

# Ingeniería y Gestión del Conocimiento RDFS

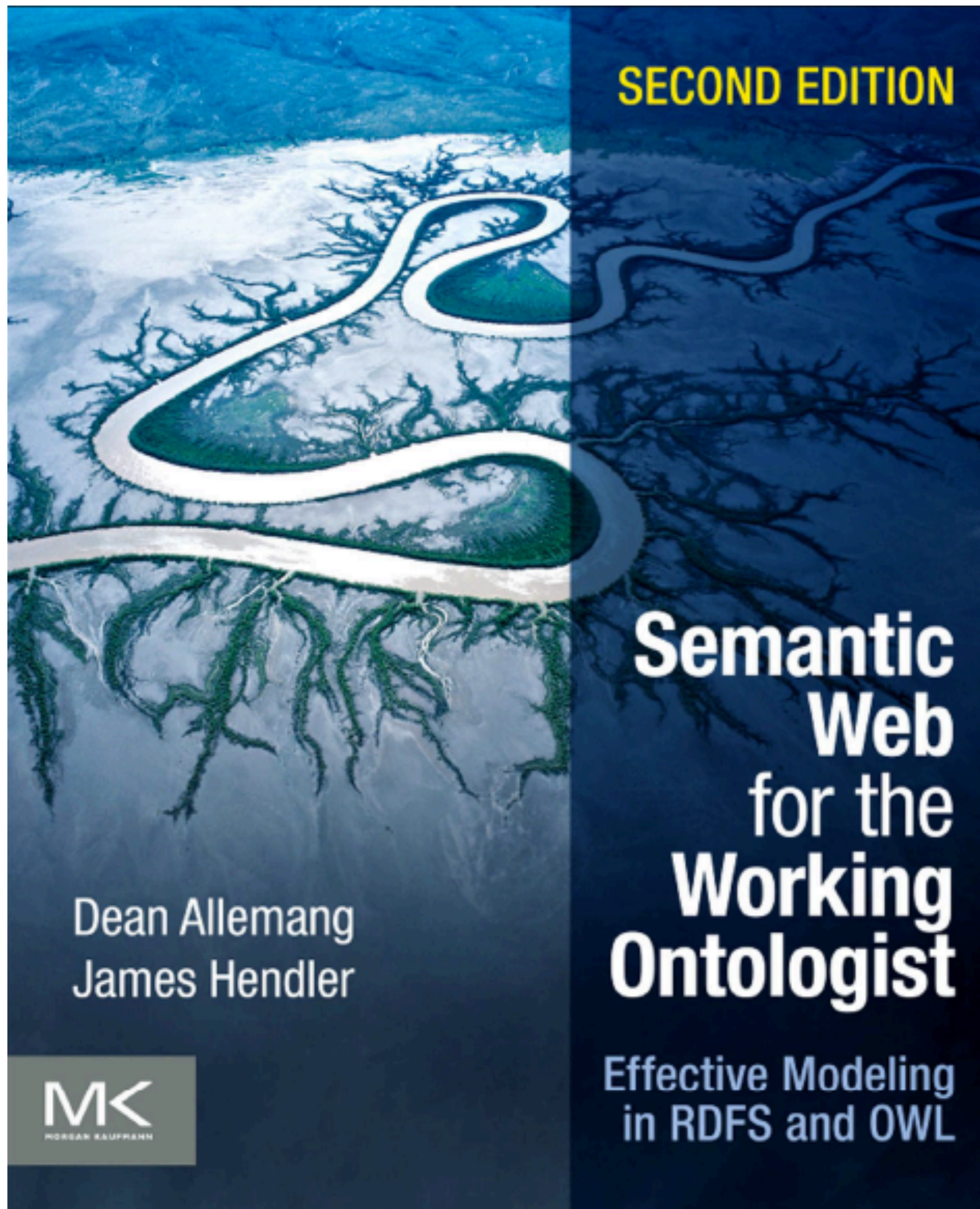
Profesor  
Andrés Melgar

# Agenda

- Introducción.
- Lenguaje de Esquema.
- Relación entre esquema y datos.
- El lenguaje de esquema RDF.
- Inferencia en RDF (subClassOf).



# Texto guía



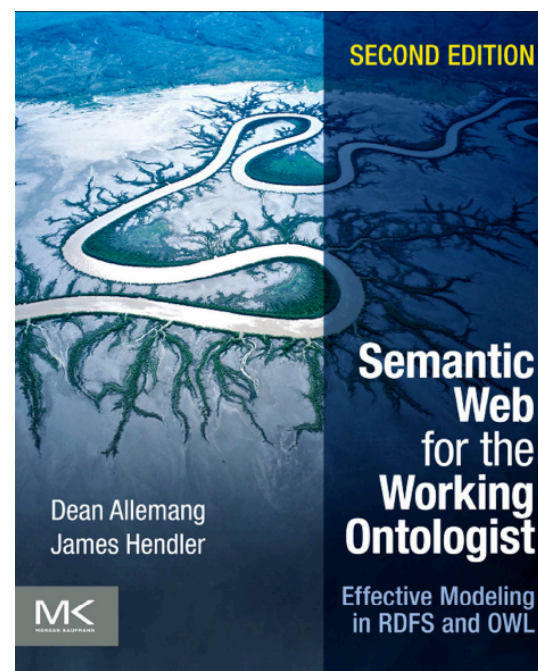
CHAPTER

RDF schema

7



# Introducción

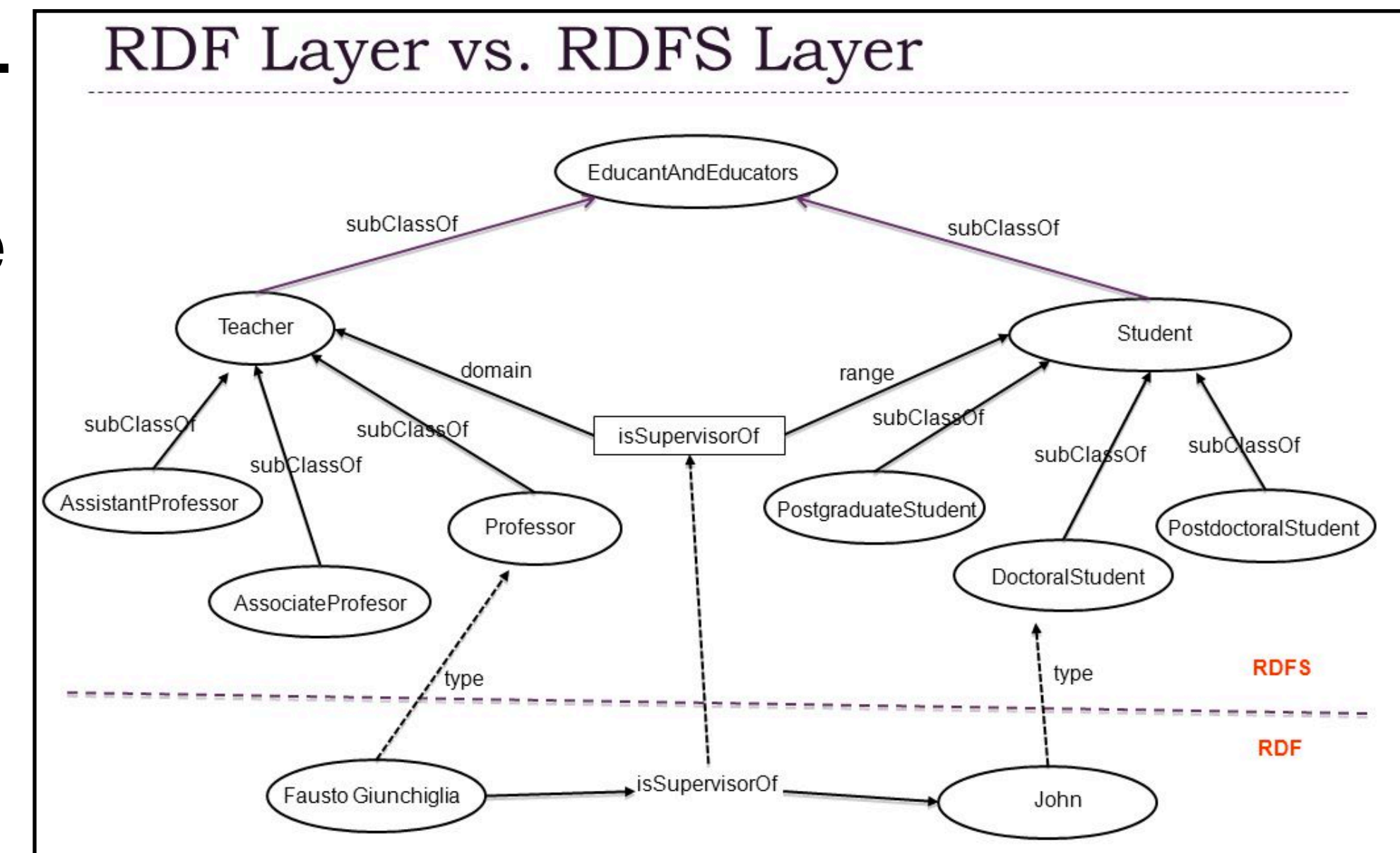


- Así como el modelado de la Web Semántica en RDF se basa de **grafos**, el modelado de la Web Semántica en RDF Schema Language (RDFS) se basa en **conjuntos**.

- Algunos aspectos de la **pertenencia** a conjuntos se pueden modelar en RDF, por ejemplo con la propiedad **rdf:type**.

- Pero RDF simplemente crea una estructura de grados para representar datos.

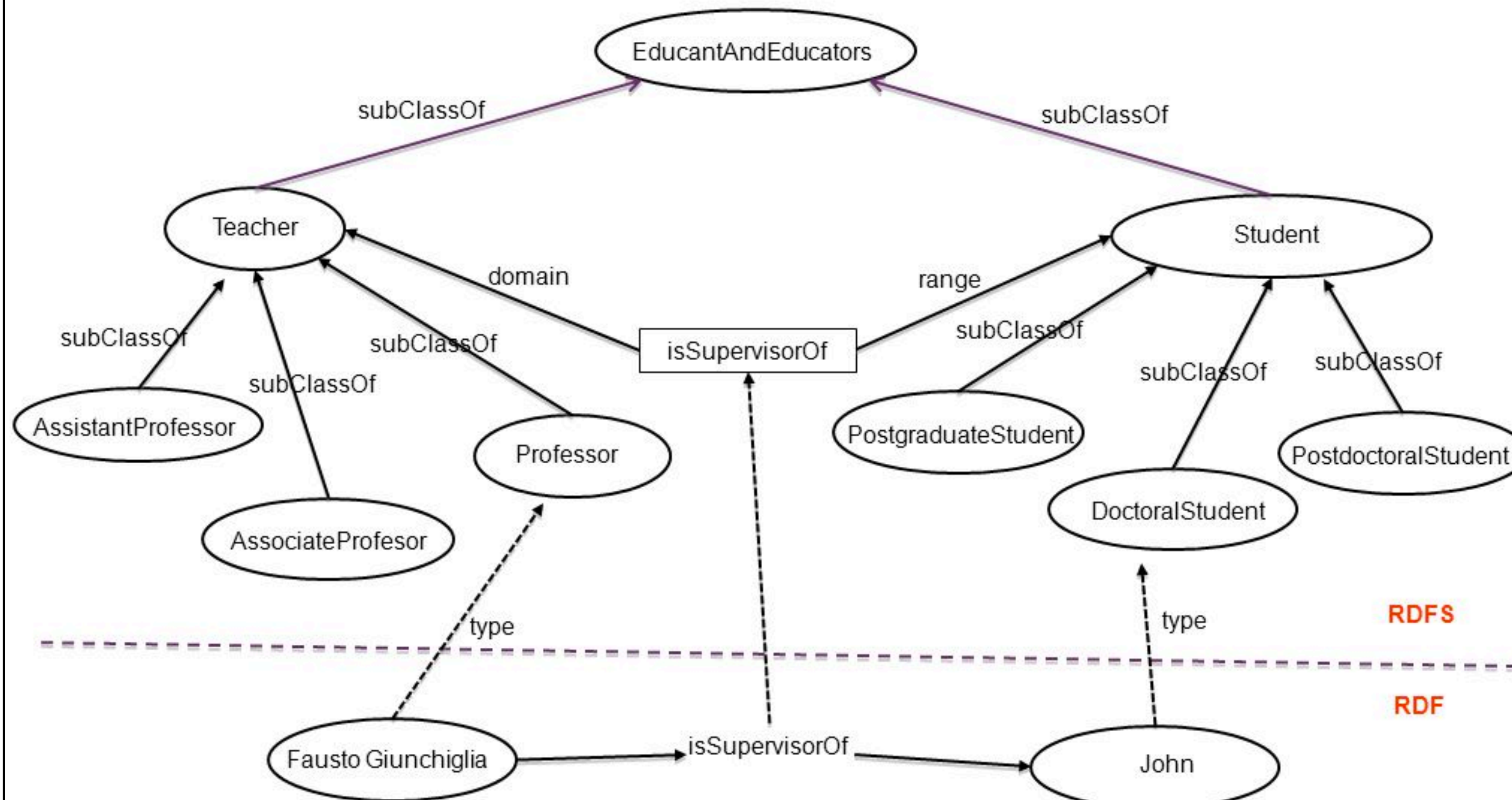
- RDFS proporciona algunas pautas sobre cómo usar esta estructura de grados de una manera disciplinada.



