

Modelo en grafo

Máster en Business Analytics y Big Data

Bases de Datos No Convencionales

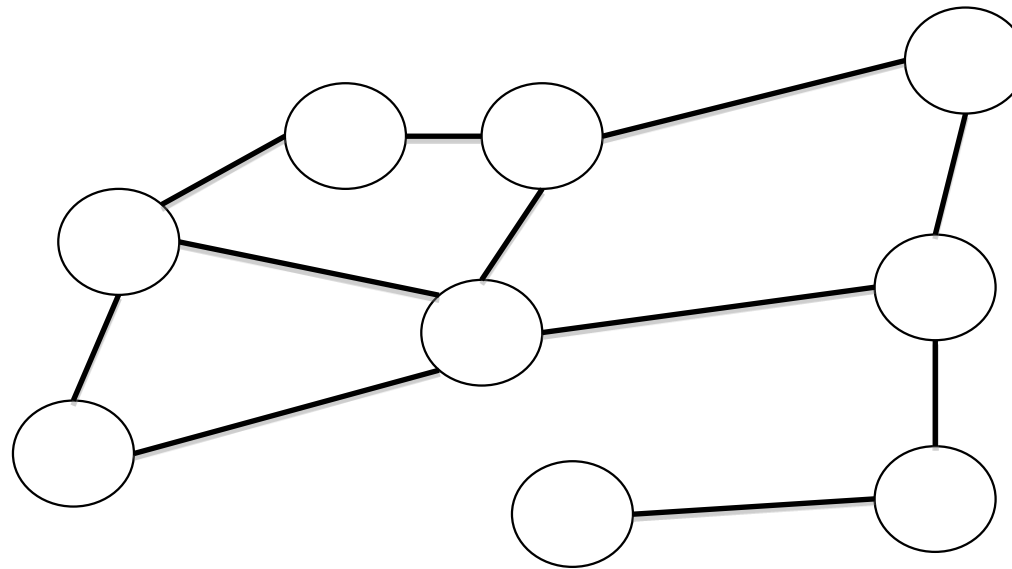


Contenidos de la sesión

- Introducción
- Ejemplo
- Modelo en grafo
- Uso de modelos en grafo

Modelo en grafo y grafos

- El modelo en grafo utiliza estructuras de grafo para representar y almacenar los datos.
- ¿Grafos?



