

DR. MIGUEL CERIANI

PROF. ALEJANDRO VAISMAN

ONTOLOGÍAS EN LA WEB

2. RDF: MODELO Y FORMATOS

REPASO DE LA CLASE 1

- Ideas Generales:

- ▶ Web Semántica
- ▶ Datos Enlazados (Linked Data)

- Modelo de Datos:

- ▶ Datos en forma de Grafos, hechos de Triplas
- ▶ URIs para los conceptos

- Arquitectura Software:

- ▶ Linked Data Browser

REFERENCIAS PARA ESTA CLASE (RDF)

- ▶ Caps. 3 y 4 de "Semantic for the Working Ontologist"
- ▶ RDF 1.1 Primer
<https://www.w3.org/TR/rdf11-primer/>
- ▶ Semantic University:
[RDF-101](#), [RDF Nuts & Bolts](#), [XSD Datatype Cheat Sheet](#)

RDF (RESOURCE DESCRIPTION FRAMEWORK)

- ▶ **Recursos (Resources):** conceptos *representados con IRIs o ...*
dbpedia:Leonardo_da_Vinci
- ▶ **Propiedades (Properties):** nombres relaciones *representados con IRIs*
rdf:type
dcterms:title
- ▶ **Literales (Literals):** datos simples (textos, números)
"Mona Lisa"
42
3.14

RDF (RESOURCE DESCRIPTION FRAMEWORK)

- ▶ Declaraciones o Sentencias (Statements):

triplas (sujeto/predicado/objeto)

Sujeto: *IRIs* o ...

Predicado: *IRIs*

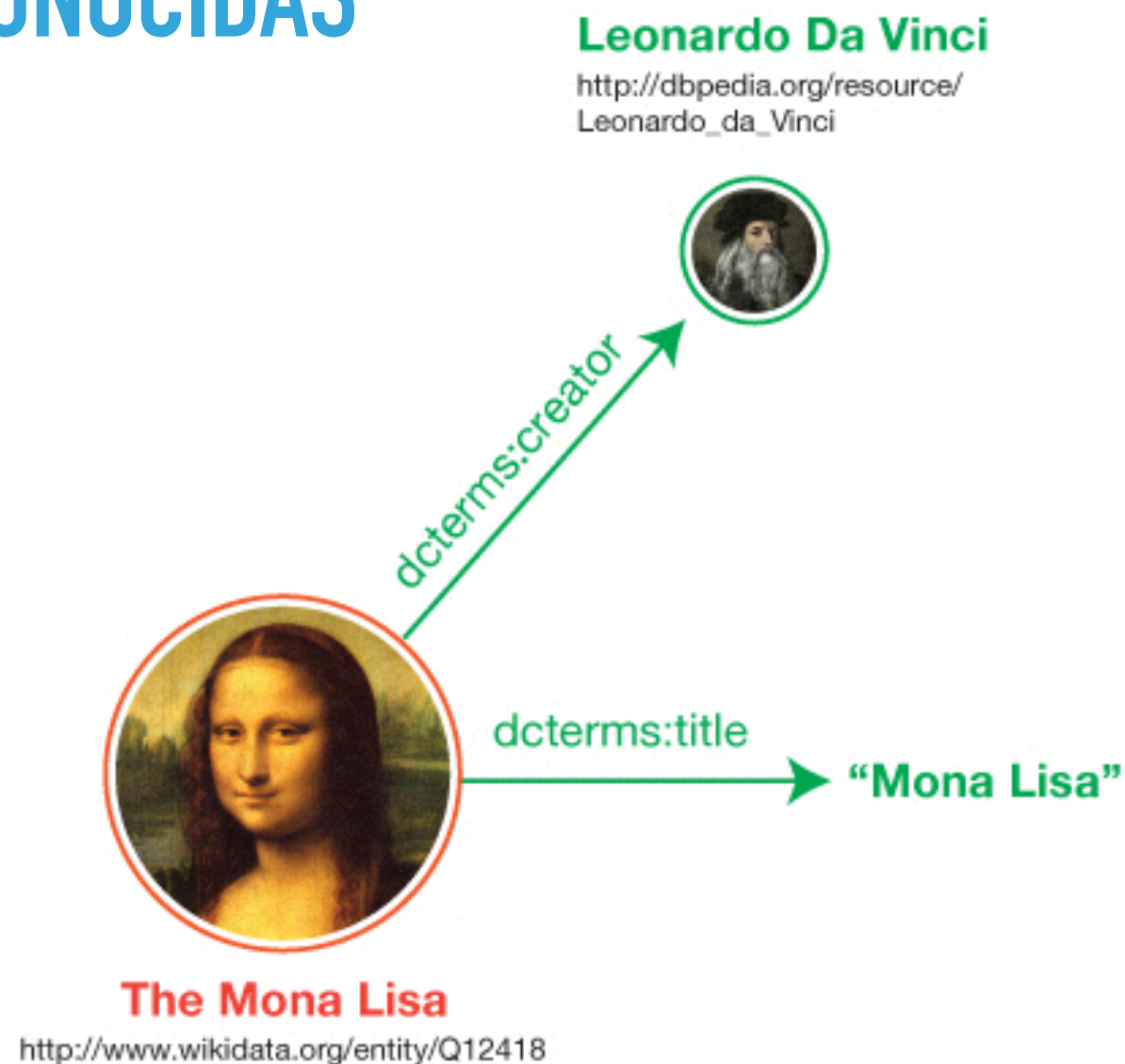
Objeto: *IRIs* o *literales*

- ▶ Grafos RDF (RDF Graphs):

Conjuntos de Declaraciones es decir...