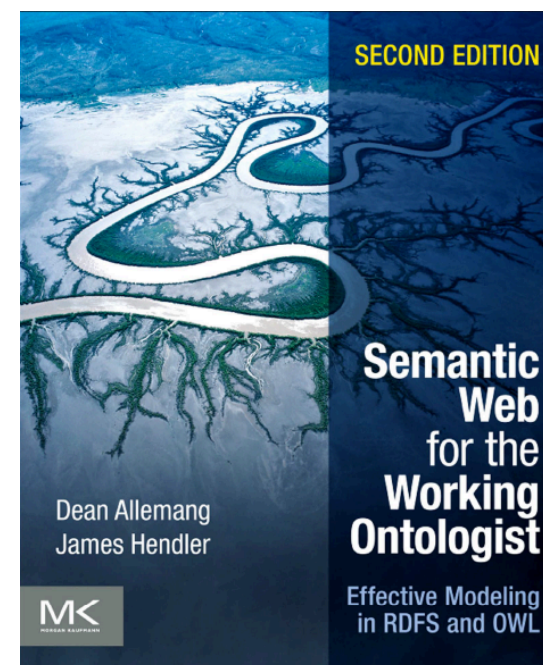
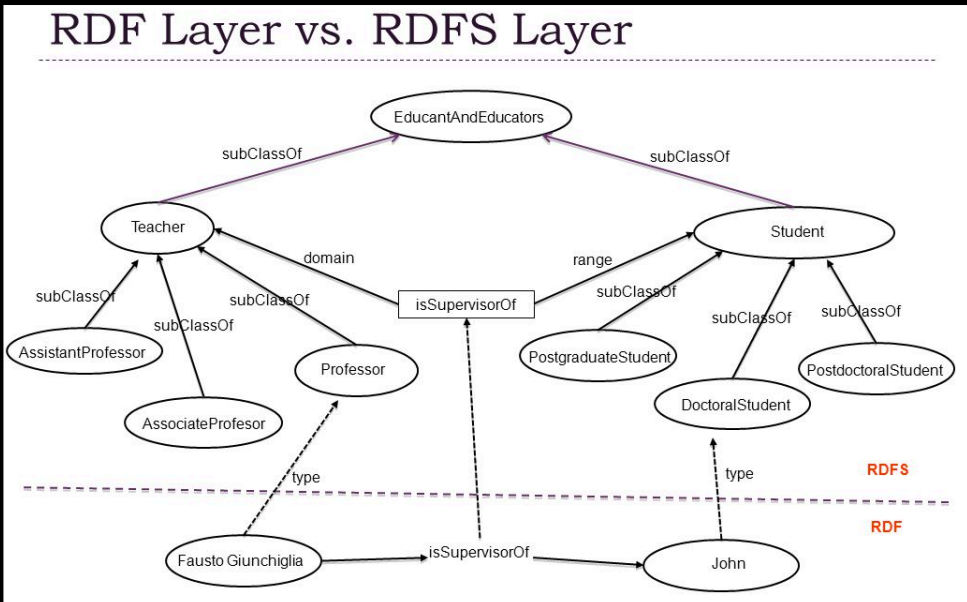


# Introducción

## RDF Layer vs. RDFS Layer



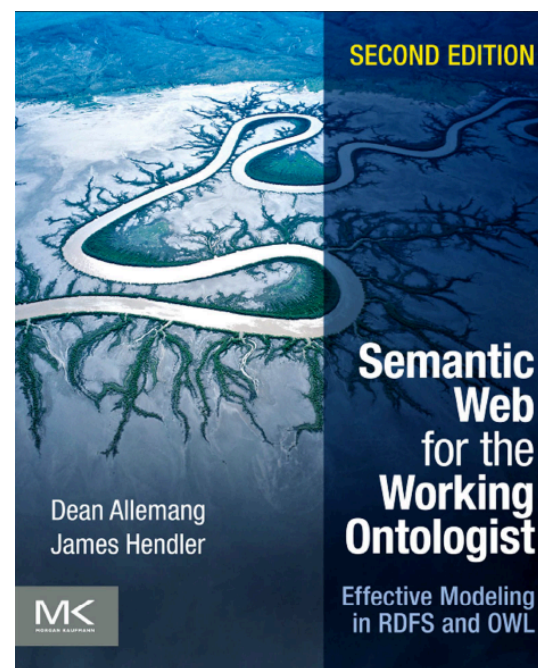




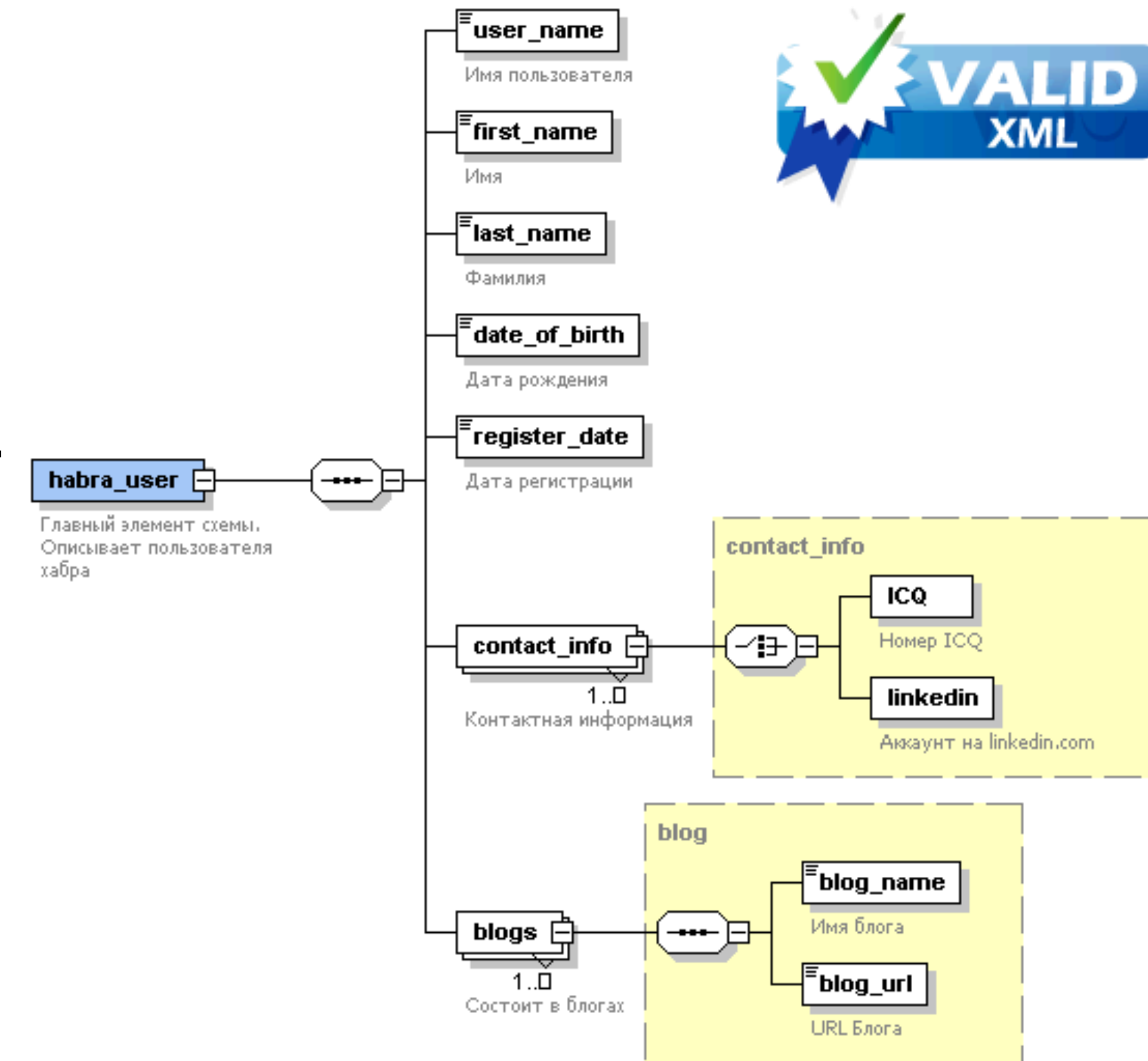
# Introducción

- RDFS proporciona una forma de hablar sobre el **vocabulario** que se utilizará en un grado RDF.
- ¿Qué individuos están relacionados entre sí, y cómo?
- ¿Cómo se relacionan las propiedades que utilizamos para definir a nuestros individuos con otros conjuntos de individuos?
- RDFS proporciona una forma para que un modelador de información exprese las respuestas a este tipo de preguntas.
- RDFS es como cualquier otro **lenguaje de esquemas**:
  - Provee información sobre la forma en que se **describirán los datos**.
  - Pero tienes diferencias substanciales frente a otros lenguajes de esquemas.

# Lenguaje de Esquema



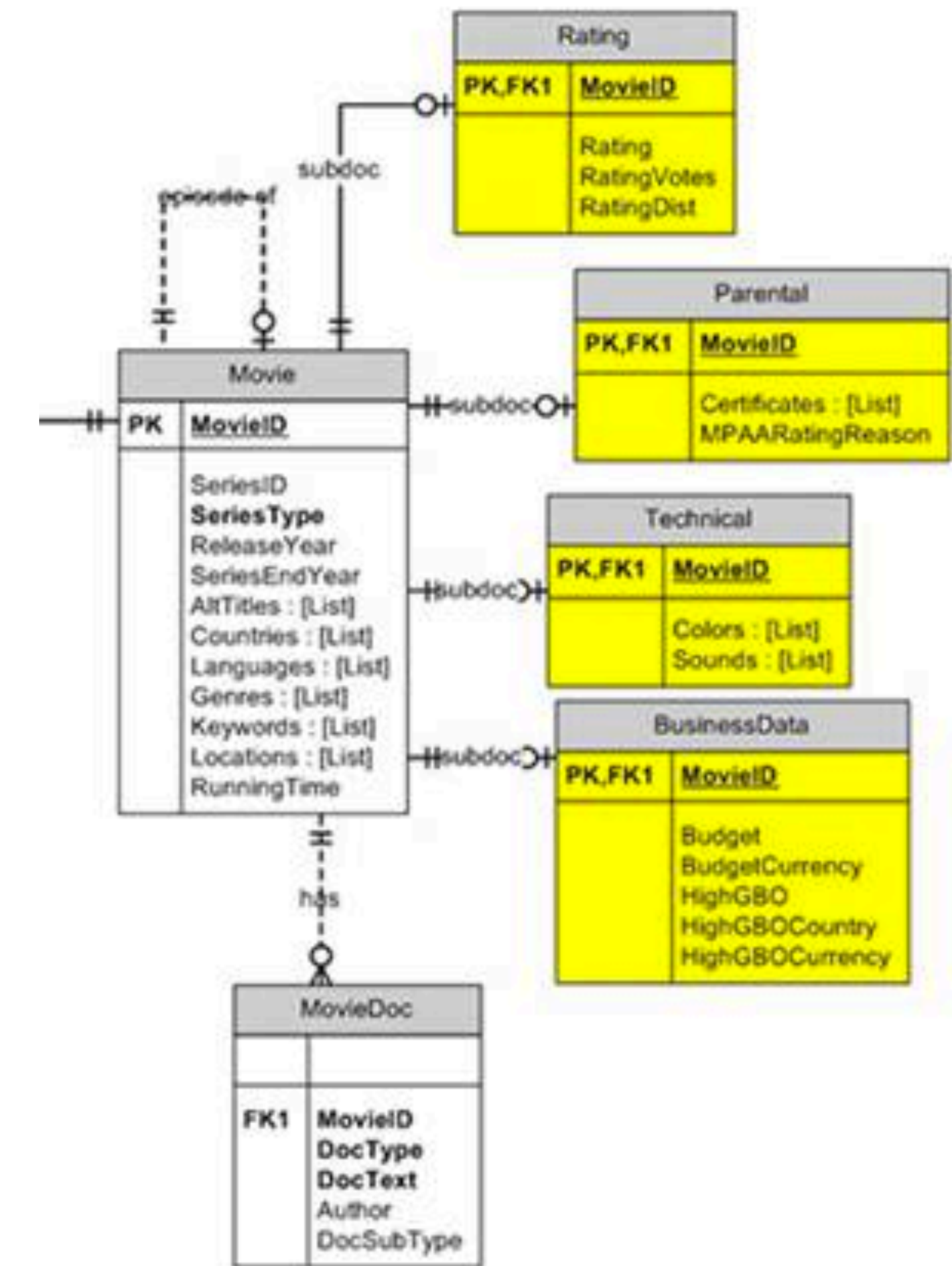
- RDFS es el lenguaje de esquema para RDF. Pero, ¿qué es un **lenguaje de esquema** en primer lugar?
- Consideremos los sistemas de **modelado de documentos** como un ejemplo.
- Para dichos sistemas, un lenguaje de esquema le permite a uno expresar el **conjunto de formatos permitidos** para un documento.
- Para un esquema dado, es posible determinar (a menudo automáticamente) si un **documento** en particular **se ajusta a ese esquema**.
- Esta es la principal capacidad provista por las definiciones de esquemas XML. Los analizadores XML pueden determinar automáticamente si un documento XML particular se ajusta a un esquema dado.





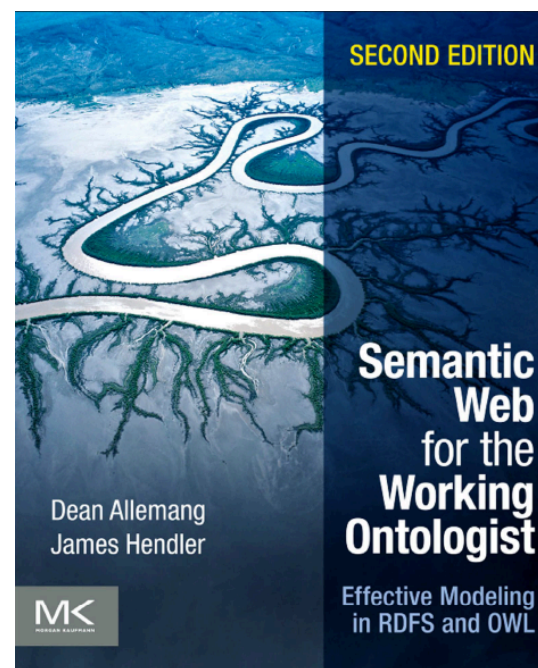
# Lenguaje de Esquema

- Otros lenguajes de esquema ayudan a **interpretar datos** particulares.
- Por ejemplo, un **esquema de base de datos** proporciona información clave para las tablas en una base de datos relacional.
- No hay nada en la tabla que indique el significado de la información en una columna en particular ni nada que indique qué columna se va a usar como índice para la tabla.
- Esta información se incluye en el esquema de la base de datos, pues no cambia de registro a registro.

