# Clasificación con ayuda de bases grafo[1]

Teoría de la clasificación, aplicaciones prácticas

**Tanguy Wettengel, Marcos Monteleone**

Curso para estudiantes de tercer año de la Licenciatura en Ciencias de la Computación
de la Facultad de Ingeniería de la Universidad Nacional de Cuyo, 2024
Duración: 3 semanas / Modalidad: Virtual

***Palabras-clave**: Clasificación, Bases de datos grafo, Modelos de datos, Didáctica*

[1] Este texto no constituye un artículo científico sino un informe destinado a documentar el trabajo llevado a cabo en un curso con este intitulado en los últimos 3 años (2022-2024). Tiene como objetivo dejar constancia de un método didáctico basado tanto en conocimientos previos de los estudiantes (en el área de Informática) como en intuiciones guiadas, en el ámbito de la clasificación. Esto último significa que las definiciones de los conceptos indispensables no han constituido puntos de partida sino siempre conclusiones de su propio análisis de ejemplos, a veces aparentemente pueriles, pero sistemáticamente orientados a la construcción paulatina de abstracciones complejas. En la medida de lo posible, la terminología empleada fue concebida a partir de nomenclaturas de las que los estudiantes ya tenían noticia y consolidada progresivamente, de manera a no desvalorizar sus marcos de referencia, sino a hacerlos evolucionar. La ausencia de bibliografía en este informe refleja la dinámica del curso, en el que no se utilizó ninguna, salvo las consultas incesantes que los participantes debieron hacer por Internet cuando el lenguaje de programación de bases de datos-grafo utilizado (Cypher), que les era completamente desconocido antes del curso, los ponía en dificultades. Por último, cabe aclarar que tanto la formulación de la arquitectura de los modelos de datos propuesta como su transposición en la estructura de la base son absolutamente originales, en el sentido en que no poseen antecedentes bibliográficos conocidos. Toda consulta y comentario sobre el presente informe puede enviarse a tanguyjardin@icloud.com

# Índice

# 1. Introducción

1. En el contexto de un curso de Comunicación Técnica como el que ha servido de marco a los trabajos citados en este informe, incluir la teoría y práctica de la clasificación en el programa podría relacionarse, a primera vista, con el diseño de catálogos, puesto que se trata de dispositivos corrientes que agrupan por categorías (es decir, *clasifican*) un conjunto de bienes o servicios.

2. Sin embargo, múltiples aplicaciones independientes de la producción de catálogos implican la clasificación como una operación previa al logro de objetivos diversos, tales como obtener estadísticas, recomendar productos a consumidores, detectar fraudes económicos, identificar autores probables de delitos, establecer perfiles de clientes, predecir posibles catástrofes naturales, estimar el valor de un servicio, describir el funcionamiento de una empresa, sin mencionar siquiera el amplísimo ámbito de la búsqueda de información en fuentes como fondos de bibliotecas, recopilaciones de leyes, observaciones antropológicas, "fake-news", historias clínicas, noticias sobre celebridades, anuncios de espectáculos, etc.

3. En la medida en que la enseñanza de la clasificación tenga como público a estudiantes de Informática, no conectar la teoría con la herramienta natural para producir clasificaciones (es decir, una base de datos), sería, en el mejor de los casos, una extravagancia. Inversamente, un curso de bases de datos desconectado de la teoría de la clasificación habilitaría a los estudiantes a usar una herramienta sin definir explícitamente par qué sirve. Conviene, en este sentido, recordar que una base de datos sirve primariamente, o bien para extraer selectivamente o bien para presentar en su totalidad datos almacenados, agrupados y ordenados del modo que se desee. En ambas situaciones, la clasificación cumple un papel primordial.

4. Por estos motivos, el módulo "Clasificación" del curso de Comunicación Técnica se ha combinado con un módulo del curso de Teoría de Bases de Datos en la Licenciatura en Ciencias de la Computación de la Facultad de Ingeniería de la Universidad Nacional de Cuyo en 2023 y 2024. Aprovechando la oportunidad para ampliar el espectro de los tipos de bases de datos utilizados por los estudiantes de esta carrera, en lugar de usar como herramienta bases relacionales, el módulo "Clasificación" integró una formación acelerada a bases grafo y al lenguaje Cypher, a la implementación de las bases diseñadas y producidas por los estudiantes en la plataforma Neo4j, y, sobre todo, a técnicas y estrategias de modelado de estructuras de datos destinadas específicamente a este tipo de bases.

5. La duración total del curso fue de tres semanas, con 12 horas de clases virtuales, sumadas a un número variable de horas-proyecto (mínimo, 12), en las que el instructor trabajó individualmente con cada uno de los binomios participantes. Al cabo de ese período, los estudiantes presentaron en público sus trabajos respectivos, mostrando en funcionamiento las bases de datos que habían diseñado e implementado, y ejecutando a lo largo de sus exposiciones las consultas apropiadas a los objetivos y usuarios previstos.

6. La segunda sección de este artículo describe los contenidos aportados durante el curso en el ámbito de la teoría de la clasificación que concierne los principales modelos de estructura de datos. La tercera describe la manera de implementar estructuras de tipo red en bases de datos grafo. En la cuarta se trata la generación de clasificaciones mediante consultas de la base de datos, mientras que la quinta presenta trabajos de estudiantes de la promoción 2024: sólo tres proyectos aparecen citados, dada nuestra intención de ilustrar en este informe objetivos de clasificación completamente diferentes. Los ejemplos incluyen código Cypher con las aclaraciones necesarias.

7. Antes de entrar en materia concretamente, conviene recalcar que los datos dentro de una base grafo **no están clasificados**: por una parte, el concepto de "clase" no existe (en el sentido estricto) en este tipo de bases; por otra, los datos sólo tienen características asignadas, sin agrupamiento ninguno en función de las que les son comunes (este punto fundamental quedará claramente demostrado en la cuarta sección de este informe). Cualquier clasificación aparece únicamente en la respuesta que la base devuelve ante una consulta específica, lo que equivale a decir que las consultas *producen clasificaciones* que no existen al margen de ellas.

## 2. Modelos de estructura de datos

8. Para introducir el tema a partir de nociones ya conocidas por los estudiantes, en particular las de *individuo* y *clase*, la distinción entre "¿Quién es?" y "¿Qué es?" aplicada a la imagen de un loro apodado "Rafa", sirvió de punto de partida, sin que se hiciese referencia a la distinción entre *clase* y *categoría*. Por generalizaciones sucesivas, los estudiantes pasaron intuitivamente de la clase "loro" a las clases "ave" y "animal", sentando así las bases para la definición de una taxonomía.

9. Este concepto fue consolidado mediante la referencia a las categorías de Aristóteles y a sus reinterpretaciones ulteriores (el "árbol" de Porfirio [234-305] y la clasificación de Lineo

[1707-1778]). La ventaja de aludir a estos antecedentes históricos reside no sólo en que ponen de manifiesto la diferencia entre "sustancia" y "accidentes", similar a la de "objetos" y propiedades", sino también en que implican las nociones de "generalización" y de "herencia", fundamentales en el modelado de estructuras de datos. La comprensión de las relaciones taxonómicas permitió a continuación diferenciar *taxonomías* de *tipologías*, así como clarificar otros términos utilizados generalmente de modo poco estricto (categoría, clase, instancia, jerarquía, nomenclatura, etc.).

