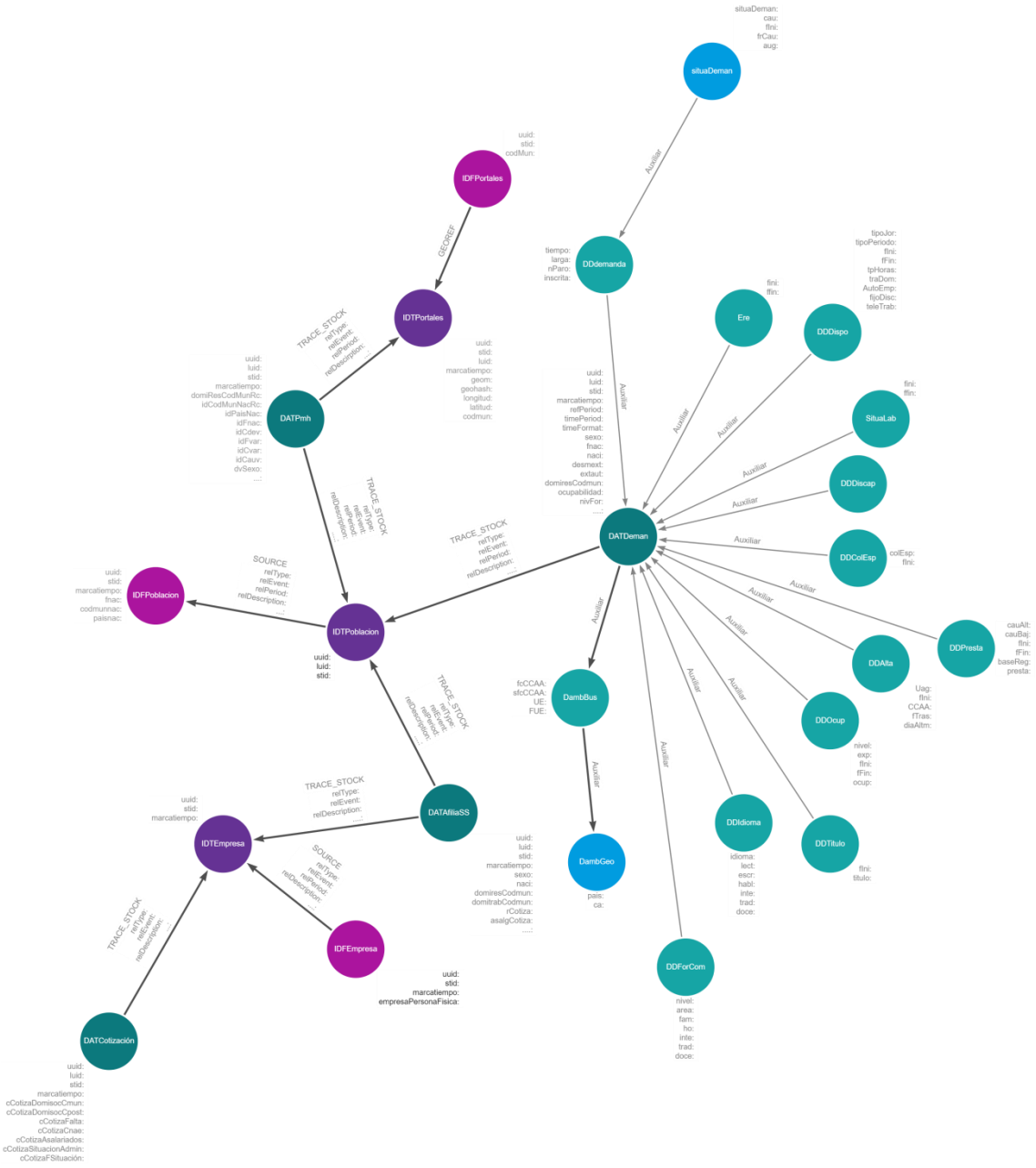

Figura 2. **Grafo resultante para la infraestructura iDatos con datos anonimizados**.

Base de datos relacional.

Además del diseño del grafo se ha implementado una base de datos relacional sobre PostgreSQL con la misma estructura de los directorios de iDatos restringida a los datos en la muestra. Esta se ha utilizado para realizar la comparación de rendimiento de las consultas equivalentes en ambos sistemas.