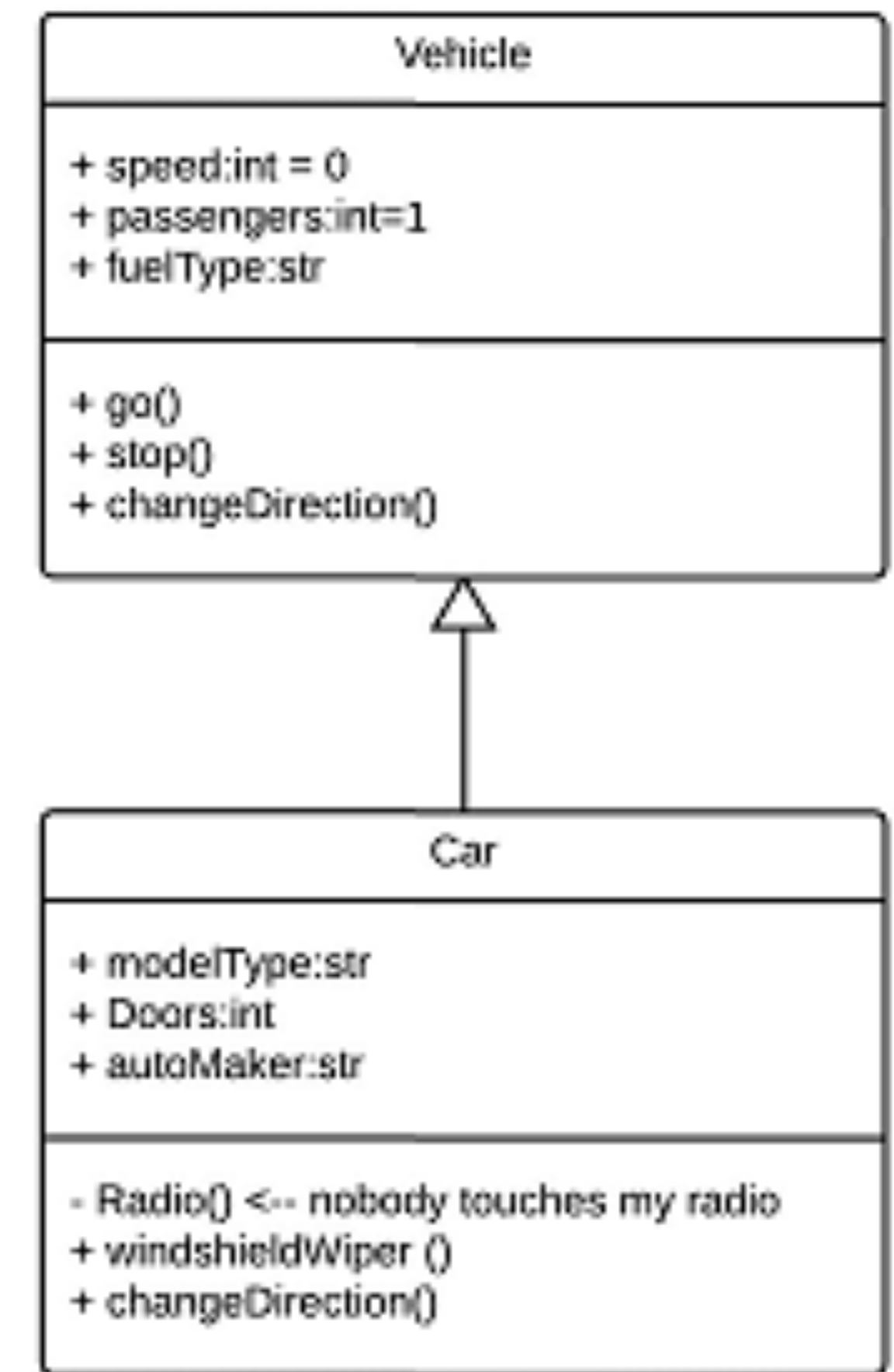




# Lenguaje de Esquema

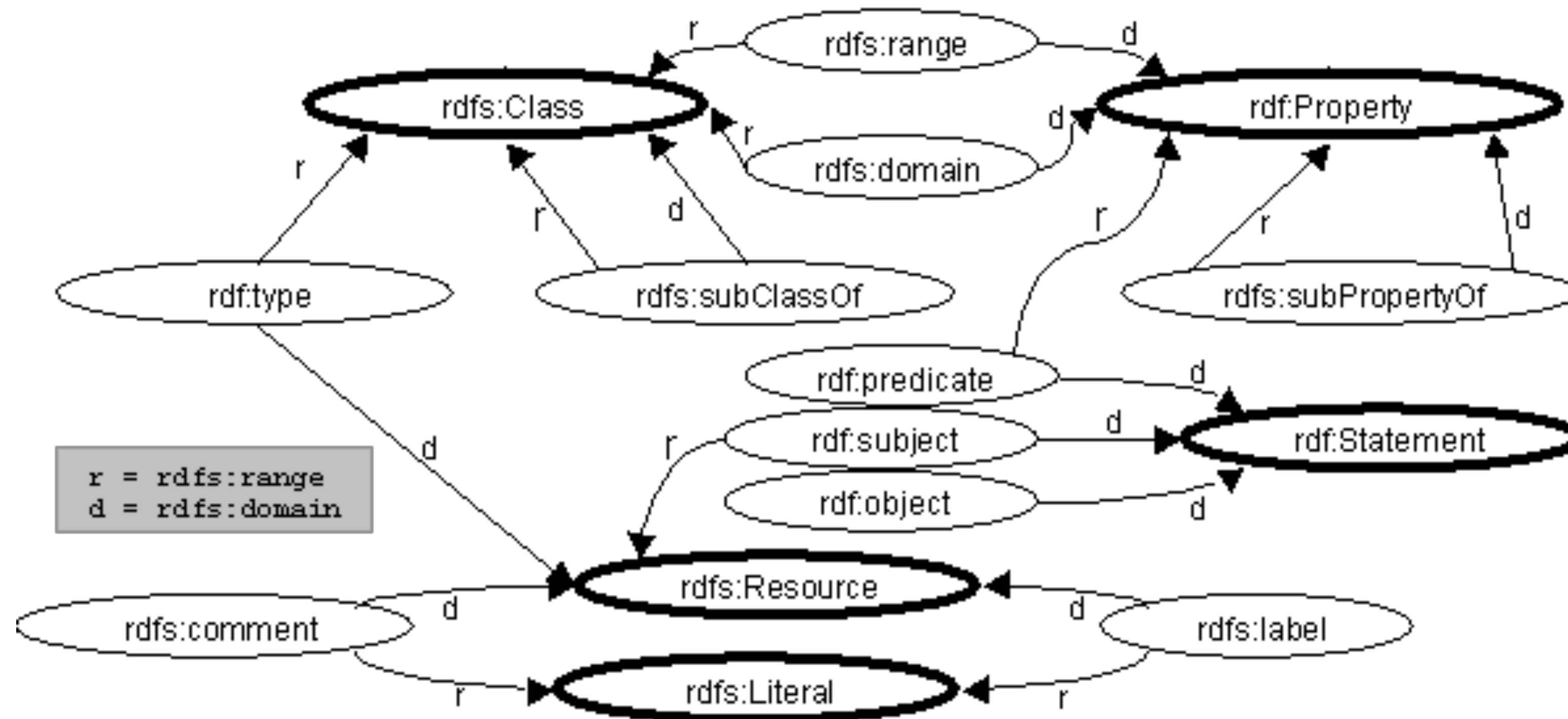


- En el paradigma orientado a objetos, la **estructura de clases** juega un papel organizador para la información también.
  - Pero en este caso, el diagrama de clases hace más que describir datos.
  - Determina, de acuerdo con la política de herencia del lenguaje en particular, qué **métodos están disponibles** para una instancia particular y cómo se implementan.
  - Esto está en marcado contraste con las bases de datos relacionales y el XML, en el sentido de que no interpreta información, sino que proporciona una manera sistemática para que alguien describa la información y las transformaciones disponibles para esa información.



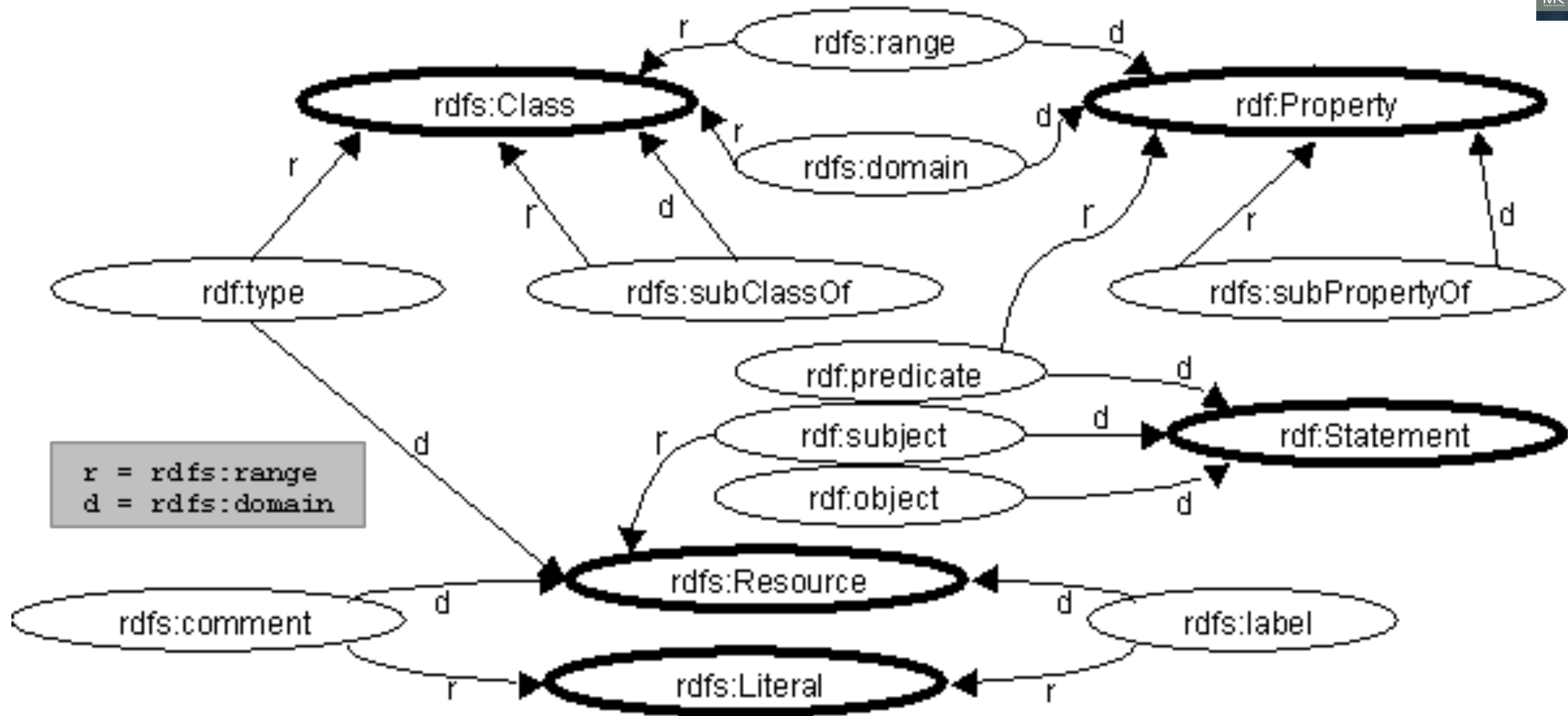
# Lenguaje de Esquema

- La idea clave del esquema en RDF es que ayudas a dar **significado** a los datos.
- Esto se logra especificando la semántica utilizando patrones de **inferencia**.