10. Los conceptos citados sirvieron como encuadre a la presentación de las tres estructuras de datos más frecuentes en el universo digital: de la más simple a la más compleja, palabras-clave, jerarquías y ontologías. Las primeras fueron descritas como simples listas, en las que los términos no tienen ni relaciones necesarias entre sí ni están sujetos a una nomenclatura. Las jerarquías, como conjuntos de relaciones taxonómicas que autorizan inferencias transitivas ("si A es un tipo de B y B es un tipo de C, A es un tipo de C"). Aquí aparecieron las diferencias entre "es un" y "es un tipo de", las que, aplicadas al caso de "Rafa" y "loro", permiten distinguir entre "Rafa *es* un loro", y "Un loro *es un tipo de* ave" (respectivamente, relación de instancia individual a clase y relación de clase-instancia a clase). Con lo cual se aclaró que "instancia" no constituye un descriptor suficiente de una entidad en una jerarquía, y que, por lo tanto, el término debe ser asociado, o bien a "individual" o bien a "clase" ("instancia individual" / "clase-instancia").

11. Con este precedente se introdujo la tercera estructura de datos, en la que se combinan relaciones taxonómicas con otras que no lo son, y en las que las propiedades que se heredan entre niveles de una jerarquía pueden tener la forma de simples atributos o de relaciones de una clase con otra. Si "automóviles" es una subclase de "vehículos", por ejemplo, y la clase "vehículos" tiene un atributo "marca", todas las instancias de "automóviles" heredarán ese atributo. "Atributo" significa aquí lo que literalmente expresa, es decir, sólo el antecedente dentro de una estructura "x:y". Si "x" representa "marca", "y" será, según el caso, "Ford", "Renault", "Tesla", etc., es decir, el "valor" asociado al atributo "marca". Si la clase "vehículo" tiene, por otra parte, una relación de tipo "propietario" con una clase que representa a personas, de modo que se pueda afirmar que una instancia de "vehículo" tiene un propietario que es una instancia de la clase "persona", todas las instancias de "automóvil" (subclase de "vehículo") heredan esa relación.

12. Este modo híbrido de atribuir características a instancias individuales dentro de una estructura de datos define las peculiaridades de una arquitectura de tipo "red". Contrariamente a las taxo-

nomías, una red no tiene raíz. Redes que describen un dominio cualquiera desde el punto de vista de usuarios con intereses y perfiles específicos se conocen bajo el nombre de "Ontologías", y tienen la particularidad de poder caracterizar un objeto, o más comúnmente, un tipo de objeto, desde diferentes puntos de vista. Así, por ejemplo, el automóvil citado en el §10 tiene una marca y posee igualmente un propietario, además de poder tener un fabricante, que a su vez produce en tal o cual país, una potencia de motor, un tipo de energía de propulsión, la cual tiene un precio por litro en una fecha dada, una proveniencia, etc., etc.

13. Lo más interesante de estas estructuras, es que los puntos de entrada no están definidos (dado que no hay raíz). De modo que se puede utilizar la misma ontología para descubrir lo que se refiere a vehículos que para descubrir lo que se refiera a combustibles, dentro de los límites de la información que se haya integrado, como, por ejemplo, cuál es el combustible utilizado por la mayor cantidad de marcas de vehículos, cuál es su proveniencia, dónde residen los dueños de vehículos que usan ese tipo de energía, etc. Esta plasticidad, además de su arquitectura grafo nativa (nodos relacionados), convierte las ontologías en soluciones extremadamente convenientes para el diseño de modelos implementables en el tipo de bases utilizado durante el curso.

## 3. Modelos de estructura de datos diseñados para bases grafo

14. Diseñadas gráficamente teniendo en cuenta las convenciones que aparecen en el Apéndice, las ontologías fueron presentadas como el candidato ideal para diseñar la estructura de una base grafo. Sobre todo porque la transposición del modelo de datos a la estructura de la base es prácticamente literal si se respetan convenciones, además de ser fácilmente inteligible para clientes sin experiencia ni en informática ni en representaciones de estructuras de datos, cualesquiera sean, con mínimas explicaciones.

15. Aunque las bases grafo no contienen sino instancias individuales, la arquitectura de modelos de datos propuesta aquí representa únicamente clases y las divide en dos tipos: las "concretas", que tienen instancias individuales, y las "abstractas" que tienen sólo clases como instancias. La función de las clases abstractas es únicamente transmitir características a las instancias individuales de clases concretas. Todas las clases tienen propiedades (que transmiten a sus propias instancias o, en el caso de ser abstractas, a las de una clase concreta subordinada). Cuando una clase tiene una relación con otra en el diagrama, esa relación conecta las instancias (directas o indirectas) de ambas. Por último, las relaciones entre clases pueden tener propiedades igualmente: en este caso, esas propiedades se replican en las relaciones entre instancias individuales.

16. La Figura 1 representa gráficamente una ontología diseñada para una compañía de seguros. Las clases "concretas" tienen contorno punteado y aparecen en amarillo cuando los valores de sus atributos no están predefinidos (=) y en verde cuando se los elige de una lista predeterminada (= /). Las "abstractas" tienen contorno pleno y sus etiquetas aparecen en minúscula entre paréntesis. Las relaciones son de dos tipos: o taxonómicas o etiquetadas. Estas últimas pueden tener propiedades a su vez. En el caso documentado, los valores posibles de sus atributos están predefinidos (= /). Las relaciones taxonómicas tienen por función conectar clases que transmiten sus características (atributos, relaciones) a clases subordinadas.

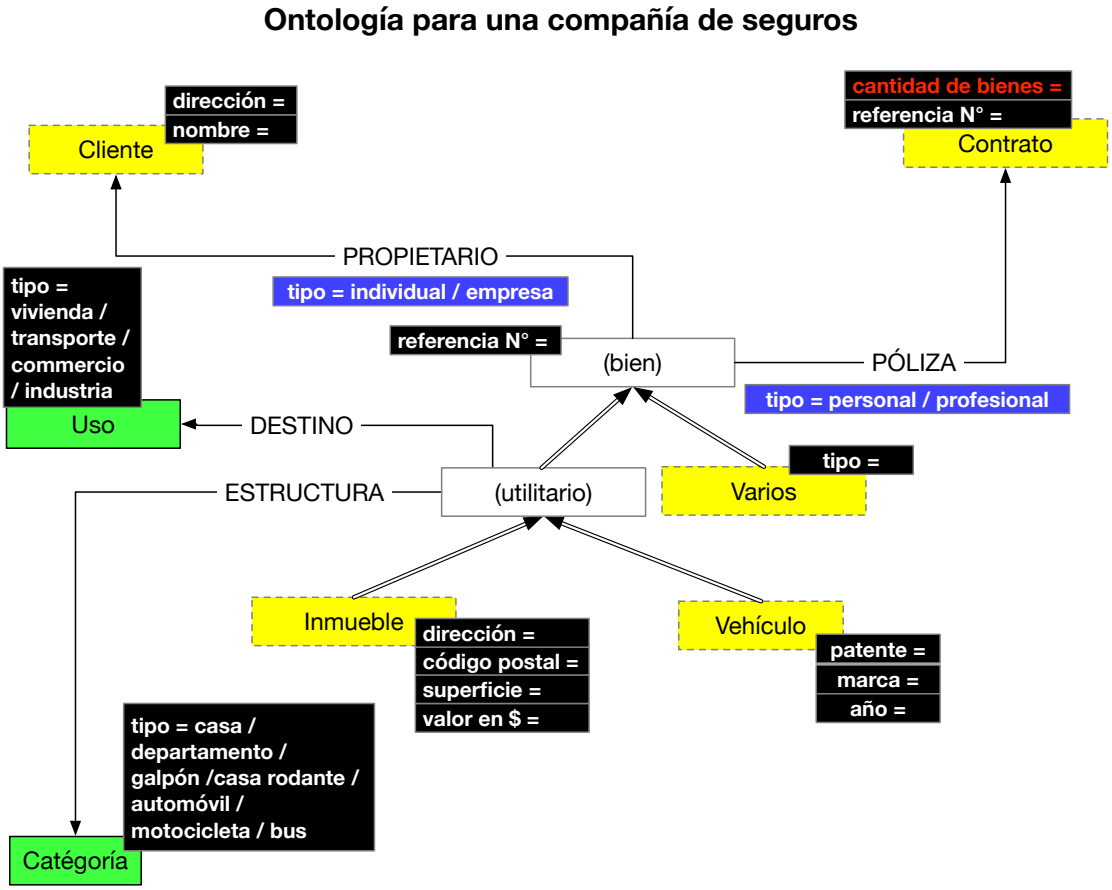



Figura 1: Un modelo de datos diseñado como una ontología

17. En la práctica, el modelo funciona del modo siguiente: una instancia de **Inmueble**, por ejemplo, con *referencia N°* "9856" tiene como *dirección* "Perú 215", como *código postal*, "5500", como *superficie*, "120m²", como *valor en $*, "164000", como ESTRUCTURA, "casa", como DESTINO, "vivienda", como PROPIETARIO de *tipo* "individual" al **Cliente** de *nombre* "Juan González" con *dirección* "Uruguay 352" y como PÓLIZA de *tipo* "personal" el **Contrato** con *referencia N°* "2387523". La propiedad *cantidad de bienes* (en rojo) asociada a la clase **Contrato** tiene