La base de datos relacional integra las tablas IDT, IDF de los tres directorios: Población y Hogares, Calles y Direcciones y Unidades Económicas. Respecto a las fuentes de datos, se ha generado una tabla por cada fuente, en la que se registran los datos en los ficheros de Afiliados a la Seguridad Social y Demandantes de Empleo según Sispe en dos fechas: junio y septiembre de 2017. También se ha incluido el Padrón Municipal de

Habitantes de julio de 2017 y las Cuentas de Cotización a la Seguridad Social en las mismas fechas, junio y septiembre de 2017.

Las relaciones de tipo TRACE_STOCK entre las fuentes y los datos maestros, y las SOURCE, entre los IDT e IDF, se almacenan en la tabla URD de cada directorio. El modelo relacional se puede consultar en la Figura 3.

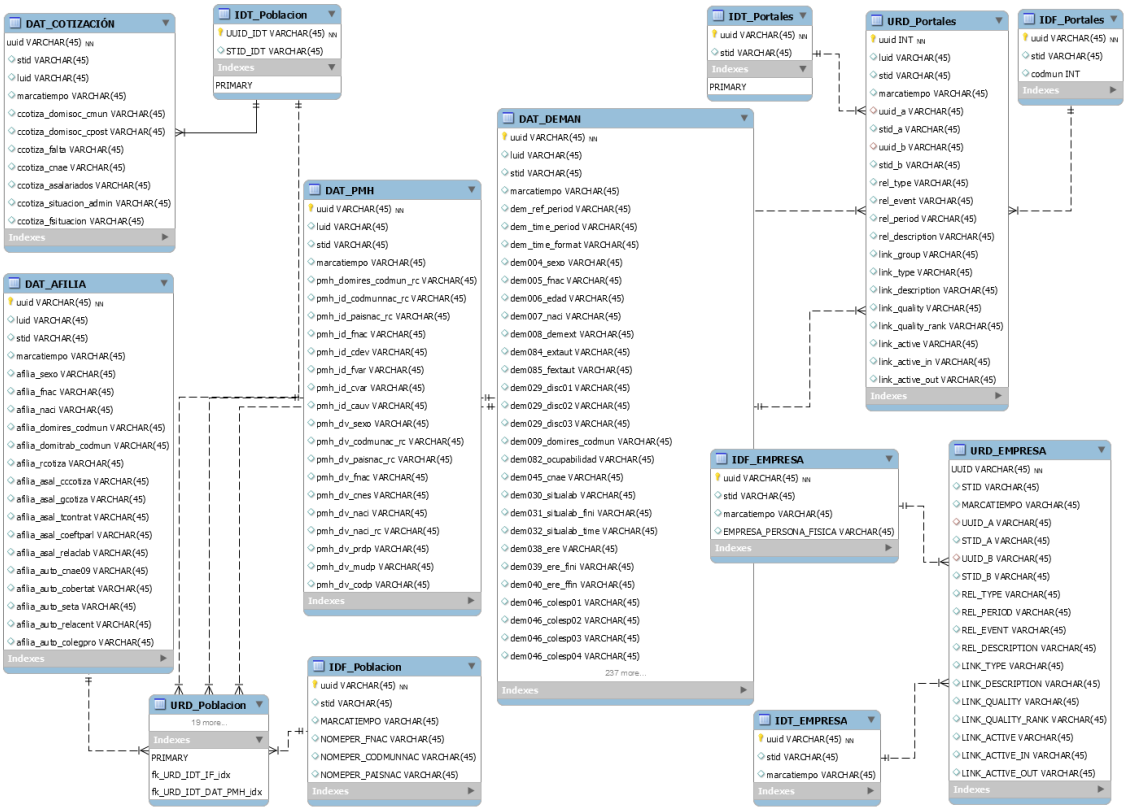

Figura 3. **Diseño de la base de datos que emula la infraestructura iDatos.**

Resultados

El objetivo principal de este trabajo ha sido el diseño del grafo, y realizar un estudio comparativo del rendimiento de consultas sobre la base de datos relacional PostgreSQL frente a la base de datos Neo4j. Las consultas elegidas para las pruebas se pueden dividir en dos grupos principales:

- Consultas complejas que permiten explotar información en la base de datos:
  - Empresas que han aumentado el número de empleados y en qué cantidad
  - Empresas que han disminuido el número de empleados y en qué cantidad
- Consultas que resuelven problemas habituales en la obtención de información en la Estadística de Población Activa Registrada (EPA-Reg)
  - Personas que se encuentran en afiliados y demandantes en el mismo mes y año.
  - Personas en archivos de afiliados con valores diferentes de sexo y fecha de nacimiento