Nodos

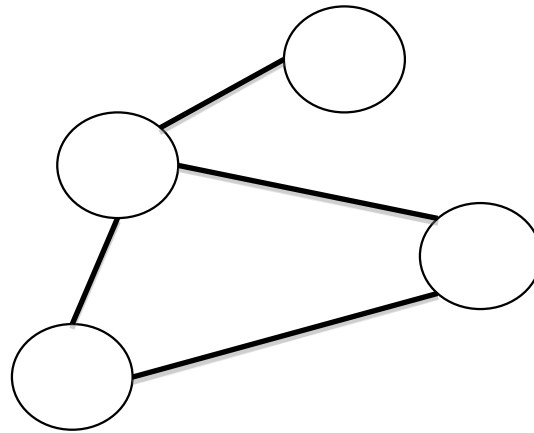
Aristas

Grafos: conceptos básicos

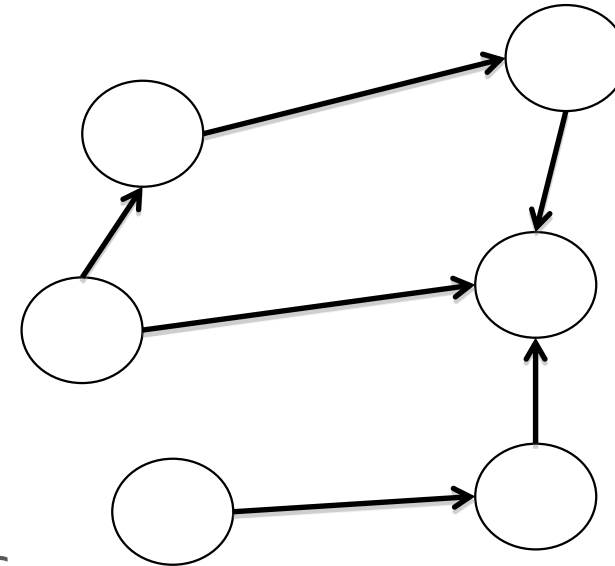
- Tipos de aristas

—————

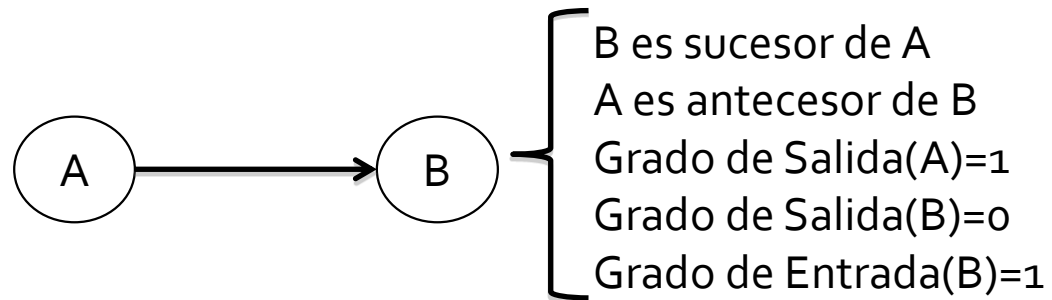
- Tipos de grafos



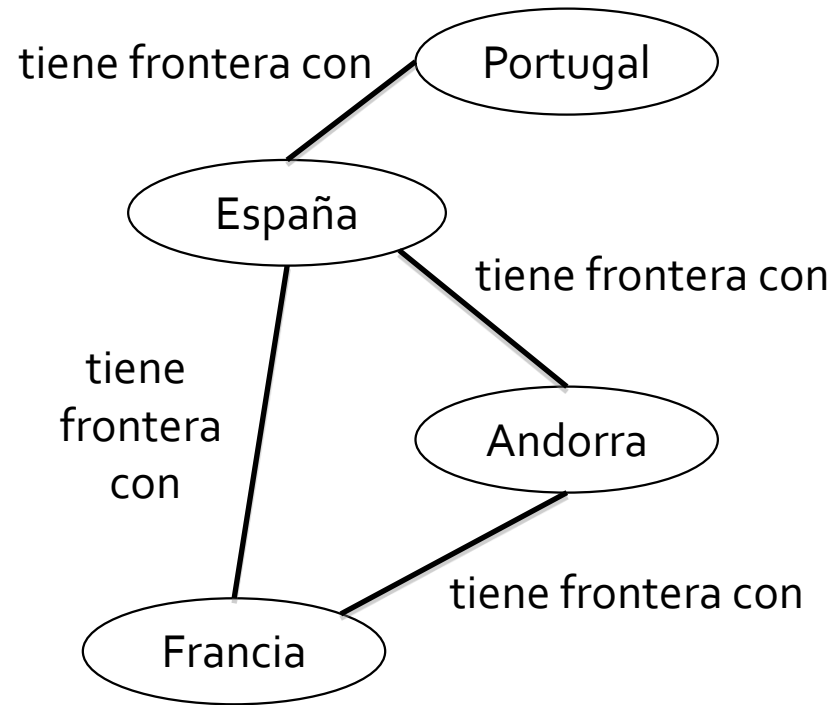
—————→



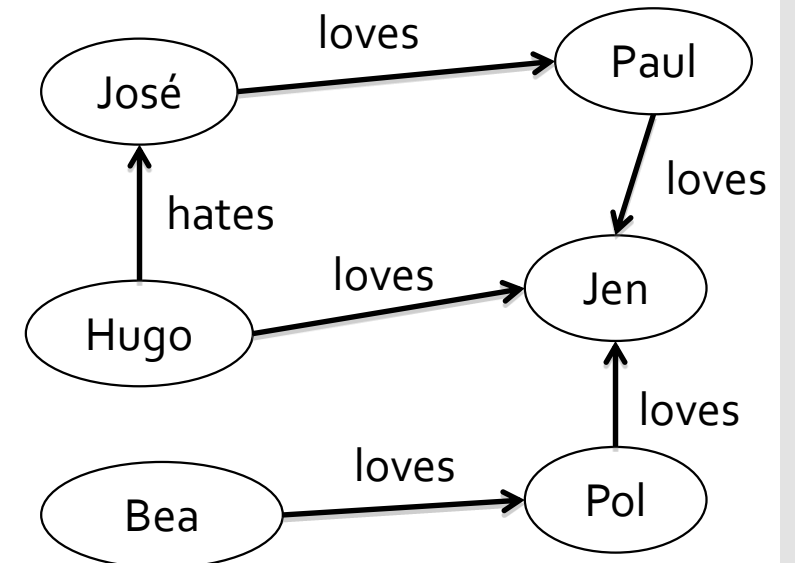
- Si se usan aristas dirigidas...