un valor calculado, que es la suma de los bienes asegurados que figuran en la *referencia N°* "2387523", por ejemplo, "2", en el caso de que en la misma póliza figure el inmueble citado y un vehículo.

18. El ejemplo pone de manifiesto que la referencia del inmueble (*referencia N°* "9856"), por ejemplo, es una característica heredada de la clase abstracta "(bien)", a través de otra clase abstracta "(utilitario)", que es una clase-instancia de la anterior, y que transfiere sus propiedades (las propias y las heredadas) a las instancias de la clase concreta "Inmueble", a su vez clase-instancia de "(utilitario)". La ventaja de utilizar de este modo clases abstractas es que las generalizaciones que permiten (asociar, por ejemplo, una referencia a toda instancia indirecta, que se trate de inmuebles, de vehículos o de otros bienes - aquí llamados "Varios" -, como collares, piedras preciosas, animales, etc.) evita tener que repetir tanto atributos como relaciones entre clases. El modelado gana así en términos de economía pero también en precisión, puesto que especifica el grado de abstracción en el que se sitúan las características de las instancias.

19. Reemplazando el término "clase" por "grupo", más acorde con la arquitectura de las bases grafo, la Figura 2 aclara la proveniencia de las características de una instancia ("nodo"):

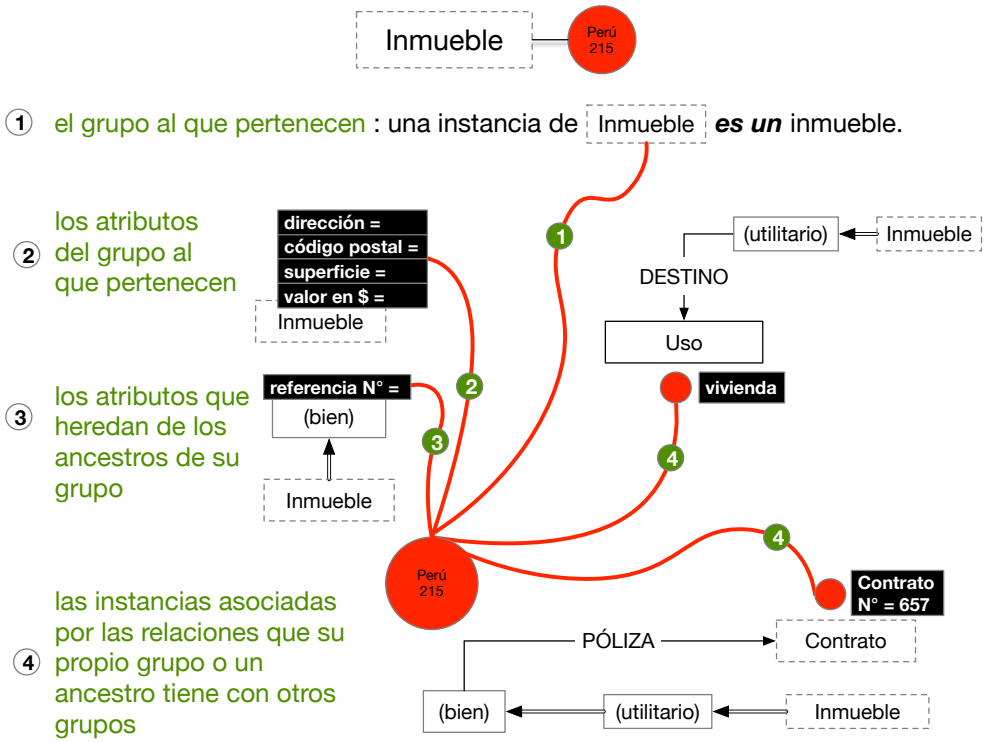



Figura 2: Origen de las características de las instancias

20. Siguiendo la lógica de este principio de modelado, la Figura 3 describe una instancia con referencia al modelo de datos de la Figura 1:

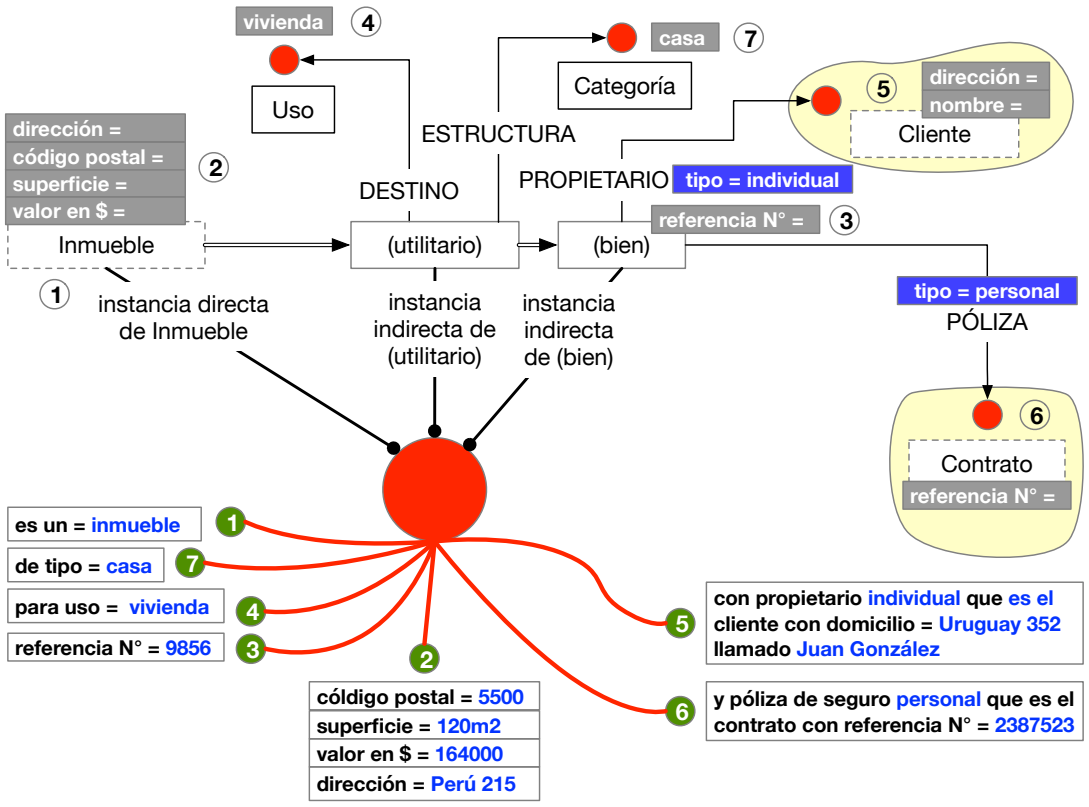



Figura 3: Un ejemplo de instancia

21. La Figura 4 representa la traducción de esta misma instancia y de sus características en una base datos grafo. Como se puede observar, lo que aparece en el modelo de la Figura 1 como "clases" está aquí representado como *etiquetas* (‘labels") de instancias (por ejemplo, ": Contrato" referida