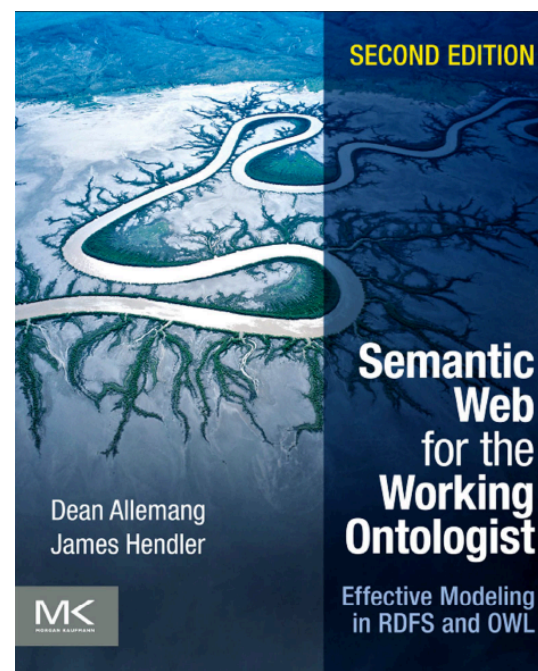




# Lenguaje de Esquema



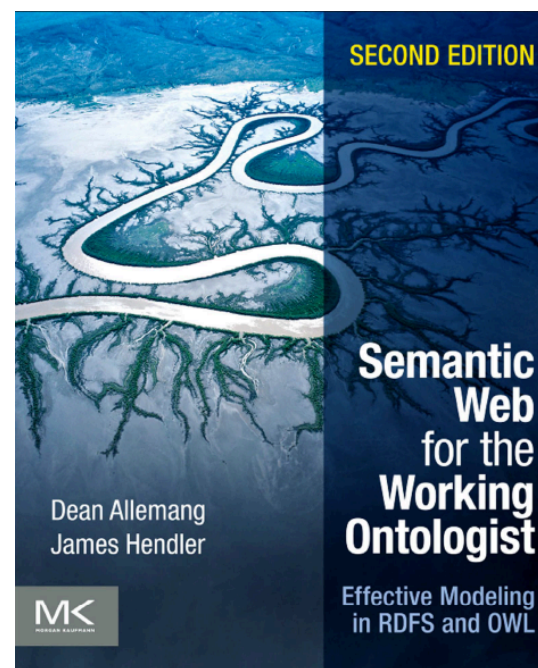
# Relación entre esquema y datos



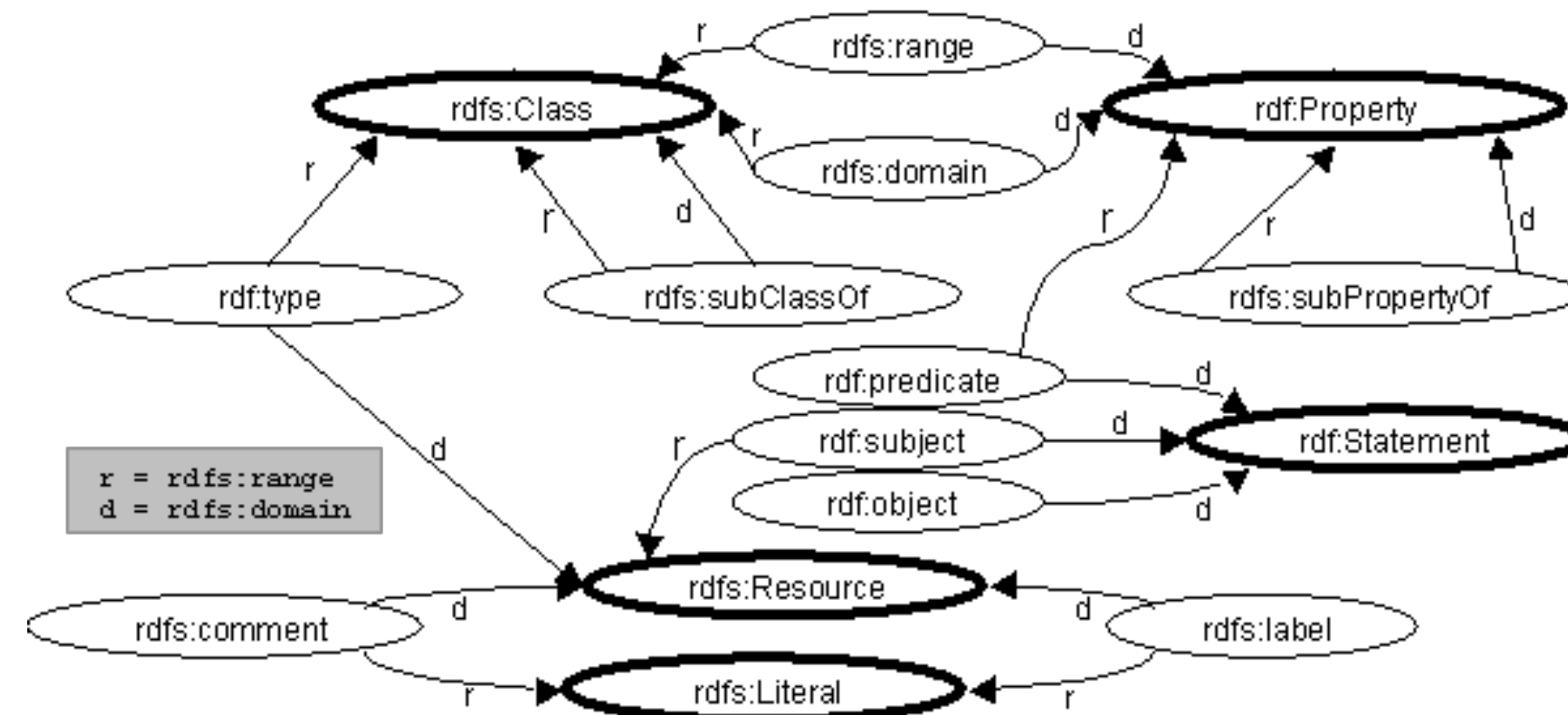
- En la mayoría de los sistemas de modelado, existe una clara **división entre los datos y su esquema**.
  - El esquema para una base de datos relacional normalmente no se expresa en una tabla en la base de datos.
  - El modelo de objetos de un sistema orientado a objetos no se expresa como objetos.
  - Un DTD XML no es un documento XML válido.
- Pero en muchos casos, las versiones modernas de tales sistemas **modelan el esquema de la misma forma que los datos**.
  - El protocolo de meta-objetos de Common Lisp y la API de introspección de Java representan los modelos de objeto como objetos en sí mismos.
  - La definición de hoja de estilo XML define estilos en un lenguaje XML.



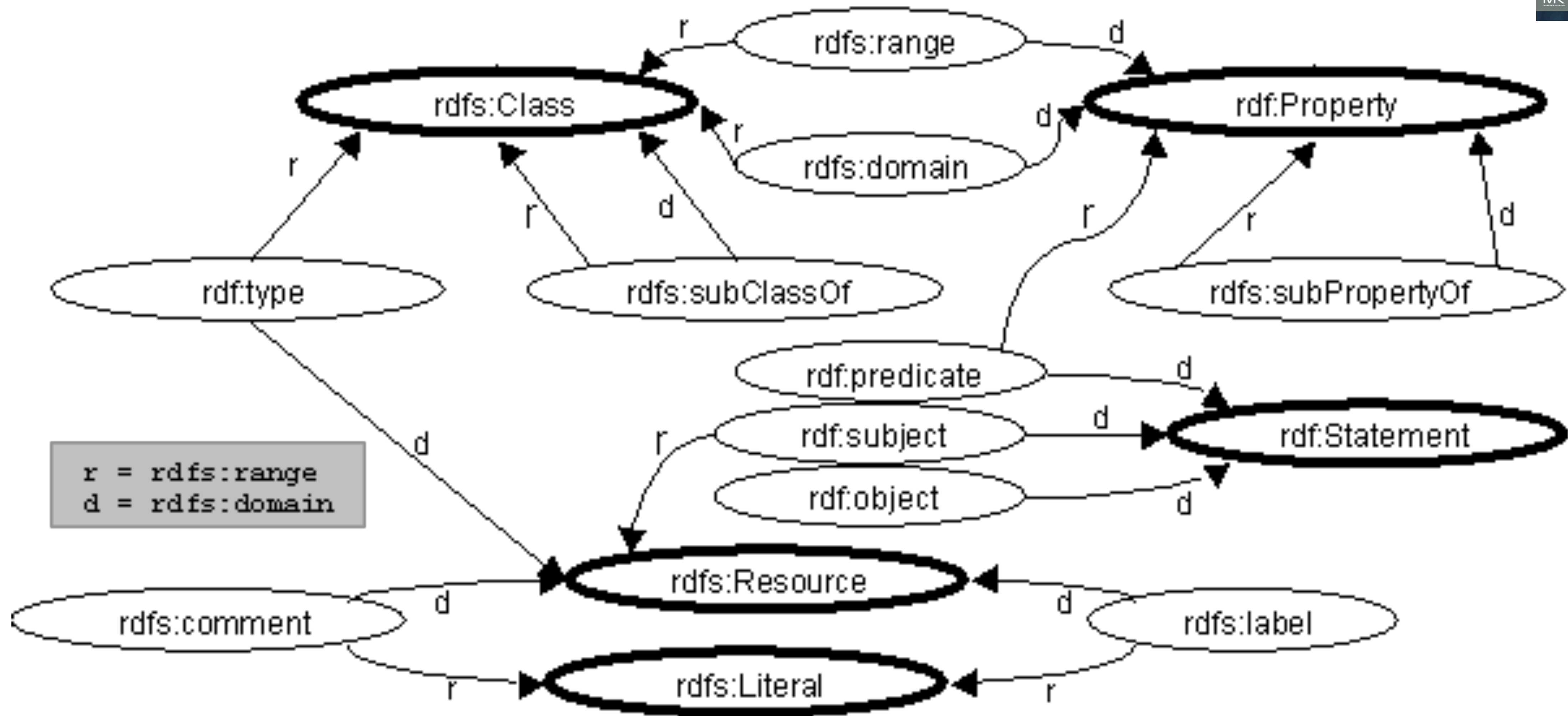
# Relación entre esquema y datos



- En el caso de RDF, el **lenguaje de esquema se define en RDF**.
- Toda la información de esquemas en RDFS se define con triplas RDF.
- La relación entre los recursos en RDF y los recursos del esquema se realiza con triplas, al igual que las relaciones entre cualquier otro recurso.

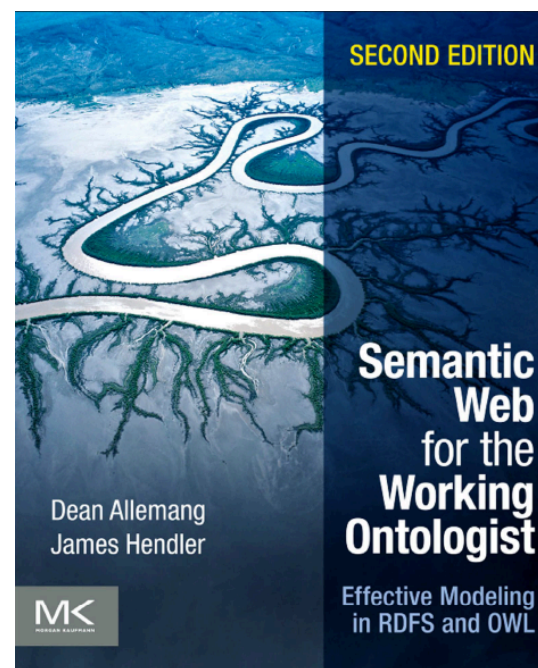


# Lenguaje de Esquema

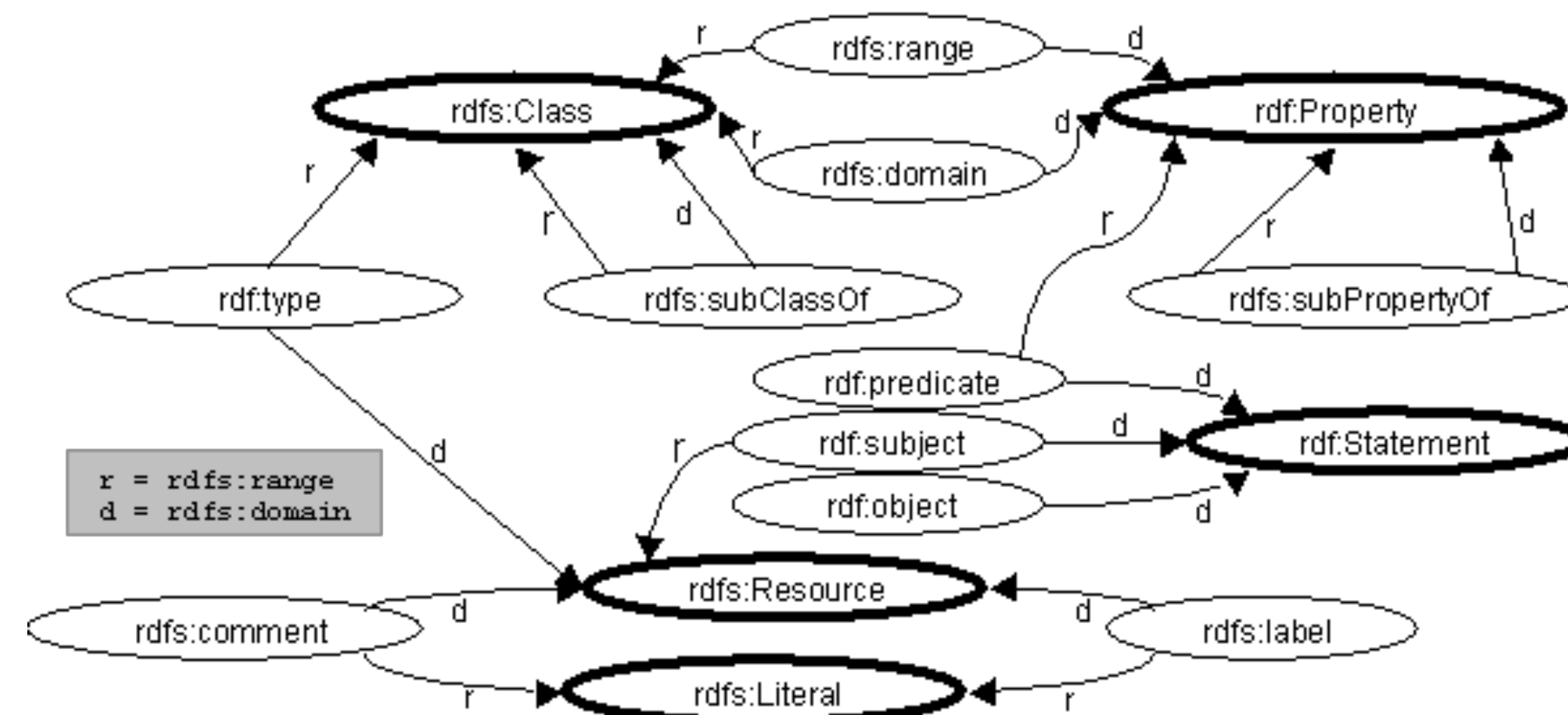




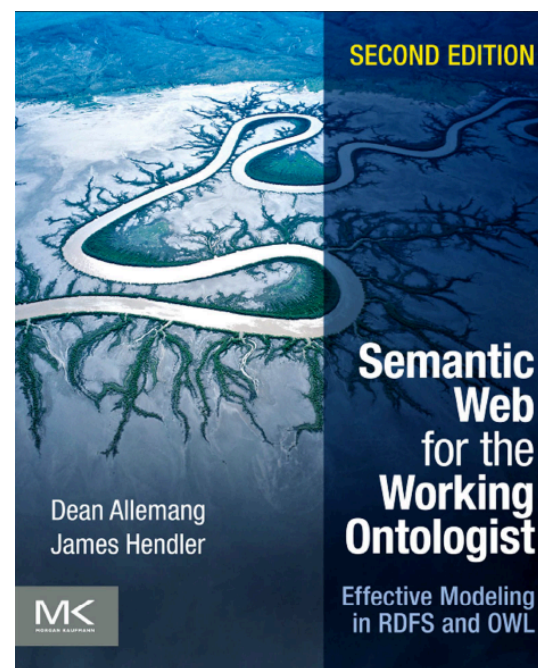
# Relación entre esquema y datos



- En **RDF**, todo es expresado en tripletas.
- El **resultado de las afirmaciones** sobre triples es expresado en nuevas triples (inferencia).
- Las estructuras que dirigen estas **inferencias**, son expresadas en triples.