Grafos: etiquetas



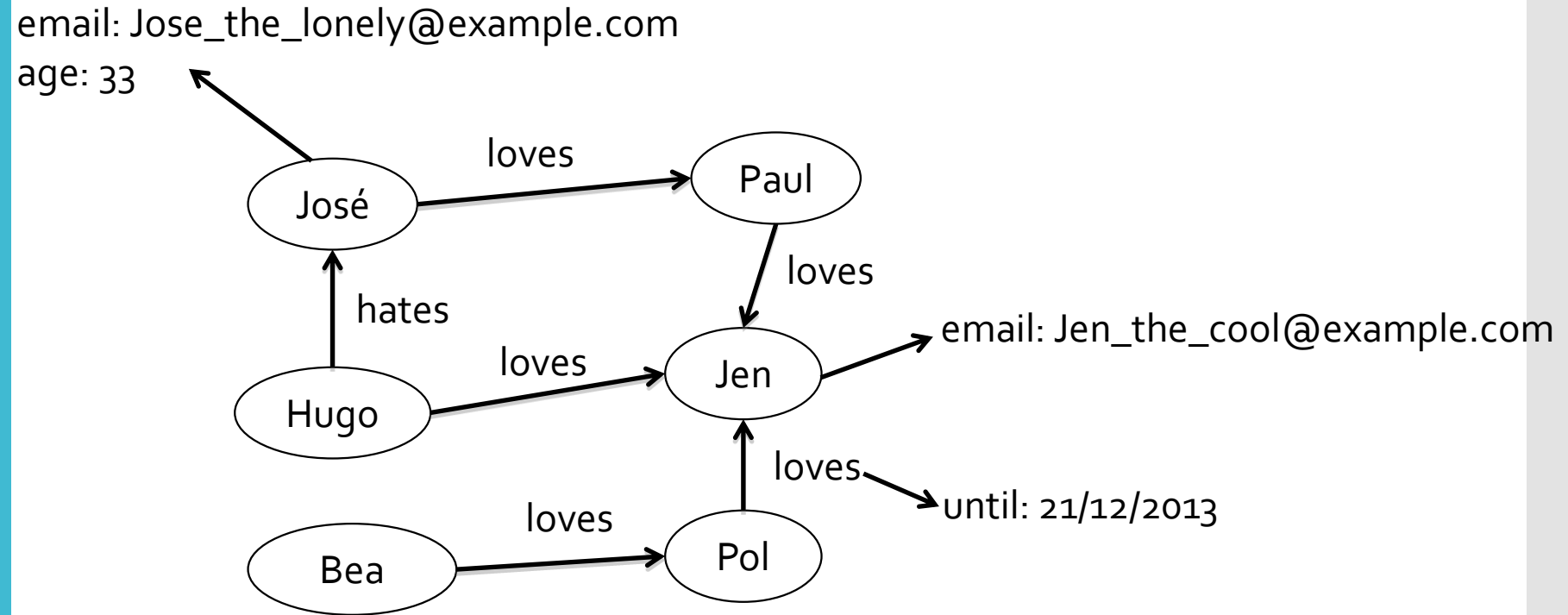
Grafo no dirigido etiquetado



Grafo dirigido etiquetado

Grafos: propiedades

- Pueden asignarse tanto a nodos como a aristas.
- Están formadas por un par <clave, valor>.



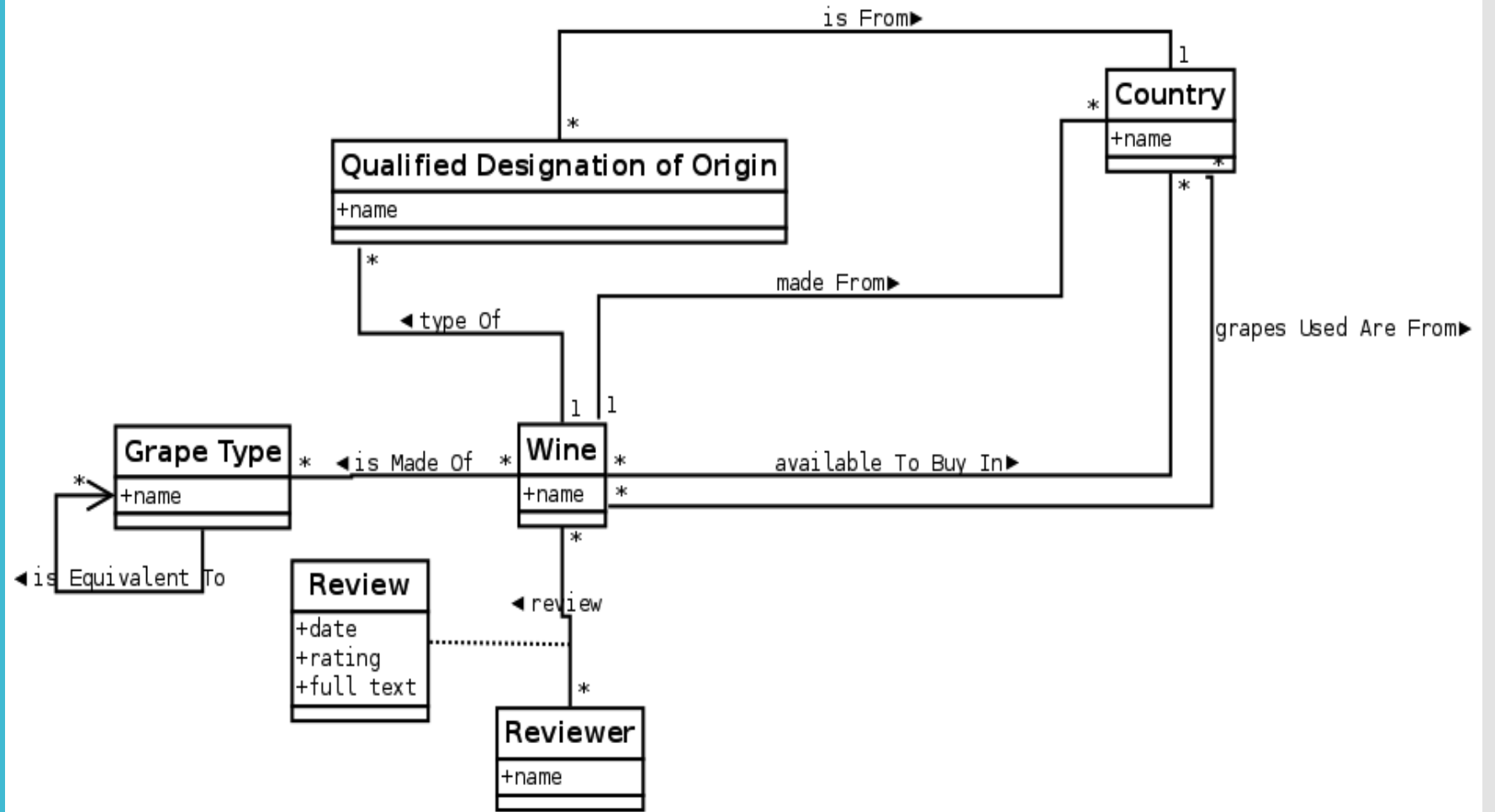
Introducción

- El modelo en grafo utiliza estructuras de grafo para representar y almacenar los datos.
- Es especialmente útil cuando:
 - Existen muchas relaciones en los datos
 - La importancia de los datos recae mayoritariamente en sus relaciones.
- Algunos ámbitos de aplicación son:
 - Redes (redes sociales, logística, mapas, análisis de redes)
 - Aplicaciones semánticas
 - Biología computacional
- Existen adaptaciones del modelo en grafo que permiten almacenar ontologías (*triplestores*).