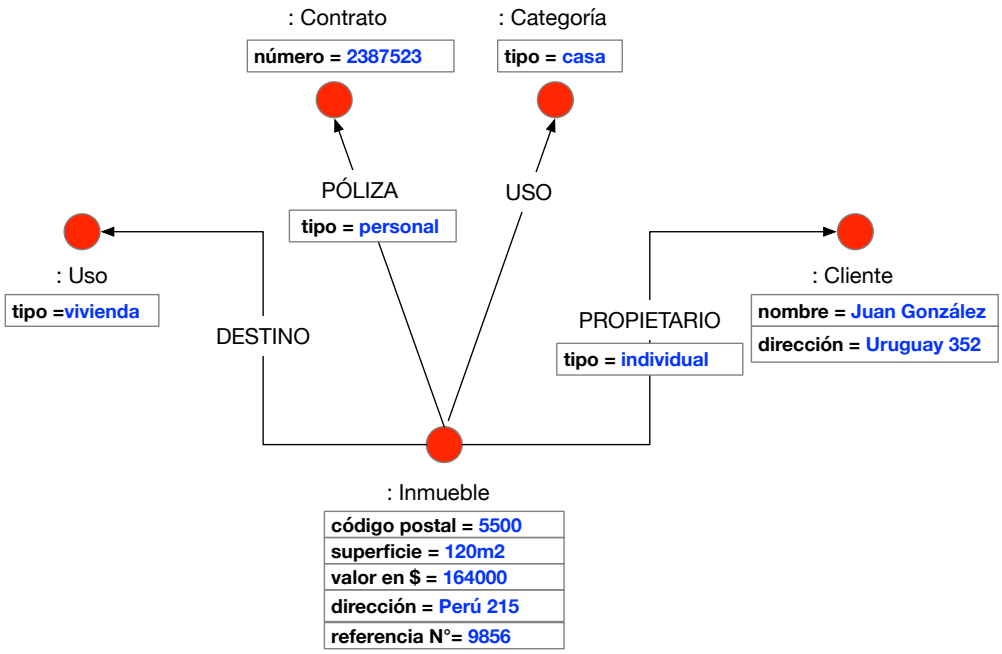



Figura 4: Representación de instancias dentro de la base grafo (nodos con propiedades y relaciones)

a la clase **Contrato** o ":Inmueble", a la clase **Inmueble** del modelo). Estas etiquetas permiten agrupar instancias cuando se ejecutan consultas: es posible, en una consulta, por ejemplo, extraer todas los nodos que lleven la etiqueta ": Inmueble" para generar un grupo que no existe como tal dentro de la base.

22. A diferencia de las clases, las etiquetas no conllevan en la base restricciones obligatorias de características, dado que dos nodos con la misma etiqueta pueden exhibir propiedades de naturaleza diferente. Sin embargo, la estrategia elegida para el curso prevé que los nodos con la misma etiqueta compartan todas sus propiedades, no sólo para facilitar la transposición del modelo a la base, sino también para aprovechar los conocimientos de los estudiantes en materia de modelos orientados objeto. La excepción es el caso en que atributos heredados no tengan un valor definido, en cuyo caso el resultado de una consulta especificará, en la línea de la tabla que corresponda al nodo afectado, la inexistencia de valor para el atributo en cuestión. La sintaxis que especifica la etiqueta asociada a un nodo en Cypher es una dupla constituida por dos puntos y una designación, colocada inmediatamente a continuación del nombre efímero de la variable que identifica un nodo en una consulta ("x:Inmueble", por ejemplo).

23. Por último, la Figura 4 pone en evidencia que las clases abstractas del modelo (Figura 1) no aparecen representadas en la base, sino que únicamente los atributos y relaciones que transmiten lo están. En esta particularidad reside el principal contraste entre el modelo y la estructura de la base de datos, que no son isomorfos a pesar de ser equivalentes: mientras que el modelo de la estructura de los datos es una *conceptualización*, la estructura de la base, es su *implementación*.

## 4. Clasificaciones

24. *Clasificar* es agrupar entidades por características compartidas.[2]. Desde este punto de vista, cualquier extracción de una base de datos equivale de por sí a una clasificación, puesto que genera sistemáticamente dos clases: una, que comprende los datos extraídos; otra, el resto de los datos contenidos de la base. Las "características compartidas" de cada conjunto, son simplemente el hecho de que cumplan o no cumplan con las restricciones definidas por la consulta. Concretamente, lo que una consulta extrae de la base constituye una clase, lo que no extrae, otra.

[2] Los agrupamientos pueden definirse, tanto como *clases* o como *conjuntos* o como *categorías*, según la perspectiva del análisis (dada la orientación práctica del curso, no se hicieron distinciones entre estos tres conceptos).

25. Si el resultado devuelto es, por ejemplo, la dirección de una persona en particular en un inventario de clientes, la consulta habrá generado dos conjuntos: uno contiene la dirección de la persona buscada, el otro, todo lo demás. Si se interroga la base para visualizar los nombres y direcciones de todos los clientes (asumiendo que la base sólo contenga este tipo de datos), el resultado devuelto será una lista donde figura el contenido completo de la base. En este caso, la consulta habrá generado igualmente dos conjuntos: uno, que comprende las informaciones de la totalidad de los clientes, el otro, vacío. En el caso complementario (ningún cliente satisface las restricciones de la consulta), el conjunto vacío será el producido por la devolución a la consulta.

26. Extraer una clase mediante una consulta es la operación que llamamos aquí *Selección*. Teniendo en cuenta el ejemplo previo (una base de clientes en la que figuran nombres asociados a direcciones), según la consulta, el resultado de una selección será invariablemente, en términos de clientes, "todos", "alguno(s)", o "ninguno". La selección que la consulta devuelve, en caso de contener más de un cliente (es decir, *algunos* o *todos*), puede estar ordenada, si existen en la consulta instrucciones para hacerlo, por ejemplo, según la posición de sus apellidos en una secuencia alfabética.[3]

27. Suponiendo que la dirección asociada con cada cliente especifique la ciudad de residencia por separado (es decir, calle y número, por un lado, ciudad de residencia, por otro), un ordenamiento alfabético *por ciudades* agruparía a todos los clientes previamente seleccionados que residan en la misma ciudad, lo que constituye una clasificación, en la que habrá tantos conjuntos de clientes como ciudades de residencia aparezcan en los datos que la devolución incluye.

28. Los ordenamientos producen clasificaciones efímeras, dado que agrupan por contigüidad inmediata los miembros de una clase generada por la consulta, aunque también sirven para presentar (alfabéticamente, por ejemplo) listas que no constituyen clases. La Figura 5 muestra un fragmento de una extracción de una base que clasifica productores de partes de bicicletas por sus países de origen[4]. En la primera columna aparecen estos países agrupados por orden alfabético (Reino Unido, Suiza, Taiwan), que se repiten tantas veces como productores que de allí provienen existan. Estos productores, que aparecen en la segunda columna, se presentan igualmente

---

[3] Si la consulta no especifica criterios de ordenamiento, por defecto, en Neo4j, la selección aparece según la secuencia de los identificadores internos de la base.