- Personas con valor sexo en alguna de las tablas IDF, Afiliados, Demandante y Padrón diferente
- Listado de personas que están en alguno de los ficheros de afiliados o demandantes y no están en PMDH

Para cada consulta se han realizado dos ejecuciones con cada uno de los sistemas gestores, siempre sobre una máquina virtual del servicio iaaS de la Universidad de Laguna con sistema operativo Ubuntu, 16 Gigas de RAM y 20 Gigas de disco. En la misma máquina se ha instalado PostgreSQL v. y Neo4j v. Las 4 ejecuciones de cada consulta se han realizado en el mismo día y la misma sesión de usuario para garantizar idénticas condiciones en el servidor. La Tabla 2 resume los datos que caracterizan las consultas y sus tiempos de ejecución. Cada fila recoge las tablas que están implicadas en la consulta, los tiempos empleados en las 2 ejecuciones, el sistema sobre el que se ha ejecutado y la cantidad de nodos devueltos. Los casos en que se obtiene un número diferente de casos en el sistema gestor se deben a instancias que no tienen registro en alguno de los ficheros mensuales y en el otro sí. En PostgreSQL no se está considerando y en Neo4j sí aparece. Destaca la consulta de personas que tienen más de un valor en las variables sexo o fecha de nacimiento en la tabla IDF del directorio de Población, por ser el único caso que en una de las ejecuciones se obtienen peores resultados en Neo4j. El resto de casos y ejecuciones se obtienen tiempos menores en Neo4j (Figura 4). Se ha calculado el tiempo promedio las dos ejecuciones de las consultas en cada uno de los sistemas y se ha obtenido el factor multiplicativo que indica en qué cantidad excede el tiempo en PostgreSQL a las consultas en Neo4j, siendo el valor mínimo 1,755 y el máximo 100,6976744.

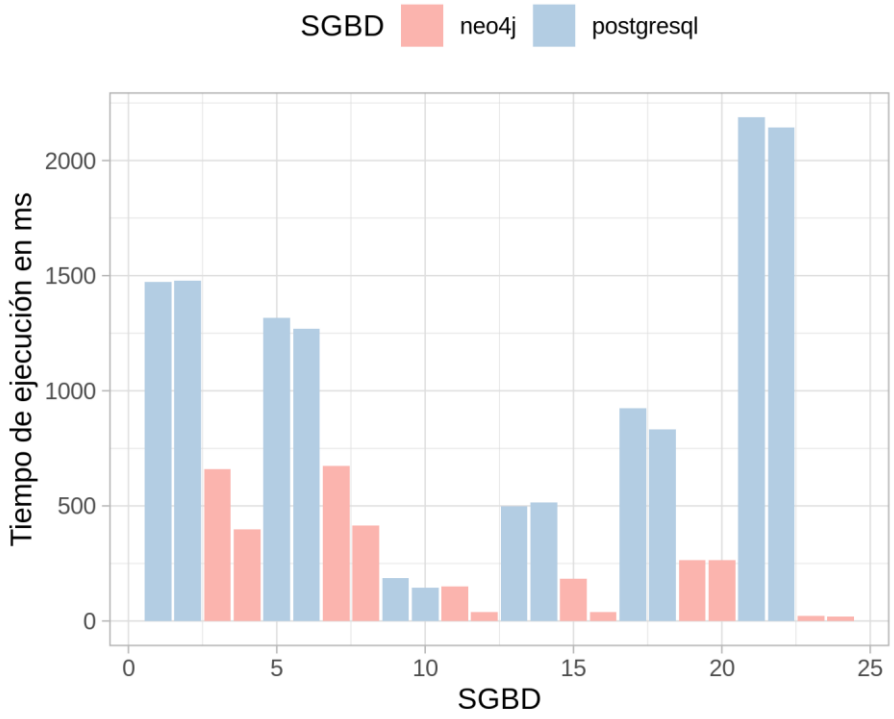

Figura 4. Tiempos de ejecución en Neo4j y PostgresQL en ms.