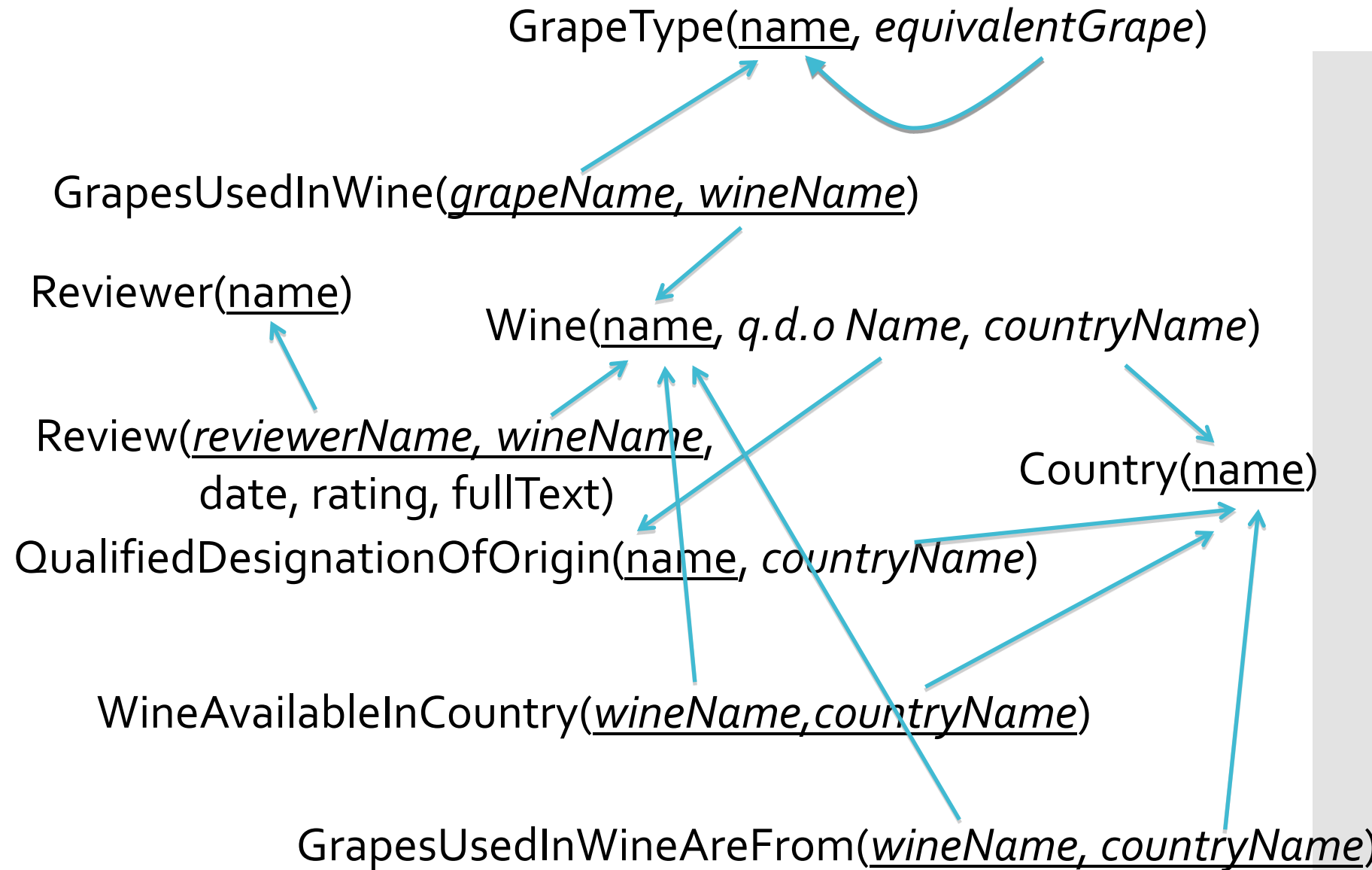


Ejemplo

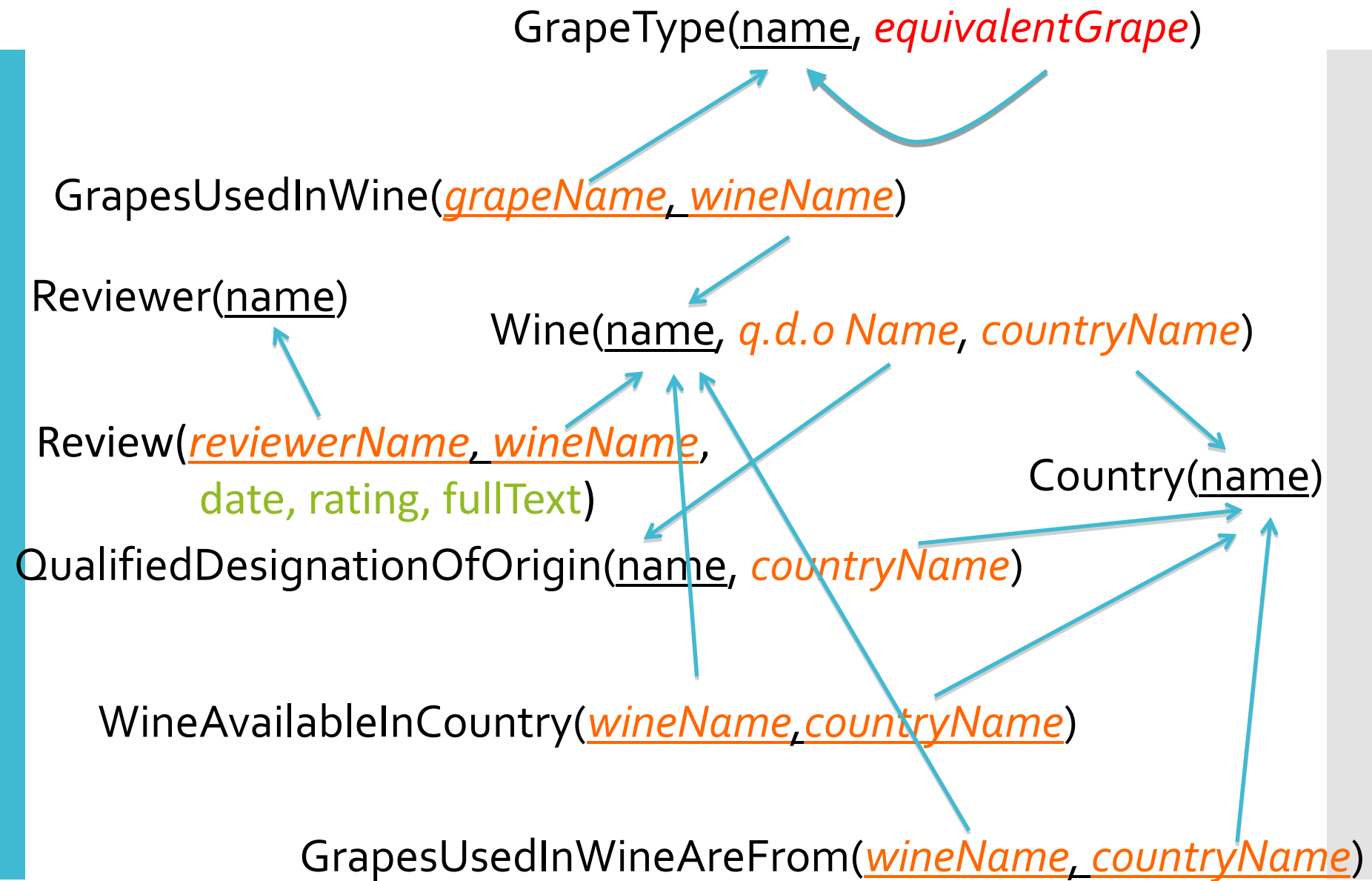
Por ejemplo,
en una
aplicación web
de vinos ...



... SU
representación
relacional no
es muy clara ...



... además,
¿Cuáles son
realmente
datos del
dominio de
interés?



... además,
¿Cuáles son
realmente
datos del
dominio de
interés?

GrapeType(name, *equivalentGrape*)

GrapesUsedInWine(*grapeName*, *wineName*)

20 atributos: 12 de ellos son claves foráneas!!!