[4] Los ejemplos referidos a bicicletas fueron tomados del proyecto a cargo de Enzo Palau y Tomás Rando

por orden alfabético, sin que este ordenamiento tenga pertinencia alguna en términos de clasificación: si no lo estuviesen, quedarían igualmente agrupados por su país de origen (ver Figura 6).

| Pais | Productor |
|---|---|
| "Reino Unido" | "Raleigh" |
| "Reino Unido" | "Renthal" |
| "Suiza" | "DT Swiss" |
| "Suiza" | "Scott" |
| "Taiwan" | "Alexrims" |
| "Taiwan" | "Funn" |

Figura 5: Productores por país (1)

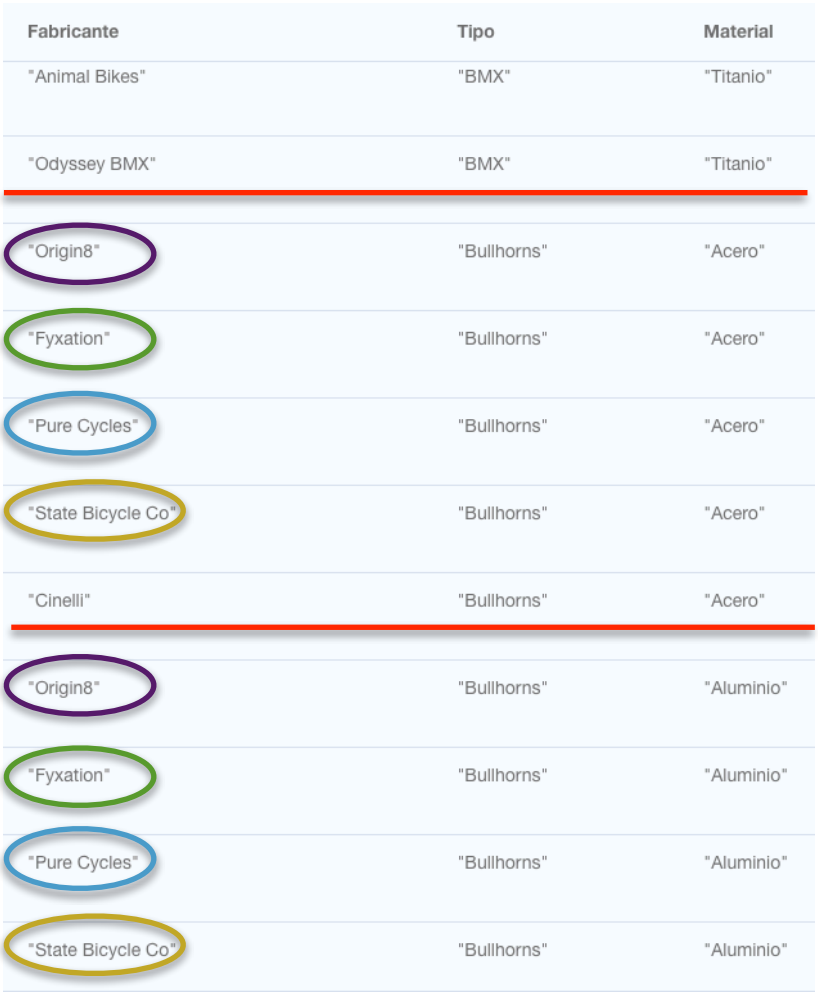
| Productor | Pais |
|---|---|
| "Scott" | "Suiza" |
| "DT Swiss" | "Suiza" |
| "Tektro" | "Taiwan" |
| "Promax" | "Taiwan" |
| "KMC" | "Taiwan" |
| "Giant" | "Taiwan" |

Figura 6: Productores por país (2)

29. La Figura 6 invierte el orden de las columnas (productores en la primera, países en la segunda), sin que esta modificación afecte la significación del resultado. El *criterio de clasificación* es en ambos casos el país de origen y lo que este criterio clasifica es a productores. El número de repeticiones del mismo país indica la cantidad de miembros de la clase "Productores": en el fragmento reproducido por la Figura 5, hay 3 clases (los del Reino Unido, dos productores, los de Suiza, 2 productores, y los Taiwan, también 2). En el fragmento reproducido por la Figura 6, hay 2 clases (los productores de Suiza, que son 2, los de Taiwan, 4). Poco importa, por lo tanto, la secuencia de las columnas: lo que cuenta es que *aquéllas que constituyen criterios de clasificación* estén ordenadas.

30. En las situaciones documentadas por las Figuras 5 y 6, existe un solo criterio de clasificación. La Figura 7 ilustra un caso en que hay dos (puede haber más de dos, naturalmente), pero en todos estos casos es necesario definir un orden de prioridades. Así ocurre, por ejemplo, en la clasificación de fabricantes de manubrios de bicicletas siguiente, los cuales han sido clasificados primariamente (en el extracto de devolución de la base que documenta la Figura 7), a partir de los *tipos* de manubrios que producen, y secundariamente a partir del *material* utilizado en los

manubrios de cada tipo. Se constituyen aquí tres clases de fabricantes en función del tipo de manubrios que producen combinado con el material utilizado, de modo que (según el fragmento extraído), una está compuesta por los que producen manubrios de tipo "BMX" en titanio, otra por los que los producen de tipo "Bullhorns" en acero, y otra por los que producen "Bullhorns" en aluminio. Según se puede observar, el mismo fabricante puede formar parte de dos clases distintas (es el caso de casi todos los que producen "Bullhorns" en acero pero también en aluminio.



| Fabricante | Tipo | Material |
|---|---|---|
| "Animal Bikes" | "BMX" | "Titanio" |
| "Odyssey BMX" | "BMX" | "Titanio" |
| "Origin8" | "Bullhorns" | "Acero" |
| "Fyxation" | "Bullhorns" | "Acero" |
| "Pure Cycles" | "Bullhorns" | "Acero" |
| "State Bicycle Co" | "Bullhorns" | "Acero" |
| "Cinelli" | "Bullhorns" | "Acero" |
| "Origin8" | "Bullhorns" | "Aluminio" |
| "Fyxation" | "Bullhorns" | "Aluminio" |
| "Pure Cycles" | "Bullhorns" | "Aluminio" |
| "State Bicycle Co" | "Bullhorns" | "Aluminio" |

Figura 7: Clases constituidas según dos criterios de clasificación

31. La frontera de estas tres clases en la devolución aparece cuando varía el contenido de la segunda columna o cuando, a contenido constante de la segunda columna, varía el contenido de la tercera, tal como lo materializan las líneas rojas trazadas sobre la tabla devuelta por la base.

32. Las consultas producen clasificaciones que corresponden a objetivos. En el caso de las Figuras 5 y 6, responden a la pregunta "¿A qué país pertenecen los productores de piezas de bicicleta?"; en el caso de la Figura 7, "¿Quiénes son los productores de los diferentes tipos de manubrio, según

sus respectivos materiales?". Las clases así generadas no tienen validez al margen de las consultas y son "efímeras" en el sentido en que no modifican el contenido de la base.

33. Los agrupamientos que consideramos aquí como clases se producen en la(s) columna(s) que constituye(n) el o los criterios de clasificación, no en la que exhibe las entidades clasificadas. Por este motivo, para leer la devolución en términos de clases, es necesario proyectar sobre la esta última los agrupamientos que aparecen en las que corresponden a los criterios de clasificación, como se describe en el análisis de las Figuras 5, 6 y 7.

# 5. Casos en que la clasificación es una herramienta auxiliar

34. Los ejemplos que siguen, que corresponden a tres trabajos de estudiantes de la promoción 2024, proponen clasificaciones cuyo objetivo no consiste en producir inventarios.

## 5.1. Informes aduaneros

35. Esta base de datos, diseñada y desarrollada por Mariano Robledo, tiene como objetivo proporcionar informaciones y estadísticas sobre entradas en el país de viajeros por puestos aduaneros diferentes (aeropuertos, puertos, rutas). Los viajeros presentan declaraciones de aduana en las que figuran el propósito del ingreso (turismo, negocios, residencia, etc.), sus países de origen y los valores de dinero o de mercancía que traen consigo. Estas informaciones permiten realizar estadísticas (como por ejemplo identificar viajeros que ingresan con una frecuencia inhabitual, declaran valores superiores a los autorizados sin impuesto, puestos aduaneros expuestos a la mayor cantidad de entradas de este u otro tipo, etc.).