Consulta	Tablas	tiempo	Ejec	SGBD	Nodos
Empresas que han aumentado el número de empleados y en qué cantidad	URD_EMPRESAS; IDT_EMPRESS; CUENTAS_COTIZACIÓN	1474	Ejec1	postgresql	37
		1479	Ejec2	postgresql	37
		659	Ejec1	neo4j	38
		398	Ejec2	neo4j	38
Empresas que han disminuido el número de empleados y en qué cantidad	URD_EMPRESAS; IDT_EMPRESS; CUENTAS_COTIZACIÓN	1318	Ejec1	postgresql	25
		1270	Ejec2	postgresql	25
		674	Ejec1	neo4j	27
		414	Ejec2	neo4j	27
Listado de personas que tienen más de un valor en las variables sexo o fecha nacimiento.	IDT_POBLACIÓN; IDF_POBLACIÓN; URD_POBLACIÓN	185	Ejec1	postgresql	173
		145	Ejec2	postgresql	173
		149	Ejec1	neo4j	173
		39	Ejec2	neo4j	173
Personas en archivos de afiliados con valores diferentes de sexo y fecha de nacimiento	DT_POBLACIÓN; URD_POBLACIÓN; DAT_AFILIADOS	499	Ejec1	postgresql	3
		516	Ejec2	postgresql	3
		184	Ejec1	neo4j	3
		38	Ejec2	neo4j	3
Personas con valor sexo en alguna de las tablas IDF, Afiliados, Demandante y Padrón diferente	IDT_POBLACIÓN; IDF_POBLACIÓN; DAT_AFILIADOS; DAT_DEMANDANTES; DAT_PMDH	845	Ejec1	postgresql	10
		683	Ejec2	postgresql	10
		263	Ejec1	neo4j	10
		263	Ejec2	neo4j	10
Listado de personas que se encuentran en afiliados o demandantes, pero que no se encuentren en el padrón.	IDT_POBLACIÓN; IDF_POBLACIÓN; URD_POBLACIÓN; DAT_AFILIADOS; DAT_DEMANDANTES; DAT_PMDH	2187	Ejec1	postgresql	9
		2187	Ejec2	postgresql	9
		23	Ejec1	neo4j	9
		23	Ejec2	neo4j	9

Tabla 2. **Resumen de resultados de las ejecuciones**

Consulta	Nodos	Exceso
Empresas que han aumentado el número de empleados y en qué cantidad	37	2,793755913
Empresas que han disminuido el número de empleados y en qué cantidad	25	2,378676471
Listado de personas (idt), número de registros en idf, número de valores diferentes de sexo y fecha de nacimiento. Solamente aquellos que tienen más de un valor en las variables sexo o fecha nacimiento.	173	1,755319149
Personas en archivos de afiliados con valores diferentes de sexo y fecha de nacimiento	3	4,572072072
Personas con valor sexo en alguna de las tablas IDF, Afiliados, Demandante y Padrón diferente	10	3,336501901
Listado de personas que están en alguno de los ficheros de afiliados, demandantes, Cuentas de Cotización y no están en PMH	9	100,6976744

Tabla 3. Factor que multiplica el tiempo de ejecución en PostgreSQL

Conclusión

El presente trabajo muestra la capacidad de Neo4j como SGBD para dar soporte a la gestión y generación de datos maestros en la infraestructura iDatos, como alternativa a PostgreSQL. Se ha presentado un diseño fácilmente adaptable a la versión completa de los directorios que conforman actualmente esta infraestructura. El volumen de datos manejado ha sido limitado para preservar el secreto estadístico por una parte y por tratarse de una experiencia piloto por otra. Se ha experimentado con consultas que resuelven problemas presentes en la elaboración de EPA-Reg. En todas las pruebas, en promedio se disminuye en al menos 1,75 veces el tiempo de ejecución. Sin embargo, y dadas las características de ambos sistemas, se espera que estas diferencias sean incluso más notables cuando se trabaje en el caso real, con volúmenes de datos muchos mayores. Finalmente, cabe destacar además de las mejoras en el rendimiento, –la expresividad del modelo, que facilita la comprensión del sistema y la sintaxis de las consultas.

Principales referencias bibliográficas

[1] DB-engines ranking https://db-engines.com/en/ranking consultado el 8 de octubre de 2021

[2] Mathur, N. (2021) Rethink Your Master Data How Connections Will Define the Future of MDM.
[3] Robinson, I., Webber, J., & Eifrem, E. (2015). Graph databases: new opportunities for connected data O'Reilly Media, Inc..
[4] Rodríguez Lucas, D. (2014). Master Data Management as a tool to improve business profitability.
[5] Yanes, J. A. G., Morales, N. M., González, A. R., Díaz, D. J. L., Villalba, R. B., & Medina, E. T. Marco de desarrollo del Sistema de Datos Integrados de Canarias (iDatos).