Se usan más atributos para representar relaciones que datos del dominio de interés.

Si necesitamos más de la mitad de la base de datos para representar las relaciones quizá necesitemos otro modelo.

WineAvailableInCountry(*wineName*, *countryName*)

GrapesUsedInWineAreFrom(*wineName*, *countryName*)



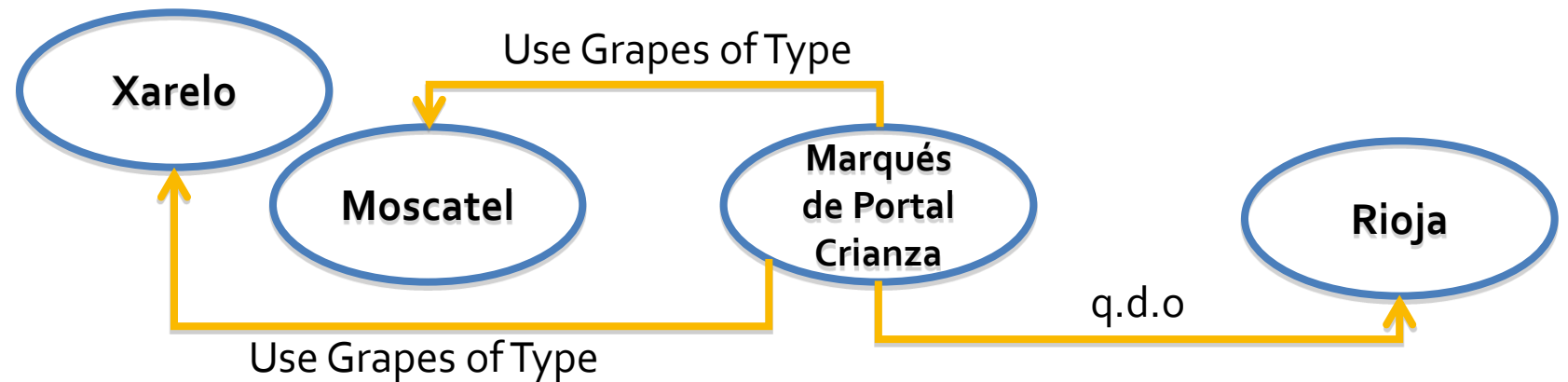
Modelo en grafo

Modelo en grafo: elementos básicos

- Nodos: permiten representar conceptos generales u objetos del mundo real.