36. Las dos consultas siguiente devuelven, por ejemplo, el número de pasajeros ingresados el primero de enero del 2024 por el aeropuerto de Ezeiza, según sus nacionalidades respectivas. El resultado es simplemente una lista de nacionalidades, puesto que se trata únicamente de la cantidad de pasajeros de cada una. La Figura 8 muestra dos formatos de devolución: el de la izquierda, donde cada nacionalidad se repite tantas veces como viajeros a la que les corresponde existan, el de la derecha, donde en lugar de repeticiones, la cantidad aparece representada numéricamente. En ambos casos, las nacionalidades clasifican a viajeros (el hecho de no hacer aparecer nombres de viajeros, indica que lo que se busca es la *cantidad por nacionalidades*, no otra cosa). "Nacionalidad" es, por consiguiente; el criterio de clasificación de los viajeros ingresados Las dos versiones que aparecen en la Figura 8 corresponden a repeticiones, sólo que la representación de las

repeticiones es diferente (por líneas con contenido repetido, a la izquierda, por número de repeticiones contadas, a la derecha).

| Nacionalidad |
|---|
| "Alemania" |
| "Argentina" |
| "Argentina" |
| "Argentina" |
| "Argentina" |
| "Bolivia" |
| "Brasil" |
| "Brasil" |
| "Brasil" |
| "Colombia" |
| "Corea Del Sur" |

| Nacionalidad | Cantidad |
|---|---|
| "Alemania" | 1 |
| "Argentina" | 4 |
| "Bolivia" | 1 |
| "Brasil" | 3 |
| "Colombia" | 1 |
| "Corea Del Sur" | 1 |
| "Costa Rica" | 1 |
| "Cuba" | 1 |
| "España" | 1 |
| "Estados Unidos" | 1 |
| "Guatemala" | 1 |

Figura 8: Nacionalidades ingresadas el 01-01-2024 por el aeropuerto de Ezeiza

37. Ambas consultas contienen una *selección* (que extrae de la base únicamente los viajeros que ingresaron por el puesto aduanero "Aeropuerto de Ezeiza" el 1° de enero del 2024), y devuelve una *clasificación* de este grupo por nacionalidades (en orden alfabético, según lo especifica el parámetro de ORDER BY incluido en la cláusula RETURN). La diferencia reside en que la segunda, en lugar de repetir la nacionalidad viajero por viajero, incluye el conteo de las líneas de

la primera. En el código Cypher de la segunda consulta siguiente se puede observar la instrucción de conteo ("count").

```
// Consulta 1: Nacionalidades por pasajeros sin conteo incluido
MATCH (p:Persona)-[:PRESENTA]->(d:Declaracion_Jurada)-[:EMITIDA_POR]->(a:Aduana)
WHERE d.fecha="2024-01-01" AND a.puesto_aduanero="Aeropuerto Ezeiza"
RETURN p.nacionalidad as Nacionalidad ORDER BY Nacionalidad
```

```
//Consulta 2: Nacionalidades por pasajero con conteo incluido
MATCH (p:Persona)-[:PRESENTA]->(d:Declaracion_Jurada)-[:EMITIDA_POR]->(a:Aduana)
WHERE d.fecha="2024-01-01" AND a.puesto_aduanero="Aeropuerto Ezeiza"
RETURN p.nacionalidad as Nacionalidad, count(p) as Cantidad ORDER BY Nacionalidad
```

38. La cantidad de viajeros por nacionalidad puede convertirse en un criterio de selección dentro de los resultados: si el interés del usuario es, por ejemplo, saber cuáles son las dos nacionalidades más representadas dentro de la selección definida en la Figura 8 (las restricciones expresadas por la cláusula MATCH), podrá modificar el parámetro de ORDER BY de la manera siguiente:

```
RETURN Nacionalidad, count(p) AS Cantidad, Aduana ORDER BY Cantidad DESC limit 2
```

en cuyo caso, el resultado (esta vez, completo) será el que aparece en la Figura 9.

| Nacionalidad | Cantidad | Aduana |
|---|---|---|
| "Argentina" | 4 | "Aeropuerto Ezeiza" |
| "Brasil" | 3 | "Aeropuerto Ezeiza" |

Figura 9: Las 2 nacionalidades más representadas en ingresos el 01-01-2024 por el aeropuerto de Ezeiza

39. Como se puede ver comparando el resultado expuesto en la Figura 8 con el de la Figura 9, ORDER BY puede servir para ordenar simplemente los resultados de la consulta pero también, combinándolo con LIMIT, por ejemplo, para seleccionar dentro del resultado un grupo que corresponda a posiciones específicas en un conjunto ordenado (los cinco primeros elementos, los tres últimos, etc.). Aquí se trata de las dos nacionalidades más representadas numéricamente dentro del grupo de viajeros ingresados. Cuando "Ordenar de mayor a menor" o la inversa es un paso intermedio para satisfacer los objetivos directos del usuario (aquí, saber cuáles son las dos nacionalidades más representadas por los pasajeros que ingresaron tal día por tal puesto aduanero), ORDER BY (combinado con LIMIT) pasa de ser una trivial operación de ordenamiento de resultados a ser una herramienta de post-selección.

40. Según una lógica similar a la que representan las devoluciones de la Figura 8, la siguiente devolución (Figura 10) clasifica a los viajeros ingresado por el puesto "Paso Libertadores" según sus objetivos  y su fecha de entrada (limitada al 1° y al 2 de enero en los datos de la base):

| motivo | Cantidad | Fecha |
|---|---|---|
| "Turismo" | 10 | "2024-01-01" |
| "Negocios" | 4 | "2024-01-01" |
| "Turismo" | 10 | "2024-01-02" |
| "Negocios" | 3 | "2024-01-02" |
| "Residente" | 1 | "2024-01-02" |

Figura 10: Motivos de ingreso por Paso Libertadores

41. Tal como se ha visto en la Figura 7, hay aquí dos criterios de clasificación: el motivo del ingreso y la fecha. La repetición de motivos ("Negocios", "Turismo"), a pesar de que la consulta incluye un conteo, indica que las entradas han sido clasificadas por motivos aunque lo son igualmente por fechas. Para visualizar los límites de una clase, la contigüidad de todas las ocurrencias de un valor repetido en las columnas indica las particiones (2 clases), y determina que aquí que la fecha es el primer criterio de clasificación, el motivo, el segundo, puesto que las repeticiones de motivos no son contiguas. Lo que se clasifica según estos dos criterios son los ingresos.

42. Razonando íntegramente por conteos, resulta simple establecer estadísticas. La consulta que sigue tiene por objetivo conocer el porcentaje de las entradas con motivo de turismo registradas el 1° de enero del 2024 en el Paso Libertadores (la devolución que aparece en la Figura 11).

```
MATCH (a:Aduana)<-[:EMITIDA_POR]-(d)<-[r:PRESENTA]-(p)
WHERE a.puesto_aduanero= "Paso Libertadores" AND d.fecha="2024-01-01"
WITH p, CASE WHEN r.motivo = "Turismo" THEN 1 ELSE NULL END as turista
RETURN count(p) as TotalViajeros, count(turista) AS TotalTuristas, 100 * count(turista) / count(p) AS PorcentajeTuristas
```

| TotalViajeros | TotalTuristas | PorcentajeTuristas |
|---|---|---|
| 14 | 10 | 71 |

Figura 11: Porcentaje de turistas entre viajeros ingresados por Paso Libertadores el 01-01-2024