# Relación entre esquema y datos

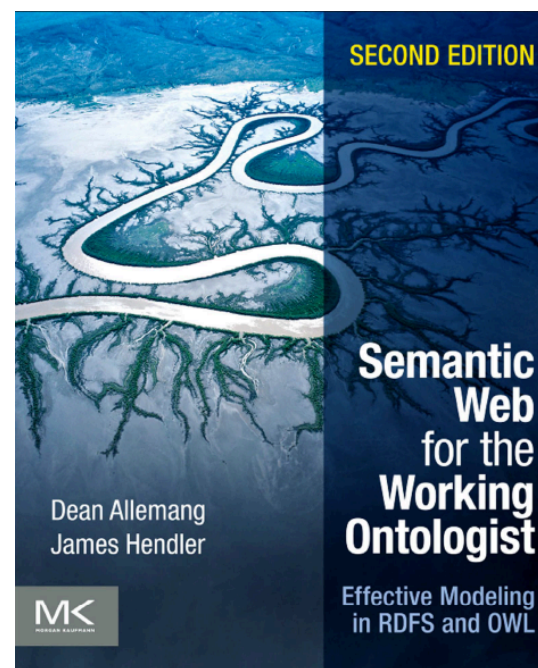


- Esta elegancia de diseño hace que sea particularmente fácil proporcionar una descripción formal de la semántica de RDFS, simplemente proporcionando reglas de inferencia que funcionen sobre patrones de triplas.
- La construcción básica para especificar un **conjunto** en RDFS se llama `rdfs:Class`.
- Como RDFS se expresa en RDF, la forma en que expresamos que algo es un conjunto es con una tripla, en particular, una tripla en el que el predicado es `rdf:type`, y el objeto es `rdfs:Class`.

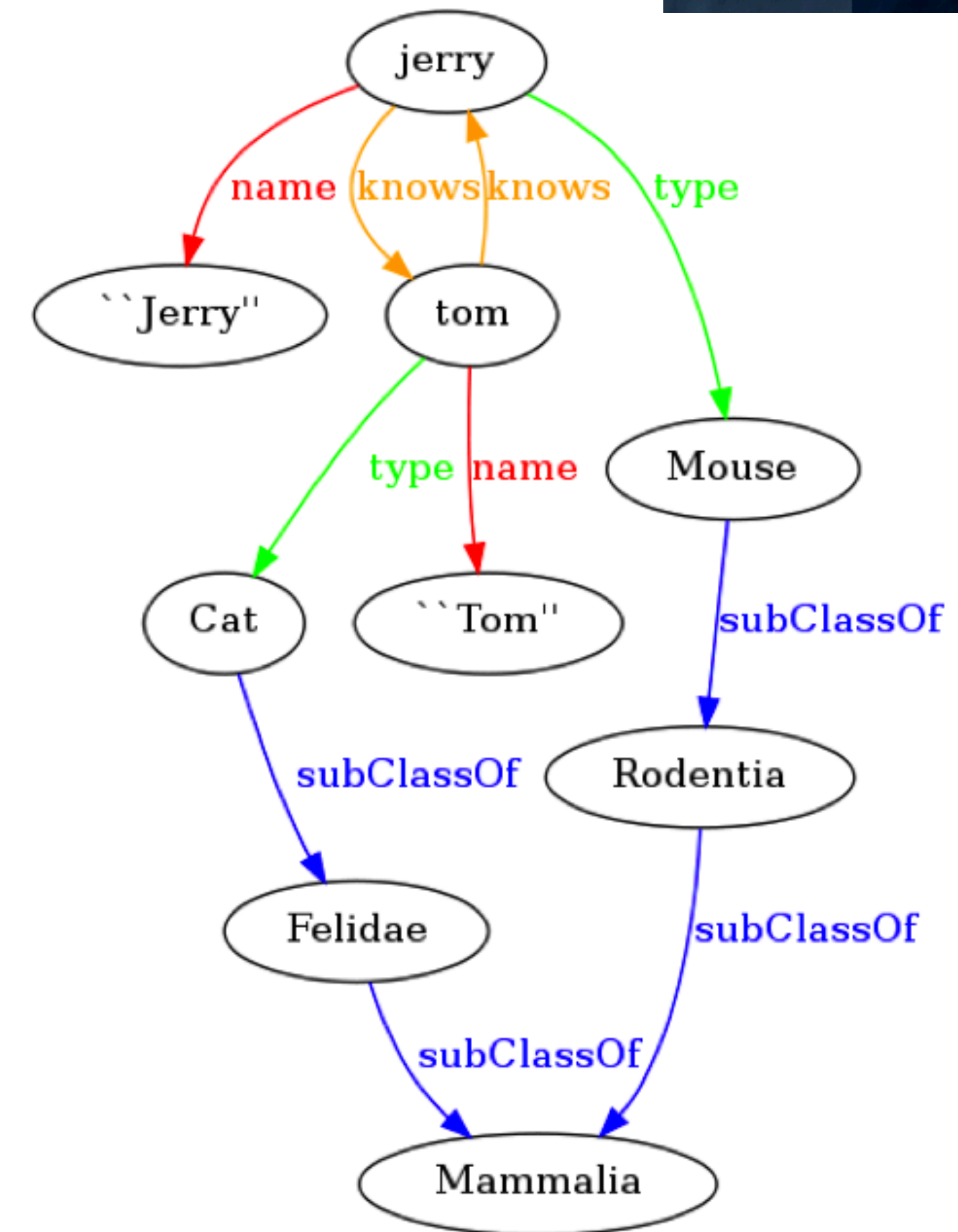
```
:AllStarPlayer rdf:type rdfs:Class.  
:MajorLeaguePlayer rdf:type rdfs:Class.  
:Surgeon rdf:type rdfs:Class.  
:Staff rdf:type rdfs:Class.  
:Physician rdf:type rdfs:Class.
```



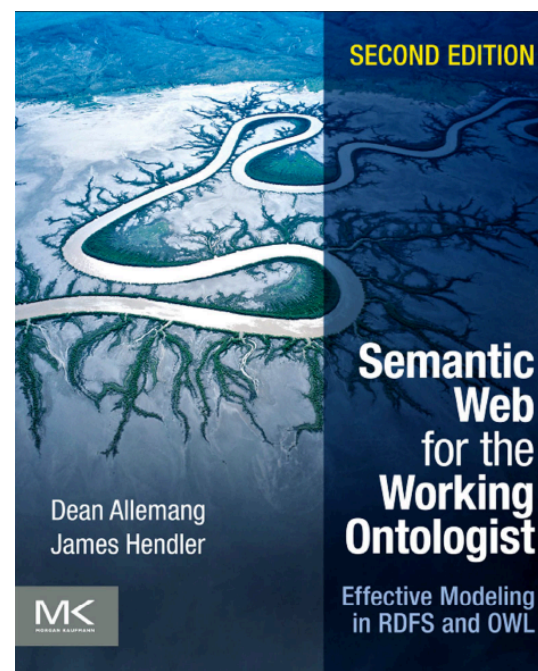
# Relación entre esquema y datos



- ¿Qué hay de nuevo en RDFS?
- En RDF se usa `rdf:type` para especificar que algo era miembro de un conjunto.
- ¿Qué se obtiene al especificar explícitamente que algo es un conjunto?
- Se obtiene una descripción del **significado** de la pertenencia a un conjunto.
- Se expresa el significado a través del mecanismo de **inferencia**.



# El lenguaje de esquema RDF

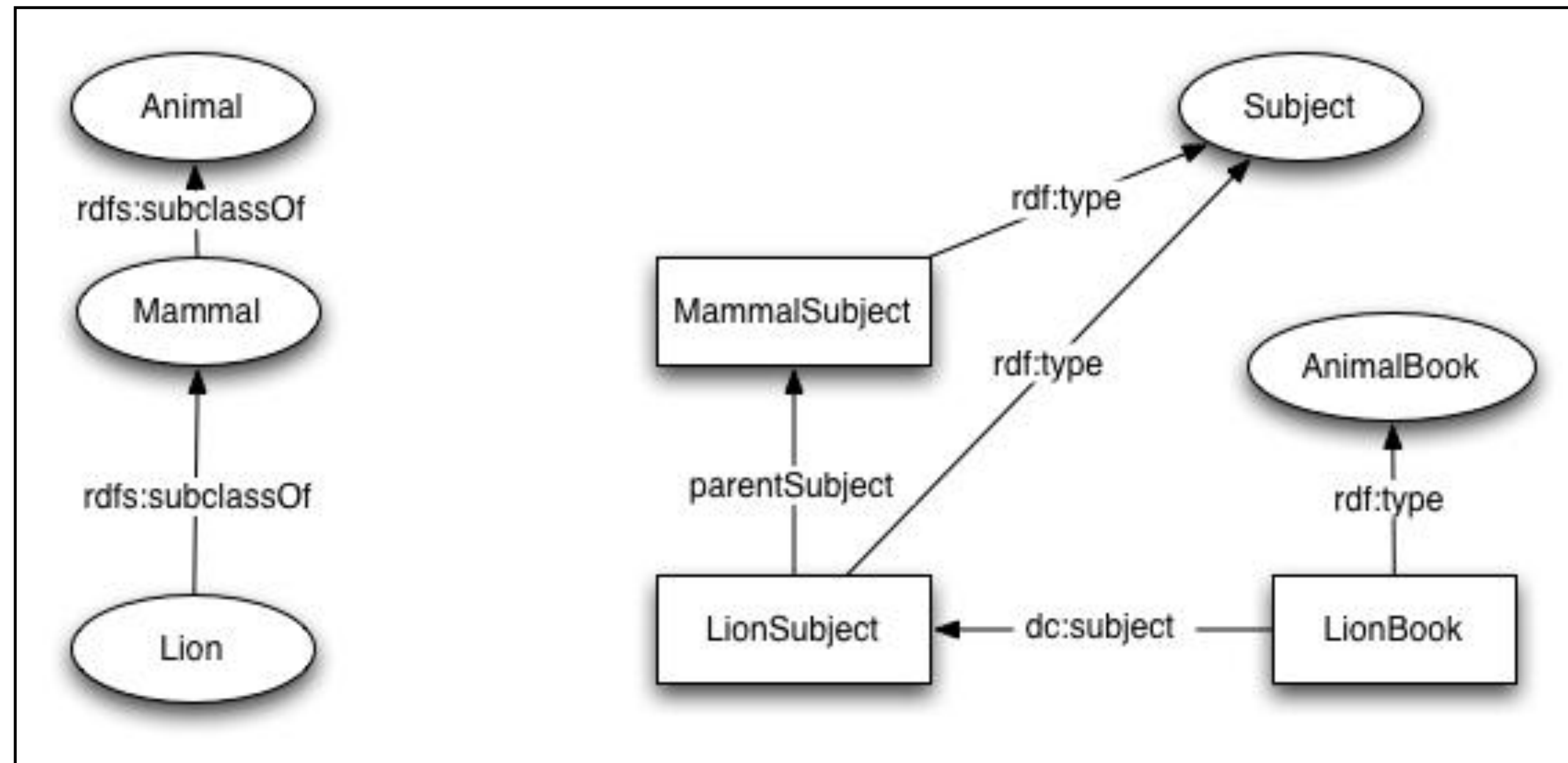


- **RDFS “extiende” RDF** al introducir un conjunto de recursos distinguibles en el lenguaje.
  - Esto es similar a la forma en que se puede extender un lenguaje de programación tradicional definiendo nuevas palabras clave definidas por el lenguaje.
  - Pero con una diferencia importante:
    - En RDF existe la capacidad de **usar cualquier recurso en cualquier tripla** (Cualquiera puede decir cualquier cosa sobre cualquier tema).
    - Al identificar ciertos recursos específicos como “palabras clave nuevas”, **no se ha ampliado el lenguaje**, simplemente se ha identificado ciertas triplas que tienen un significado especial estandarizado.
    - ¿Cómo podemos definir el “significado” de un recurso? En RDFS, el significado se expresa al especificar las **inferencias** que se pueden extraer cuando el recurso se usa de cierta manera.

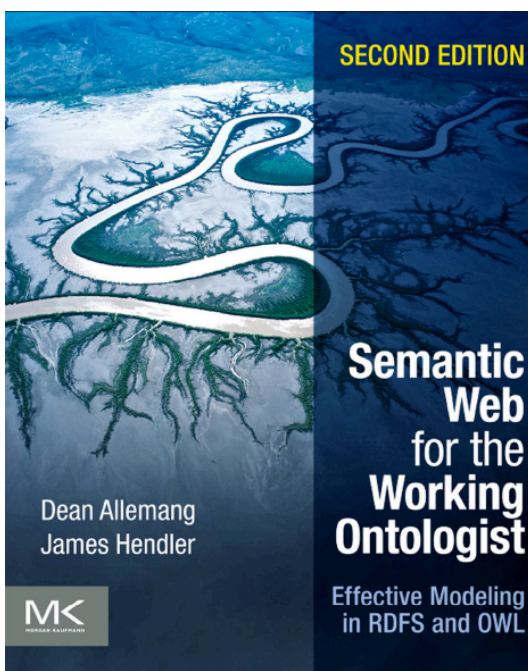


# Inferencia en RDF

## subclassOf



# Inferencia en RDF



- Supongamos que accede a la página Web de un minorista de ropa en línea y busca **“Gamuza Henley”** en la categoría de "Camisas".
- Su búsqueda aparece vacía.
- Te sorprende, porque estabas bastante seguro de que viste una **Gamuza Henley** en el catálogo en papel que llegó a tu buzón de correo.
- Así que busca el número de unidad en el catálogo y hace otra búsqueda usando este número.
- Efectivamente, está la **Gamuza Henley**. Además, encuentra que **“Henleys”** se muestra en el catálogo como una especie de **“Camisas”**.
- Usted murmura para sí mismo. “Si aparece bajo '**Henleys**', debería aparecer debajo de **“Camisas”**. ¿Cuál es el problema?”