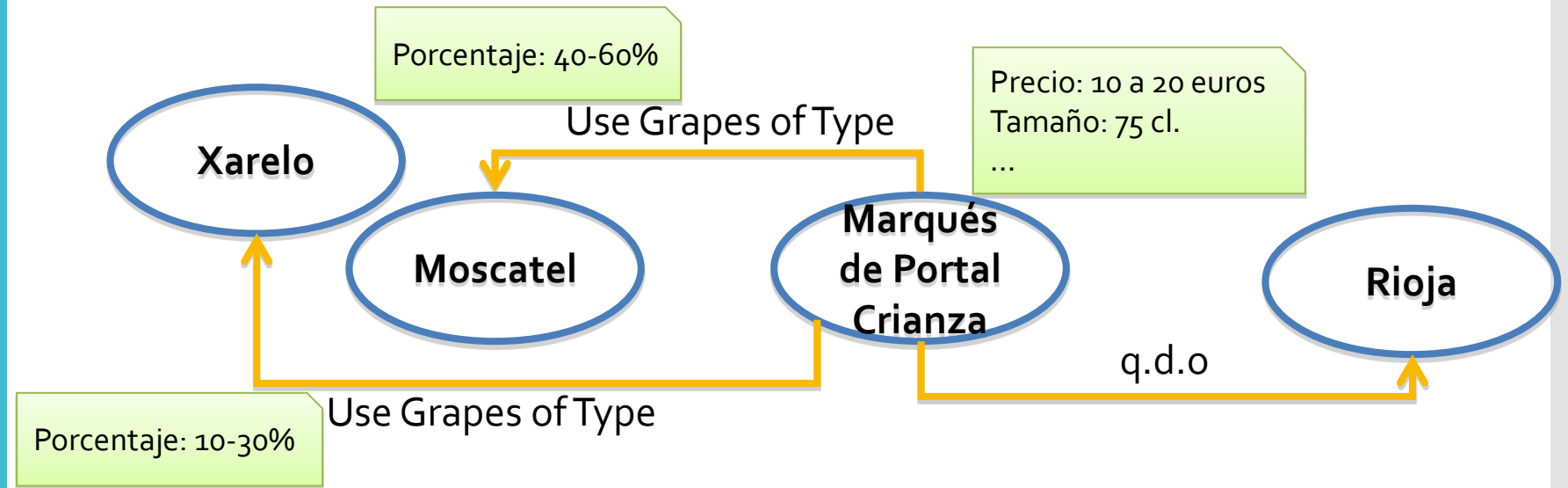
- Aristas: permiten representar de forma explícita las relaciones entre los nodos.



- Los nodos y las aristas se pueden etiquetar (bautizar).

Modelo en grafo: elementos básicos

- pueden añadir propiedades a los nodos y a las aristas



- Restricciones del modelo:
 - No permite definir aristas sin nodos origen ni destino.
 - Los nodos sólo pueden eliminarse cuando quedan huérfanos.