43. Como un indicio para el descubrimiento de fraudes, entradas del mismo viajero juzgadas como frecuentes pueden combinarse con declaraciones de exceso de divisas o mercancías. La consulta correspondiente a la Figura 12 identifica a los viajeros cuya edad está comprendida entre 20 y 40 años y que han declarado exceso de divisas o de mercancías al ingresar. Se añade la fecha de ingreso y el puesto aduanero utilizado.

```
MATCH (p:Persona)-[:PRESENTA]->(d:Declaracion_Jurada)-[:EMITIDA_POR]->(a:Aduana)
WHERE p.edad>=20 and p.edad<=40
WITH p,d,a
MATCH(d)-[:SE_ENLISTAN]->(m:Divisas) WHERE m.exceso_usd>5000
MATCH(d)-[:SE_ENLISTAN]->(e:Mercancia) WHERE e.valor_usd>500
RETURN d.fecha as Fecha, p.nombre as Nombre, a.puesto_aduanero as Aduana, m.moneda+"
"+m.exceso_usd as Divisas,e.descripcion as Mercancia, e.valor_usd as Valor_USD order by
Aduana
```

| Fecha | Nombre | Aduana | Divisas | Mercancia | Valor_USD |
|---|---|---|---|---|---|
| "2024-01-01" | "Lucía Nacarro" | "Aeropuerto El Plumerillo" | "USD 7779" | "Laptop" | 2197 |
| "2024-01-02" | "Valentina López" | "Aeropuerto Ezeiza" | "USD 18120" | "Perfume" | 2076 |
| "2024-01-01" | "Leandro Pereira" | "Foz de Iguazú" | "BRL 16191" | "Reloj" | 1295 |
| "2024-01-02" | "Seung-Ho Kim" | "Puerto Buenos Aires" | "EUR 18797" | "Celular" | 1301 |

Figura 12: Viajeros ingresados con divisas y/o mercancías a declarar

## 5.2. Sistema de recomendación para transferencia de futbolistas

44. Esta base de datos, diseñada y desarrollada por Mauro Sorbello e Ignacio Coppede, tiene como objetivo orientar a clubes de fútbol que proyecten adquirir jugadores, mediante recomendaciones de candidatos a una posición de juego específica, evaluados según su perfil y su desempeño en las últimas cinco temporadas. Antes de comentar consultas y devoluciones de la base, un breve análisis del modelo de la estructura de los datos utilizado presenta interés, por lo escueto, por un lado, pero también porque las *propiedades de relaciones*, propias a las bases grafo, constituyen aquí el elemento primordial (Figura 13).

45. La única clase con instancias no predefinidas es "Jugador". A pesar de no aparecer en el gráfico la nómina de los clubes tomados en consideración por los responsables del proyecto, se trata de una lista cerrada (=/). Dentro de las propiedades de la relación "Participa", que conecta "Juga-

dor" con "Temporada", aparecen todas las propiedades que permiten evaluar a un jugador en las temporadas que median entre 2018 y 2023, según su posición en el juego: "penales_atajados", por ejemplo, se refiere únicamente a jugadores con la posición "Arquero". Las propiedades con fondo negro y tipografía roja no corresponden a datos ingresados sino cálculos a efectuar a partir de éstos.

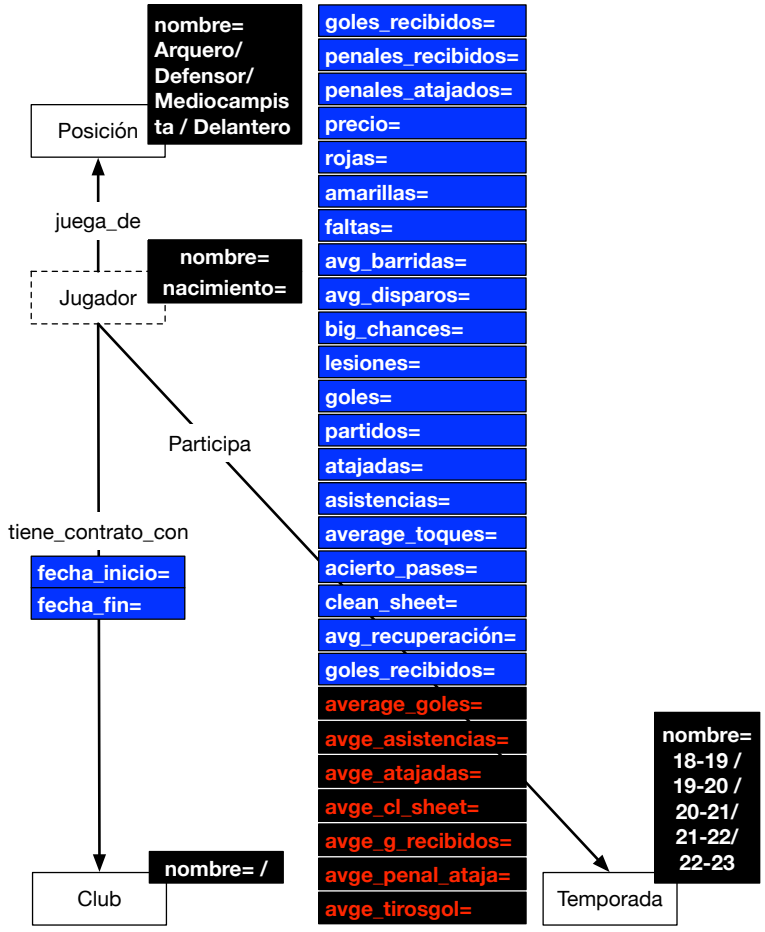



Figura 13: Modelo de la estructura de datos del proyecto de recomendaciones de futbolistas

46. Lo peculiar en este proyecto, es que los únicos contenidos que permiten cumplir con el objetivo (recomendar jugadores en función de su desempeño en las cinco últimas temporadas), no son los datos almacenados sino los valores calculados en función de éstos. Así, por ejemplo, para evaluar el rendimiento de un jugador en las cinco temporadas que se tienen en cuenta, se calcula, sobre la base de los partidos en que intervino (el valor numérico de la propiedad "partidos"), cuál es la media de partidos disputados, pero también su dispersión dentro del período considerado, de modo de diferenciar a los jugadores que intervienen irregularmente de los que lo hacen a un ritmo relativamente constante. Teniendo en cuenta estas variables en la comparación del rendimiento de Paul Pogba y Bernardino Silva, por ejemplo, el resultado de la consulta que sigue (Figura 14), indica que Silva tiene no sólo una media superior de partidos jugados (33,0

contra 20,6 de Paul Pogba), pero también una regularidad superior en sus intervenciones (12,12 de media de dispersión contra 52,68 para Paul Pogba). Según este criterio, Silva aparece así como superior a Pogba.

```
MATCH (j:Jugador)-[r:PARTICIPA]->(t:Temporada)
WHERE (r.partidos) IS NOT NULL
WITH j, collect(r.partidos) AS partidos
WHERE size(partidos) > 1 AND j.nombre IN ['Paul Pogba', 'Bernardo Silva']
WITH j, partidos,
REDUCE(s = 0.0, x IN partidos | s + x) / SIZE(partidos) AS media
WITH j, partidos, media,
SQRT(REDUCE(s = 0.0, x IN partidos | s + (x - media)^2) / (SIZE(partidos) - 1)) AS desviacion_estandar
WITH j, media, desviacion_estandar,
CASE WHEN media <> 0 THEN (desviacion_estandar / media) * 100 ELSE NULL END AS coeficiente_variacion_partidos
RETURN j.nombre AS Jugador, round(coeficiente_variacion_partidos,3) AS Dispersion_de_la_Media_de_Partidos_Disputados, media AS Media_de_Partidos_Disputados
```

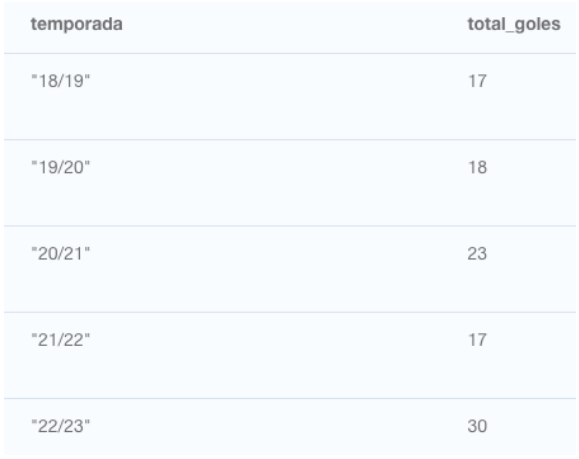
| Jugador | Dispersion_de_la_Media_de_Partidos_Disputados | Media_de_Partidos_Disputados |
|---|---|---|
| "Paul Pogba" | 52.687 | 20.6 |
| "Bernardo Silva" | 12.121 | 33.0 |