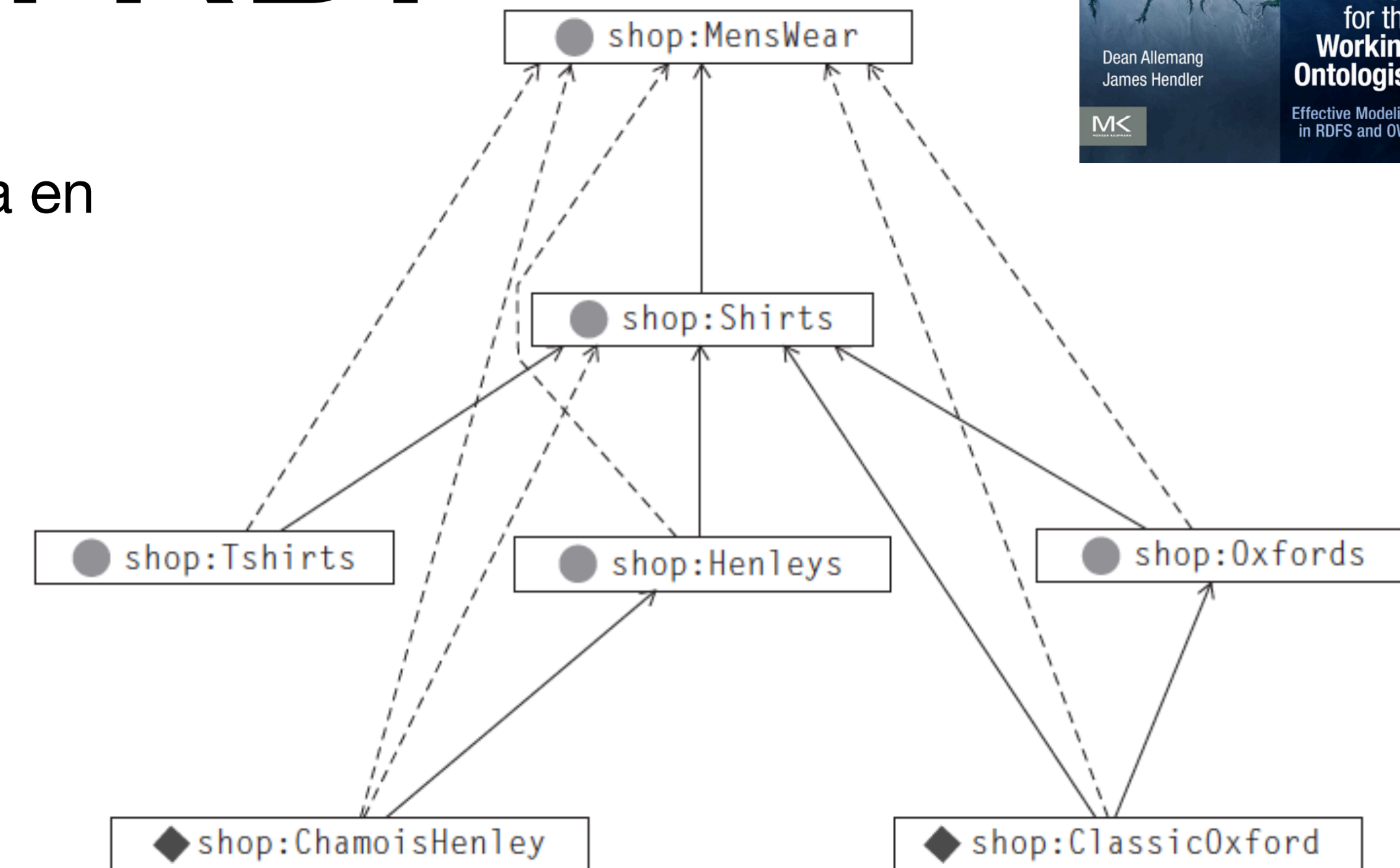


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.



# Inferencia en RDF

- ¿Qué esperamos de una búsqueda como esta?
- Queremos que cualquier búsqueda, consulta u otro acceso a los datos que hagan referencia a **“Camisas”** también miren en **“Henleys”**.
- ¿Qué tiene de especial la relación entre **“Camisas”** y **“Henleys”**?
- Cuando decimos: “Henleys es una especie de ‘Camisas’” esperamos que si buscamos en “Camisas” también busquemos en “Henleys”
- ¿Cómo podemos expresar este significado de una manera coherente y sostenible?

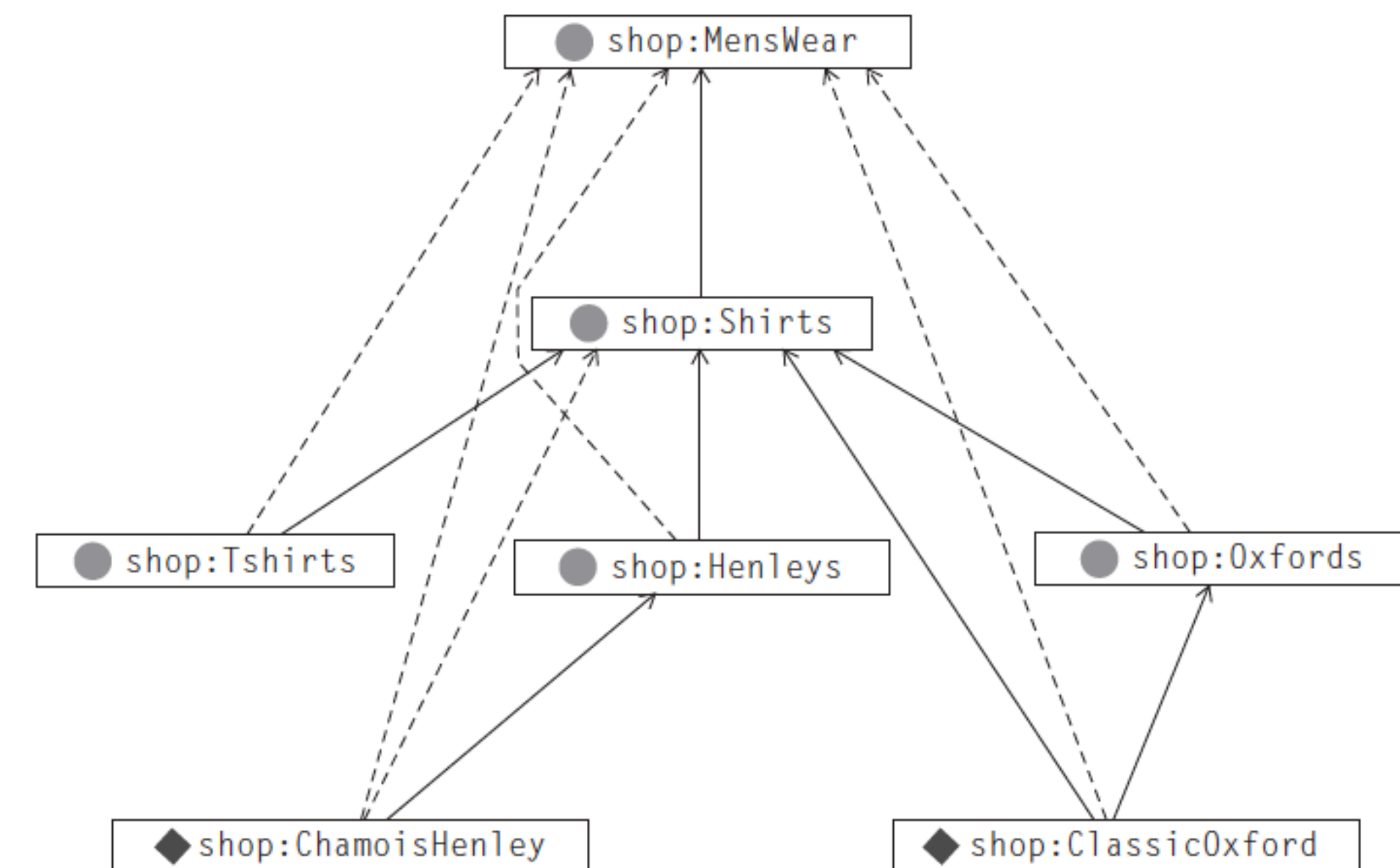


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.

# Inferencia en RDF

- Una solución a este problema es aprovechar el poder de la **consulta**.
- Después de todo, en las aplicaciones de bases de datos convencionales, es en la consulta donde se elaboran las relaciones entre los elementos de datos.

```
SELECT ?item
WHERE {?class :subClassOf* :Shirts .
      ?item a ?class . }
```

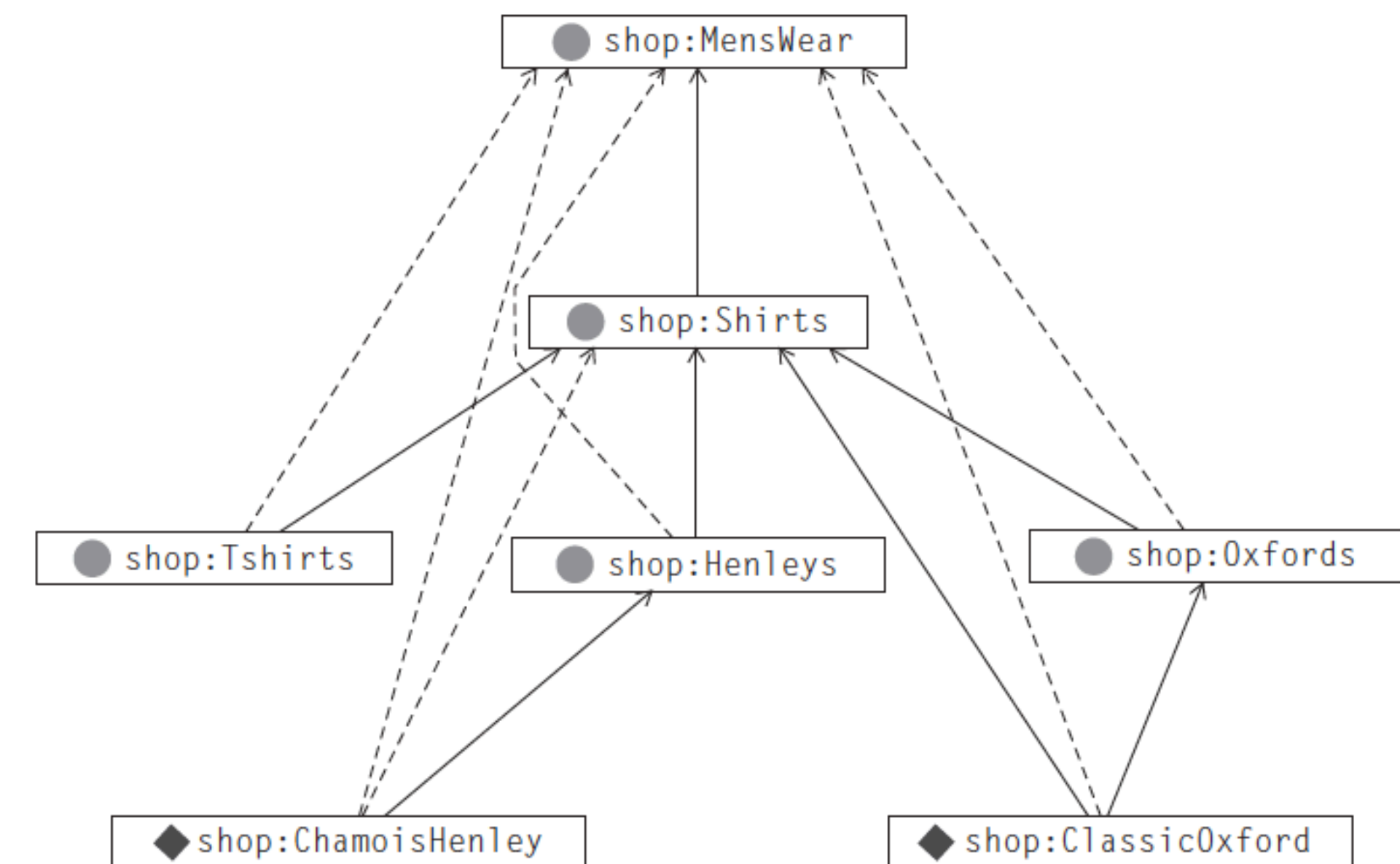


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.



# Inferencia en RDF

- Además del poder de la consulta, la Web Semántica también proporciona un **modelo de expresión de datos** que permite la **representación explícita de la relación** entre varios elementos de datos.
- En este sentido, realmente permite que un modelador de datos cree datos que estén más conectados, mejor integrados y en los que las restricciones de coherencia en los datos puedan expresarse en los datos mismos.
- Los datos pueden describir algo sobre la forma en que deberían usarse.