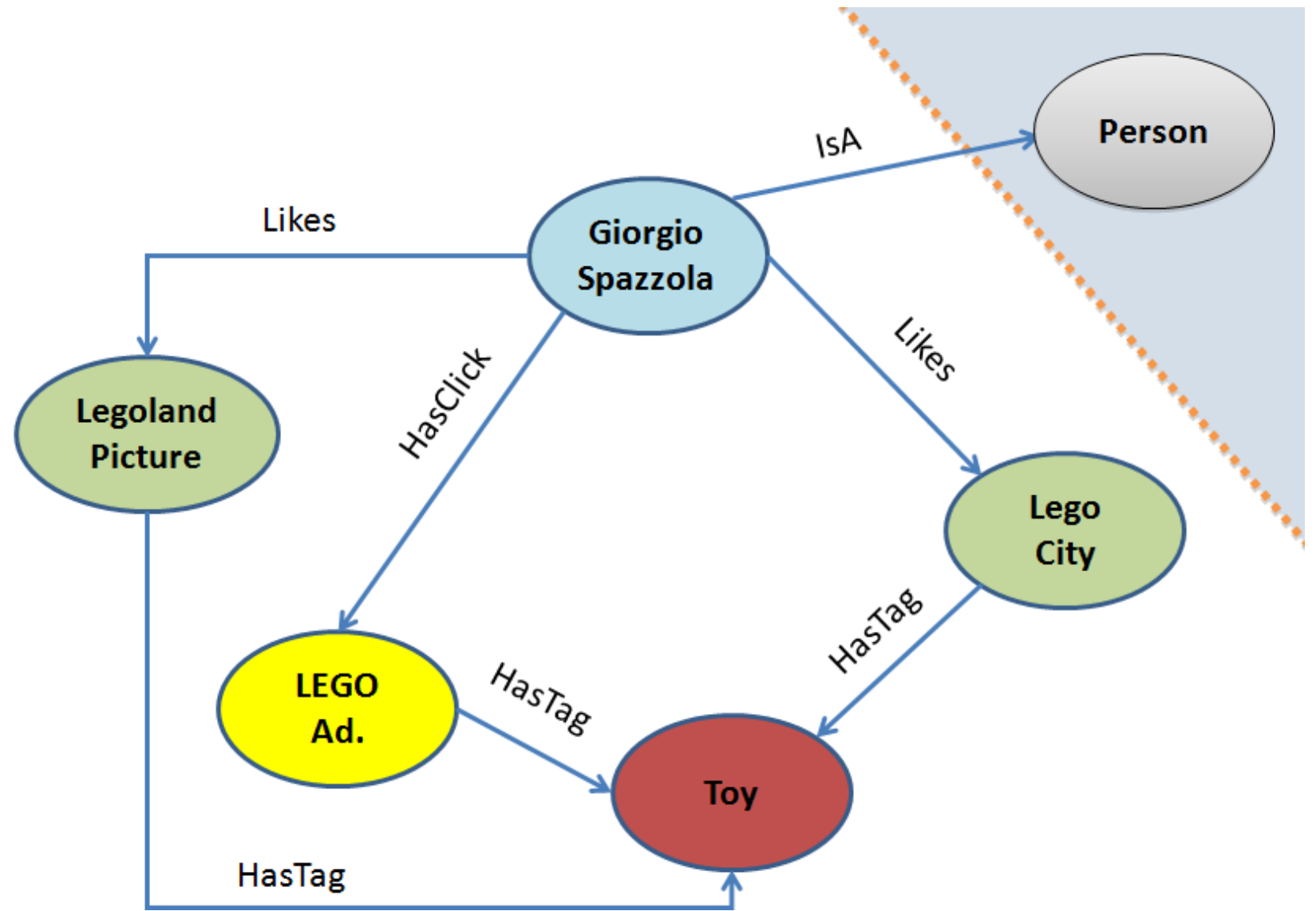
Modelo en grafo: características

- Las relaciones son explícitas entre entidades.
 - Mejora el tiempo de respuesta al navegar entre relaciones (no es necesario calcular las operaciones de join).
- *Schemaless*: el esquema está implícito en la estructura del grafo.
- No son tan fácilmente escalables como los modelos de agregación.

Modelo en grafo: características

- Respecto a los SGBD que los implementan:
 - Garantizan consistencia a través de transacciones: comúnmente ACID.
 - Suelen proveer de lenguajes de consulta de alto nivel.