Figura 14: Rendimiento comparado de dos jugadores según frecuencia de participación y regularidad

47. El ordenamiento de los resultados (las dos columnas de la derecha) clasifica a los dos jugadores en dos grupos de un elemento cada uno (Silva, el de los más eficientes, Pogba, el de los menos). Hubiese podido ocurrir que, según el segundo criterio, la media de Pogba fuera menor que la de Silva. En este caso, sería imposible clasificar simultáneamente según los dos criterios: Silva sería mejor en términos de media de partidos disputados, pero Pogba mejor en términos de dispersión, es decir, más regular en sus intervenciones. Por consiguiente, sería necesario definir cuál de ambos criterios es prioritario para medir la calidad de los jugadores. Queda nuevamente en claro aquí que, en cualquiera de los casos, la clasificación es un producto de la consulta, y que, por otra parte, salvo casualidad, habrá tantas clases como jugadores tomados en cuenta, salvo si se razona en términos de rango (de tal a tal valor de media o de dispersión, una clase, de tal a tal valor, otra, y así sucesivamente).

48. Consultas más sencillas permiten clasificar las temporadas para un mismo jugador basándose, si se trata de un goleador, en la cantidad de goles realizados (consulta siguiente, Figura 15),

```
MATCH (a:Jugador {nombre: "Harry Kane"})-[r:PARTICIPA]->(temporada:Temporada)
RETURN temporada.nombre AS temporada, SUM(r.goles) AS total_goles
ORDER BY temporada.nombre
```



| temporada | total_goles |
|---|---|
| "18/19" | 17 |
| "19/20" | 18 |
| "20/21" | 23 |
| "21/22" | 17 |
| "22/23" | 30 |

Figura 15: Cantidad de goles para un mismo jugador en las 5 temporadas

clasificando así temporadas en mejores y peores según este criterio, para el mismo jugador. El interés consiste en descubrir si hay una progresividad de rendimiento (temporadas recientes con más goles que las anteriores). De modo similar, es posible obtener la efectividad según este criterio, medida en términos de media de rendimiento, calculando la relación entre tiros de gol y goles efectivos (consulta siguiente, Figura 16).

```
MATCH (jugador:Jugador)-[r:PARTICIPA]->(temporada:Temporada)
MATCH (jugador)-[j:JUEGA_DE]->(pos: Posicion {nombre: "Delantero"})
WITH temporada.nombre AS Temporada, jugador, r.goles AS Cantidad_Goles, r.average_ti-
rosgol AS Average_Efectividad, r.average_goles AS Goles_Por_Partido, pos, r
RETURN 'Delantero goleador' AS Titulo, Temporada, jugador.nombre, Goles_Por_Partido,
Cantidad_Goles,Average_Efectividad
ORDER BY Temporada, Goles_Por_Partido DESC
```

49. Una presentación más completa de la utilidad y de las funcionalidades de esta base y de las clasificaciones que autoriza puede visualizarse en el video de presentación del proyecto que Mauro e Ignacio han puesto a disposición en el enlace https://drive.google.com/file/d/1XqJgL7mrm3i-F3kQjVV0ds6lc-ezP5Bdo/view?usp=drivesdk, cuya perennidad es, como la de cualquier contenido en el universo digital, difícil garantizar (en caso de que el enlace dejase de funcionar, los lectores pueden solicitar su reactivación a la dirección indicada en la portada de este informe.

50. Queda ahora a la apreciación de los lectores el interés y el valor del curso descrito, con la salvedad de que la totalidad de lo que aquí aparece corresponde a un trabajo de tres semanas.

| Titulo | Temporada | jugador.nombre | Goles_Por_Partido | Cantidad_Goles | Average_Efectividad |
|---|---|---|---|---|---|
| "Delantero goleador" | "18/19" | "Robert Lewandowski" | 0.667 | 22 | 0.152 |
| "Delantero goleador" | "18/19" | "Harry Kane" | 0.607 | 17 | 0.169 |
| "Delantero goleador" | "18/19" | "Karim Benzema" | 0.583 | 21 | 0.201 |
| "Delantero goleador" | "18/19" | "Mohamed Salah" | 0.579 | 22 | 0.161 |
| "Delantero goleador" | "18/19" | "Heung-min Son" | 0.387 | 12 | 0.161 |
| "Delantero goleador" | "18/19" | "Anthony Martial" | 0.37 | 10 | 0.265 |
| "Delantero goleador" | "18/19" | "Marcus Rashford" | 0.303 | 10 | 0.121 |
| "Delantero goleador" | "18/19" | "Thomas Müller" | 0.188 | 6 | 0.0 |
| "Delantero goleador" | "18/19" | "Ángel Correa" | 0.0 | 0 | 0.0 |
| "Delantero goleador" | "19/20" | "Robert Lewandowski" | 1.097 | 34 | 0.244 |

Figura 16: Goleadores clasificados por media de tiros gol convertidos

# 6. Conclusión

51. El curso descrito pone de manifiesto el interés de relacionar el aprendizaje de herramientas informáticas con la necesidad de resolver problemas reales *expuestos previamente*, en lugar de presentarlas bajo la forma de conocimientos y habilidades cuya aplicación práctica "se verá después". Considerado bajo el punto de vista del aprendizaje de la clasificación, el curso subraya el beneficio de aprovechar las intuiciones de los estudiantes para construir abstracciones *a partir de ellas* en lugar de presentarlas como conceptos desconectados de sus conocimientos previos. Por sobre todo, haber relacionado una actividad tan elemental como la de agrupar elementos por sus características compartidas con los antecedentes históricos de nociones como "objeto", "propiedad", "herencia" o "taxonomía", tan centrales en el campo conceptual de la clasificación como en el de la informática, sitúa explícitamente lo aprendido en las tres semanas del curso

dentro de una tradición intelectual con 25 siglos de existencia y, de este modo, integra a los estudiantes en la historia, no como meros espectadores, sino como continuadores activos (y, como se puede observar, a veces, entusiastas).

# 7. Apéndice

**Convenciones gráficas para diseñar modelos de datos**

| ENTIDAD | TIPO | SUBTIPO | REPRESENTACIÓN |
|---|---|---|---|
| Grupos | creación de individuos con ≥ 1 valor de propiedad a **ingresar** | | Cliente |
| | creación de individuos con ≥ 1 valor de propiedad a **seleccionar** | | Uso |
| | **Grupo abstracto** (sin instancias directas) | | (bien) |
| Propiedades | de individuos | a **ingresar** | **precio en $ =** |
| | | a **seleccionar** | **tipo = /** |
| | | a **calcular** | **fecha de renovación** |
| | de relaciones | a **ingresar** | **cantidad =** |
| | | a **seleccionar** | **estatuto = /** |
| | | a **calcular** | **número de bienes** |
| Relaciones | de subgrupo a grupo | | Coche ↑ Coche deportivo — Un coche deportivo **es un tipo de** coche |
| | de grupo a grupo | | (bien) — PROPIETARIO → Cliente<br>Un bien tiene un propietario que es un cliente<br>Un cliente es el propietario de un bien |
| Individuos | de un grupo terminal | | la casa de Pedro — Edificio — La casa de Pedro es un edificio |
| | de un grupo abstracto por medio de un grupo terminal | | la casa de Pedro — Edificio ⟹ (bien)<br>La casa de Pedro es un bien |