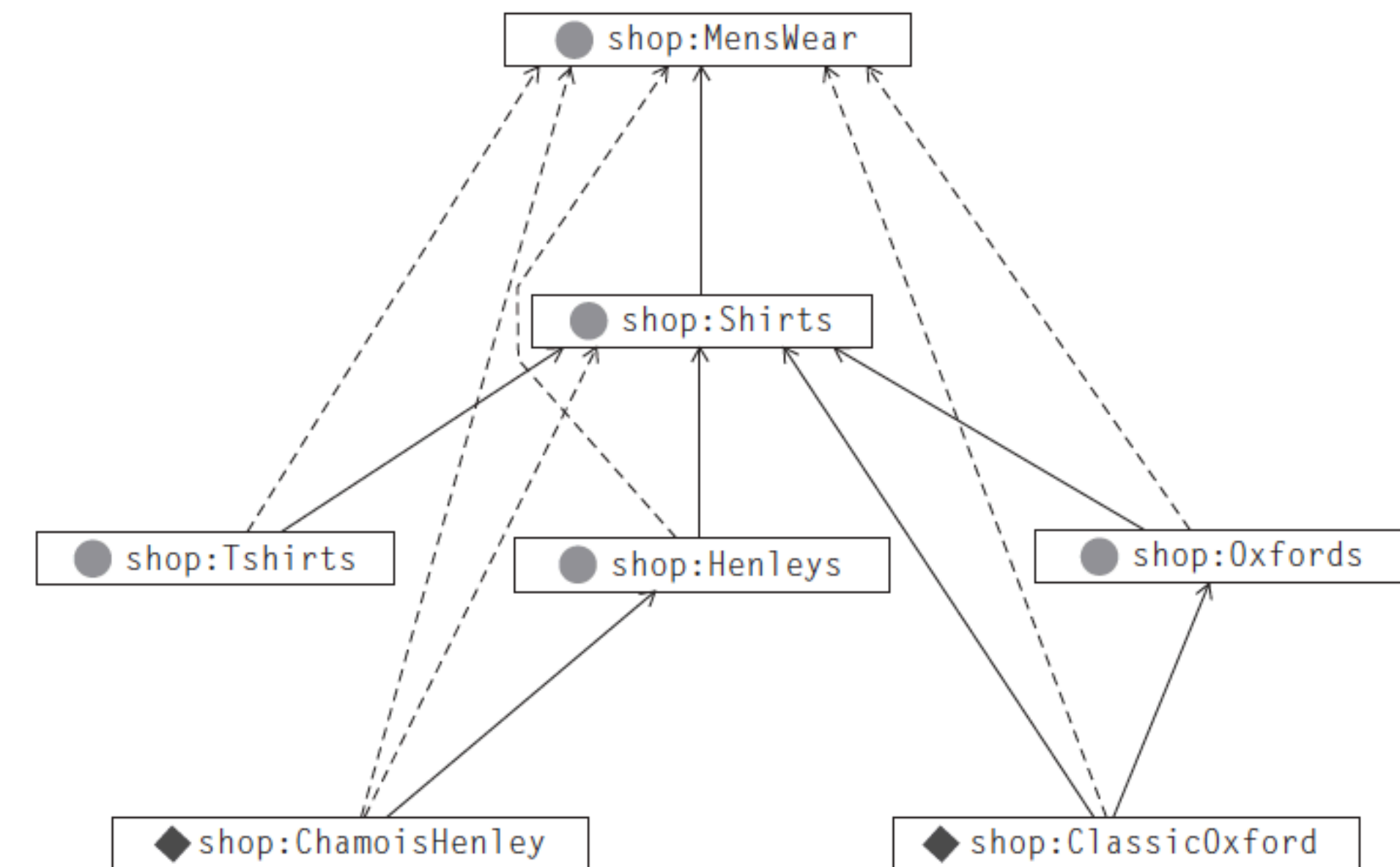


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.

# Inferencia en RDF

- La arquitectura de la Web Semántica incluye una serie de capas en la parte superior de la capa de RDF para **describir las restricciones** de consistencia en los datos.
- La clave de estos niveles es la noción de **inferencia**.
- En el contexto de la Web Semántica, la inferencia simplemente significa que, dada determinada información, podemos determinar otra información relacionada que también podemos considerar como si hubiera sido establecida.

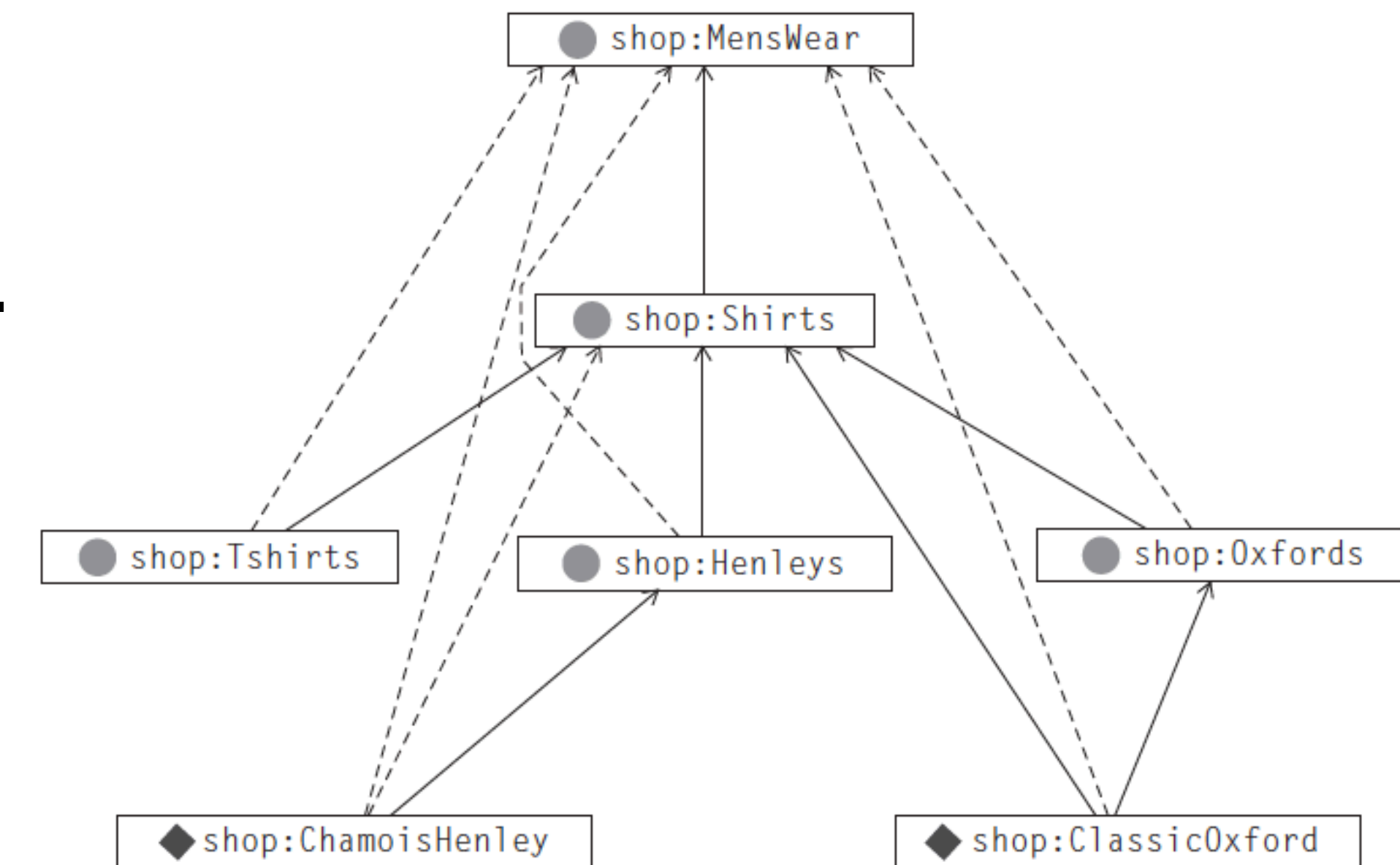


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.



# Inferencia en RDF

- En el ejemplo de Henleys / Camisas, inferiríamos que cualquier miembro de la clase “Henleys” también es miembro de la clase “Camisas”.
- La inferencia es un mecanismo poderoso para tratar la información, y puede abarcar una amplia gama de elaborados procesos.
- Con el fin de hacer que nuestros datos sean más integrados y consistentes, las inferencias muy simples a menudo son más útiles que las elaboradas.

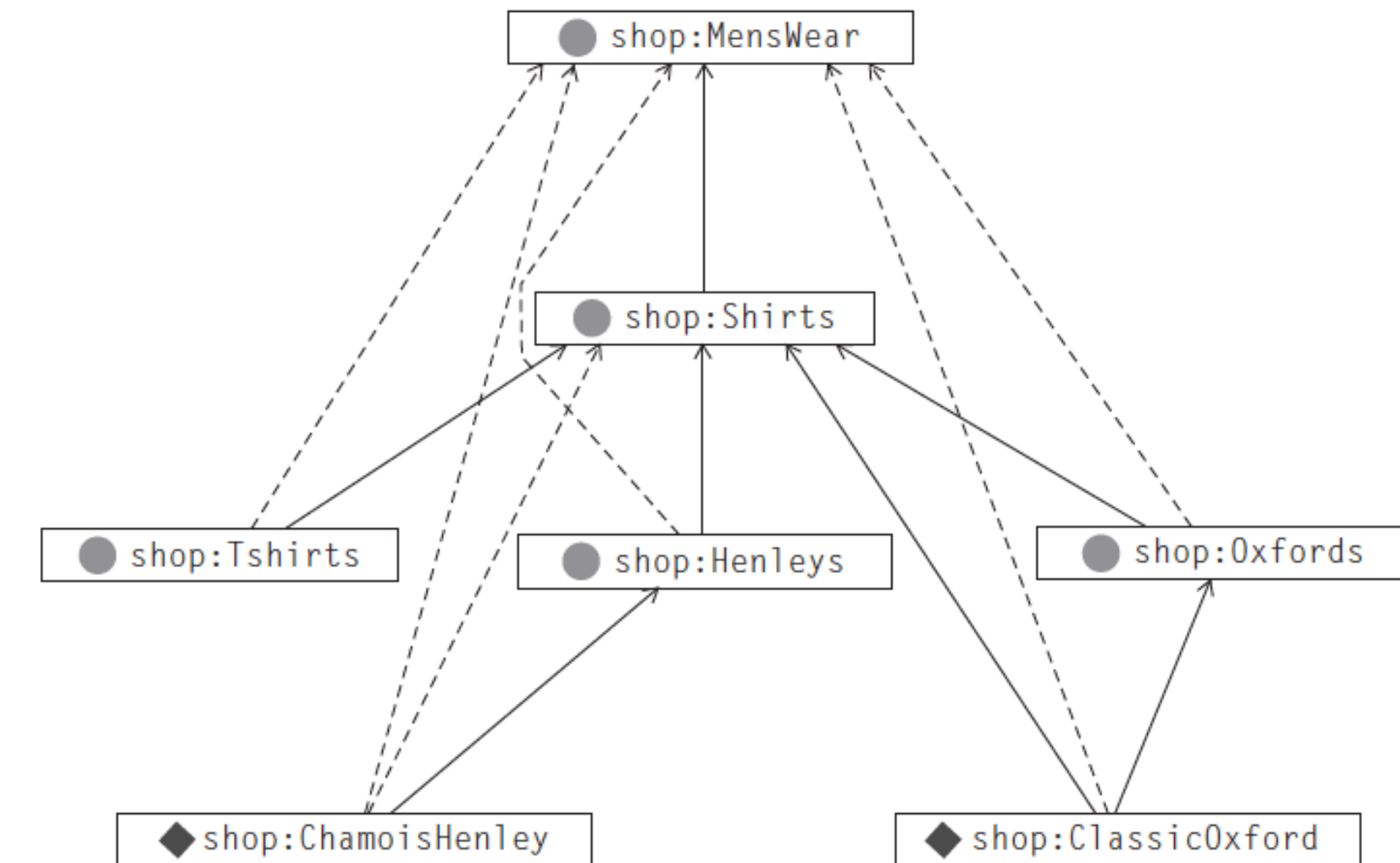
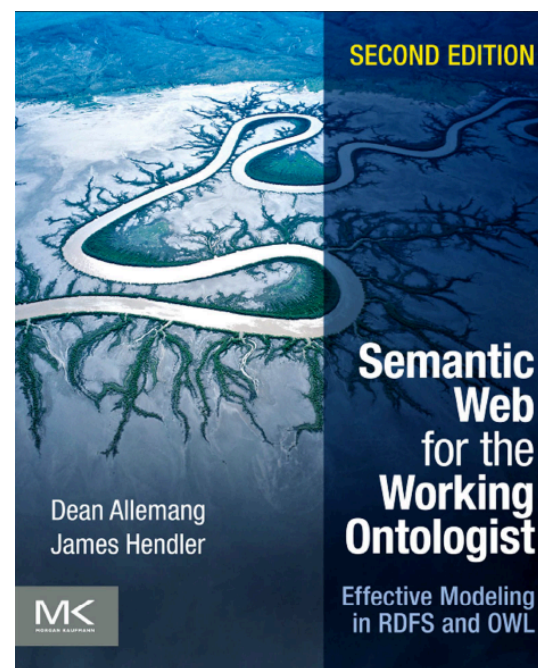


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.

# Inferencia en RDF



- Para hacer que nuestros datos parezcan más conectados y consistentemente integrados, debemos ser capaces de **agregar relaciones** en los datos.
- Estas relaciones restringirán la forma en que se ven los datos.
  - Queremos poder expresar la relación entre **“Henleys”** y **“Camisas”** que nos dirá que cualquier artículo en la categoría **“Henleys”** también debería estar en la categoría **“Camisas”**.
  - Queremos expresar el hecho que si una cadena de hoteles tiene un hotel en un lugar determinado, entonces en dicho lugar existen servicios de la cadena de hotel.
  - Queremos expresar la lista de planetas en términos de las clasificaciones de los diversos cuerpos en el sistema solar.