Ejemplo



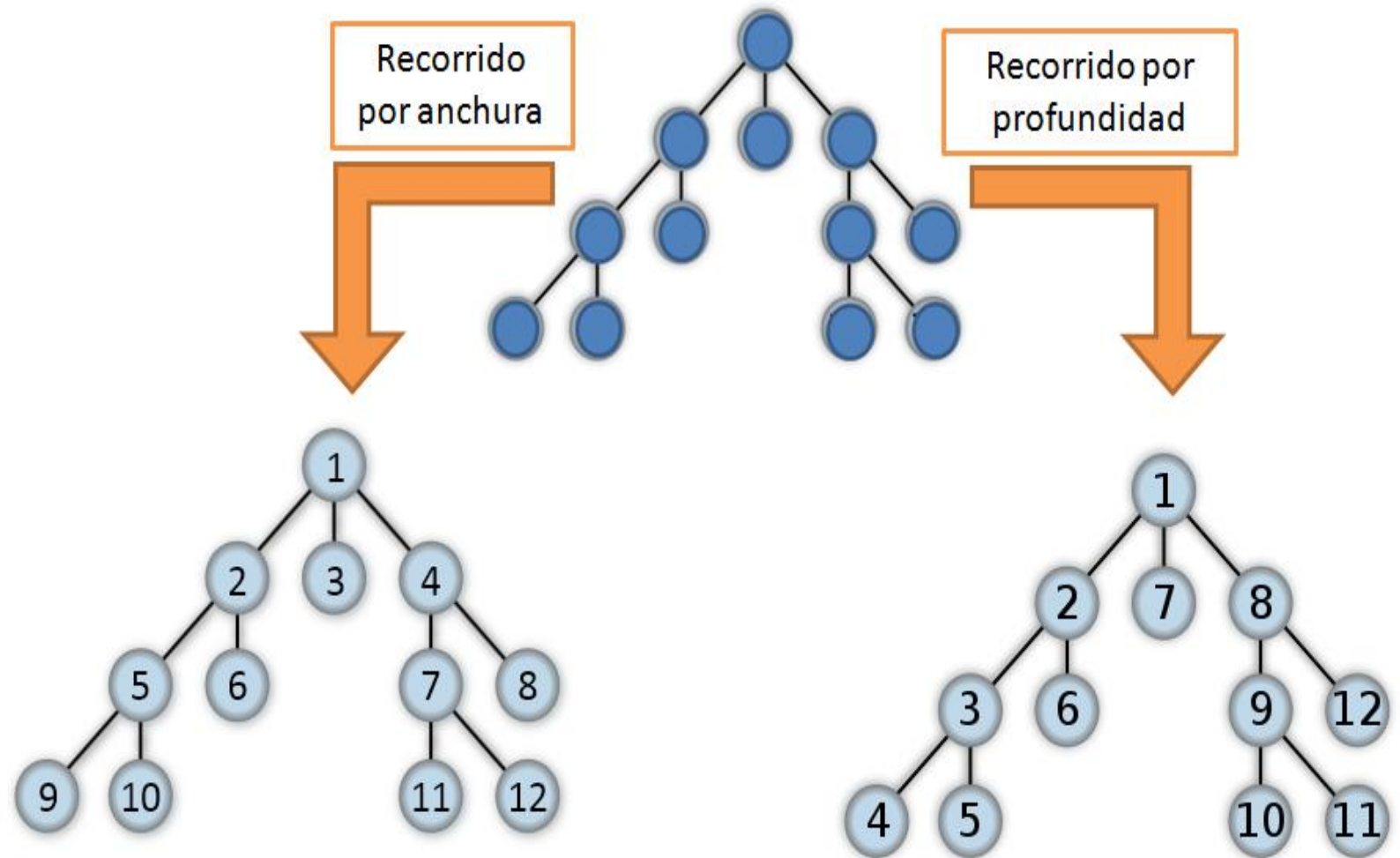


Uso de modelos en grafo

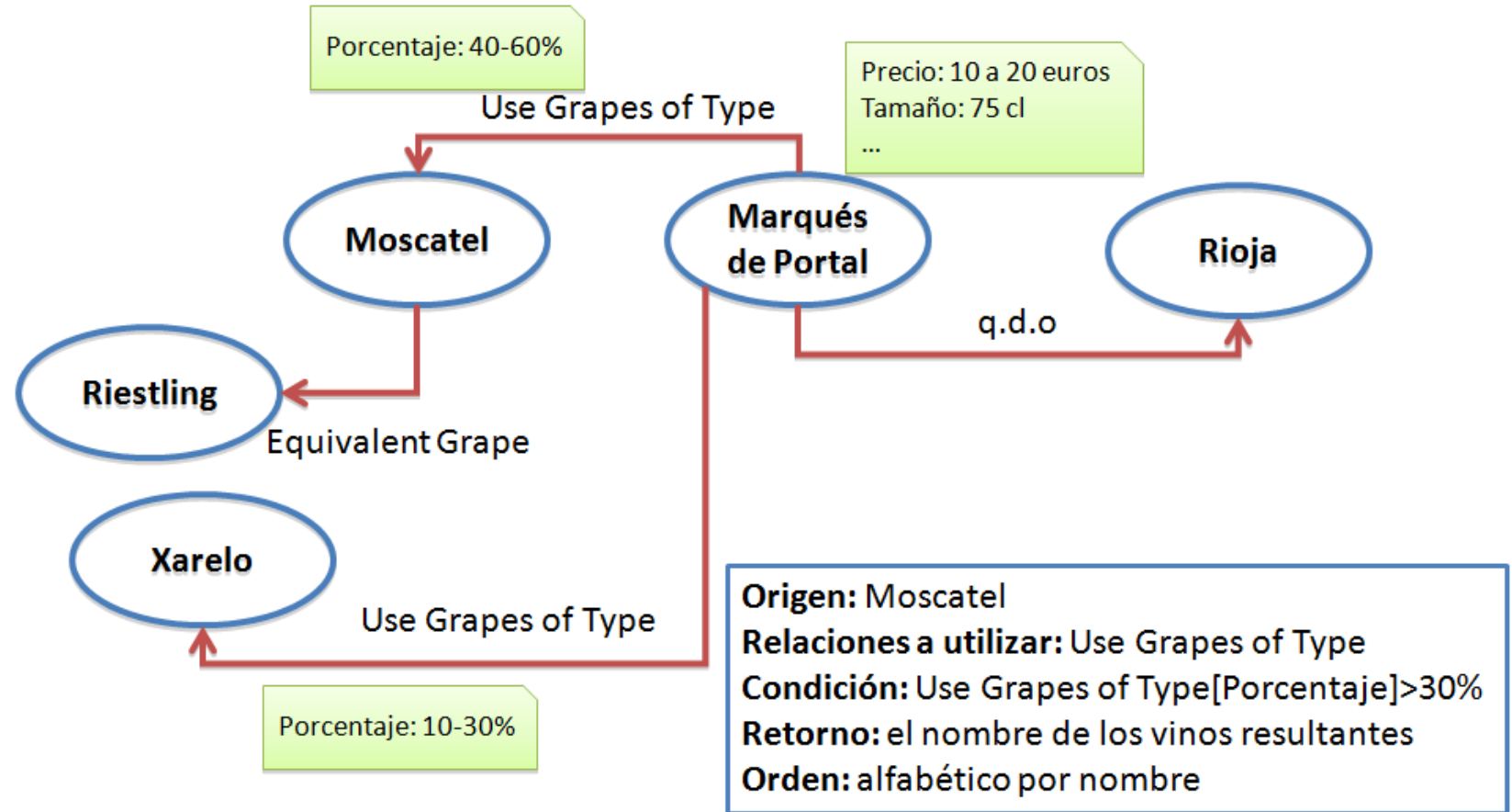
Acceso a los datos

- El acceso a los datos en un modelo en grafo se realiza mediante navegación, donde se define:
 - Origen de la búsqueda
 - Patrón de navegación:
 - Qué relaciones (arcos) utilizar y cuándo
 - Condiciones que deben satisfacer los datos
 - Tipo de navegación (profundidad vs anchura)
 - Nivel de profundidad
 - Otros:
 - Nodos, aristas y propiedades a devolver
 - Orden
 - Etc.

Recorrido por anchura y profundidad



Acceso a los datos: ejemplo



Consideraciones de diseño

- No requiere la creación explícita de un modelo de datos, pero es importante hacerlo a priori para:
 - Identificar las relaciones que existen entre los datos
 - Evitar utilizar nombres distintos para la misma relación semántica
 - Añadir al grafo conceptos cuando puedan ser útiles en la explotación de datos
- Se debe tener en cuenta que, aunque las relaciones sean dirigidas, pueden recorrerse en ambos sentidos.
 - Dirección implica semántica, no navegabilidad.

Extensiones: almacenes RDF

- Permiten almacenar tripletas RDF:
URI –[URI predicate]--> URI o valor escalar
- Los datos RDF pueden verse como un grafo dirigido etiquetado.
- El modelo de datos es simple y está consensuado.
- Existe un lenguaje de consulta potente y estándar
- Formatos de intercambio de datos estándar
- Más eficiencia al tratar datos RDF



Recursos y enlaces

Lecturas

- I. Robinson, J. Webber & E. Eifren (2013). *Graph Databases*, O'Reilly. (<http://graphdatabases.com/>)
- M. Gyssens & J. Paredaens (1990). "A Graph Oriented Object Database Model", *ACM SIGMOD International Conference*. (<http://bit.ly/JL2QWc>).
- P.J. Sadalage & M. Fowler. (2013). *NoSQL Distilled. A brief Guide to the Emerging World of Polyglot Persistence*, Pearson Education. (<http://bit.ly/1koKhBZ>).
- Joe Celko's (2013). *Complete Guide to NoSQL*. Elsevier. (<http://www.sciencedirect.com/science/book/9780124071926>)
- E. Redmond, J Wilson (2012). *Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement*, The Pragmatic Bookshelf. (<http://pragprog.com/book/rwdata/seven-databases-in-seven-weeks>)