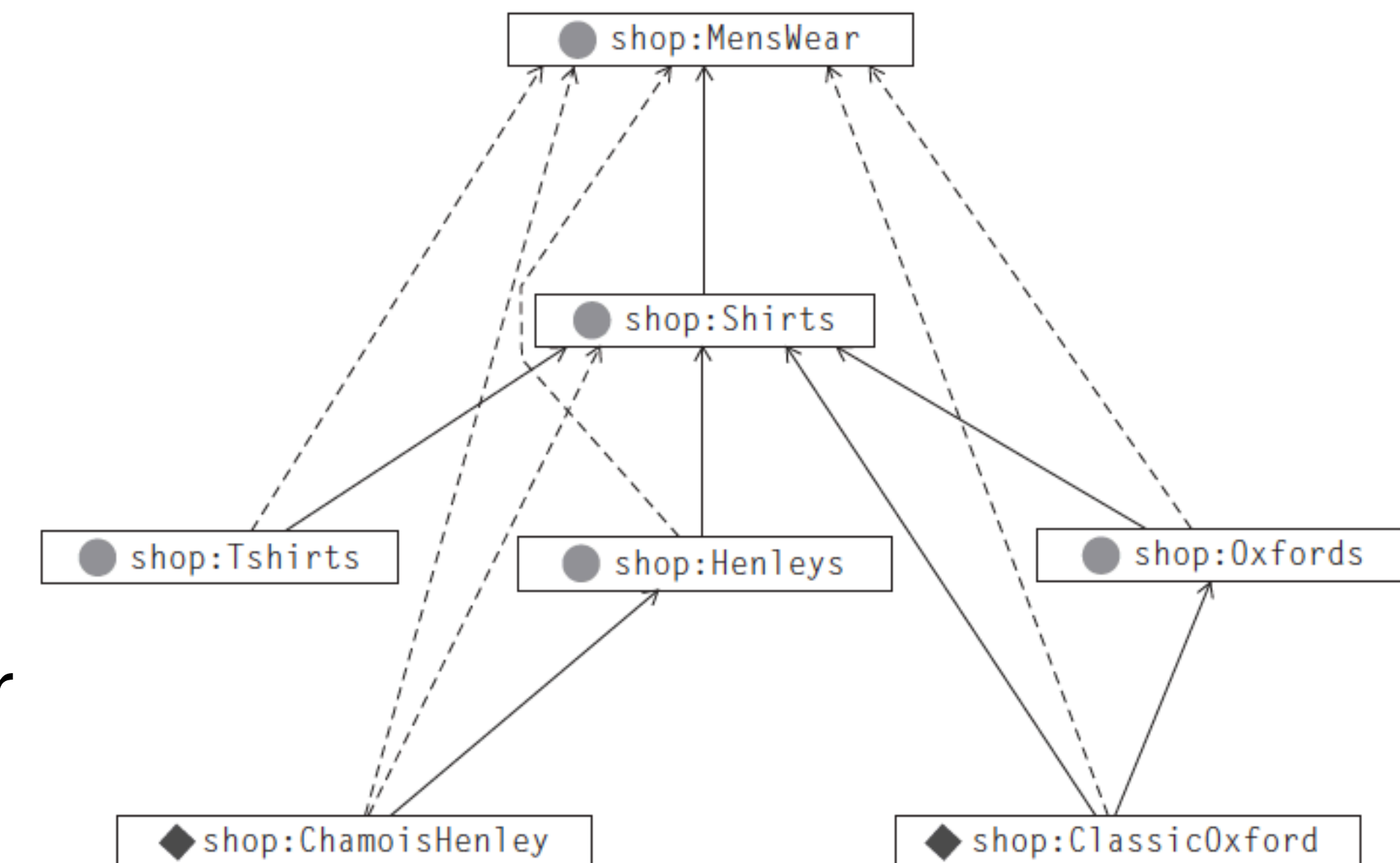


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.



# Inferencia en RDF

- Muchas de estas relaciones son familiares para los modeladores de información en muchos paradigmas.
- Para nuestro ejemplo la relación entre “Henleys” y “Camisas”.
  - Los escritores de tesauros están familiarizados con la noción de **término más amplio**. “Camisas” es un término más amplio que “Henleys”.
  - Los programadores orientados a objetos están acostumbrados a la noción de **subclases o extensiones de clase**. “Henleys” es una subclase de, o se extiende, la clase “Camisas”.
  - En el lenguaje de esquema RDF, es decir que, “Henleys” **subClassOf** “Camisas”.

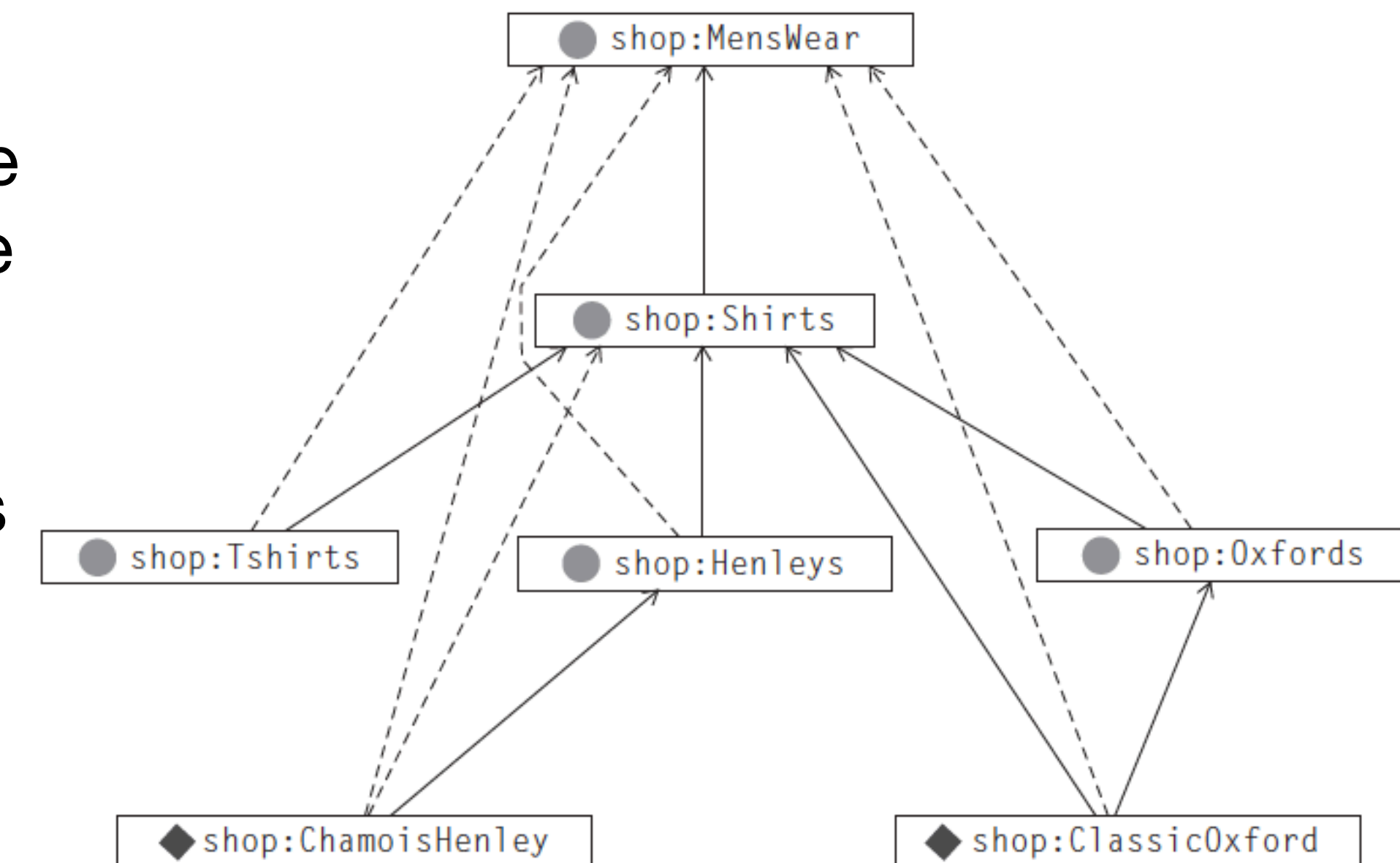
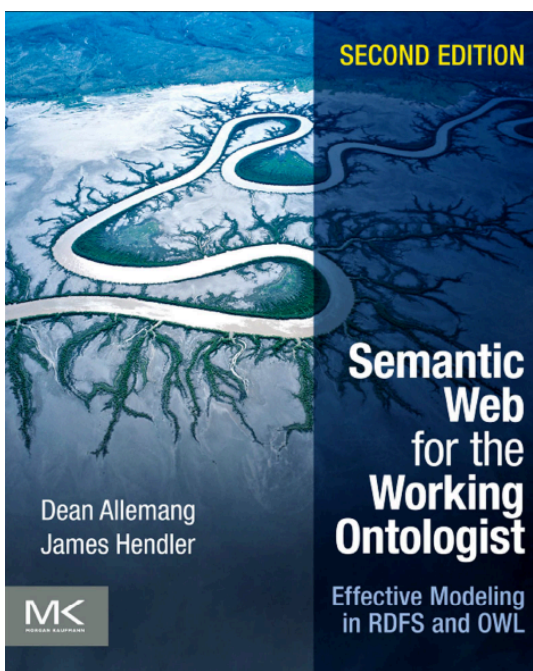


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.

# Inferencia en RDF

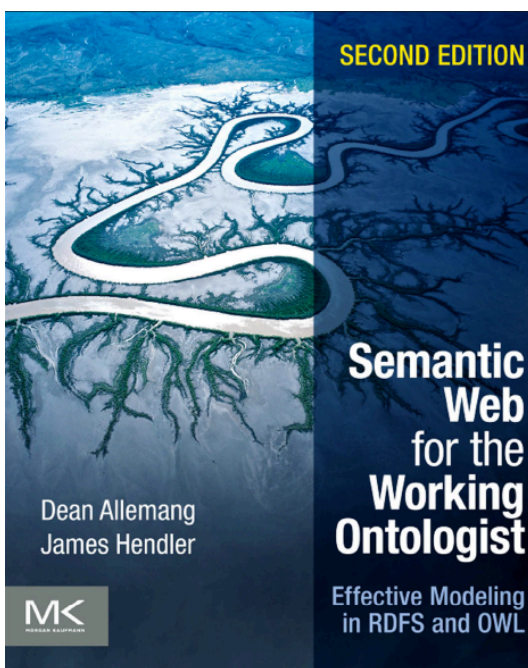


- La infraestructura de la Web Semántica proporciona una especificación formal y elegante del **significado** de varios términos como `subClassOf`.
- Por ejemplo, el significado de “B es una `SubClassOf` C” es :
  - **“Cada miembro de la clase B también es miembro de la clase C”.**
  - Esta especificación se expresa en la forma de una **inferencia**.
  - A partir de la información **“x es miembro de B”**, uno puede obtener la nueva información, **“x es miembro de C”**.

```
IF
?A rdfs:subClassOf ?B.
AND
?x rdf:type ?A.
THEN
?x rdf:type ?B.
```



# Inferencia en RDF



- En lenguaje simple, se dice que si una clase A es una subclase de otra clase B, cualquier cosa de tipo A también es de tipo B.
- Esta declaración simple es la definición completa del significado de **subClassOf** en el lenguaje de esquema RDF.
- En RDF se refiere a esta regla como la regla de propagación de tipo.
- Esta interpretación muy simple de la relación de subclase lo convierte en un caballo de batalla para el modelado RDFS.
- Se corresponde estrechamente con el constructo SI / ENTONCES de los lenguajes de programación:
  - Si algo es un miembro de la subclase, ENTONCES es un miembro de la superclase.

```
IF
?A rdfs:subClassOf ?B.
AND
?x rdf:type ?A.
THEN
?x rdf:type ?B.
```

# Inferencia en RDF

```
shop:Henleys rdfs:subClassOf shop:Shirts.  
shop:ChamoisHenley rdf:type shop:Henleys.
```

Ask:

```
SELECT ?item  
WHERE {?x rdf:type shop:Shirts . }
```

Answer:

?item

Shop:ChamoisHenley

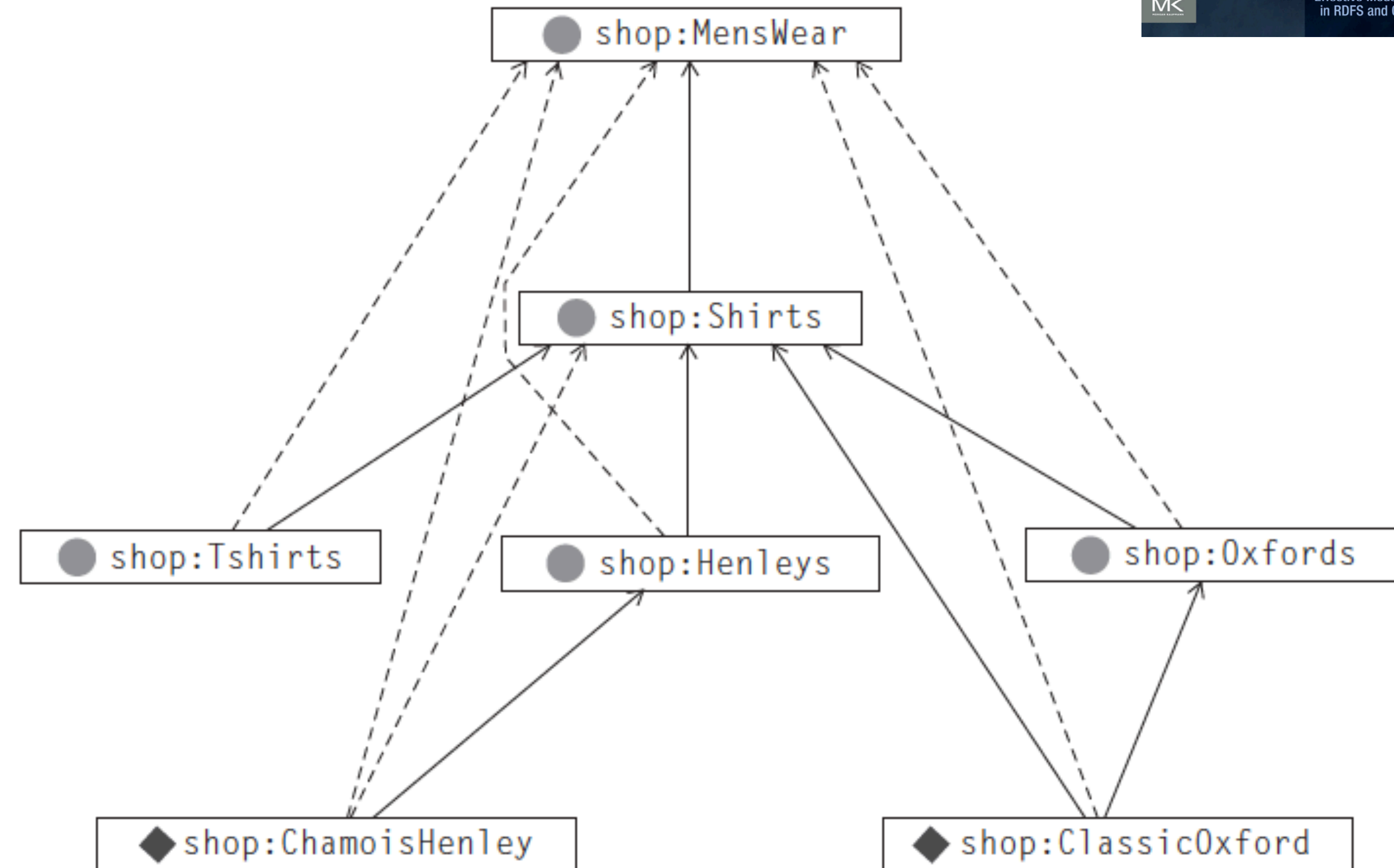


FIGURE 6.3

All triples in the catalog model. Inferred triples are shown as dashed lines.