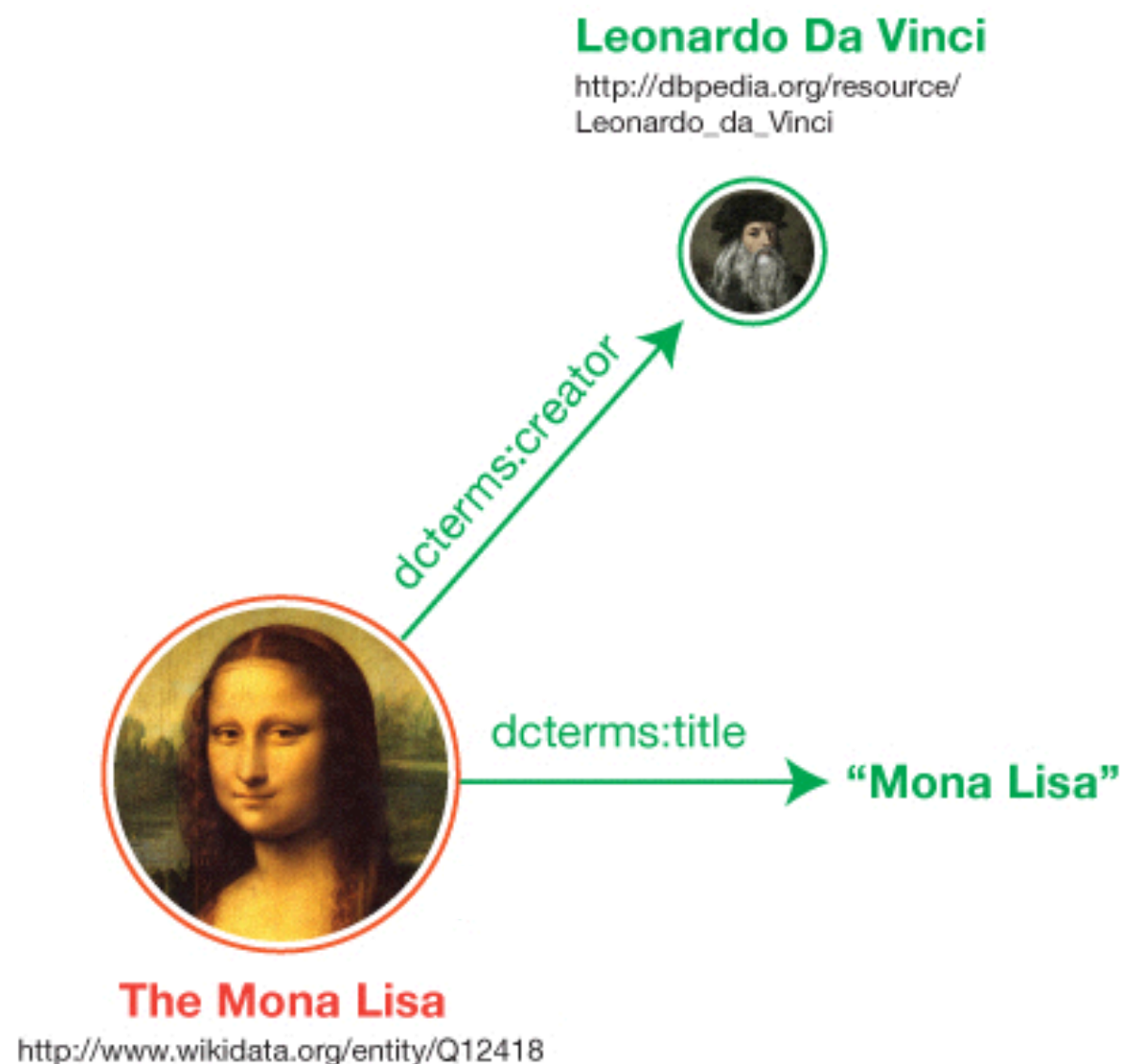
G es un grafo RDF si $G \subseteq (IRIs \cup B) \times IRIs \times (IRIs \cup L)$

GRAFO Y TRIPLAS YA CONOCIDAS



```
<Leonardo da Vinci> <is the creator of> <the Mona Lisa>.  
<the Mona Lisa> <has as title> <Mona Lisa>.
```

TRIPLAS EN RDF

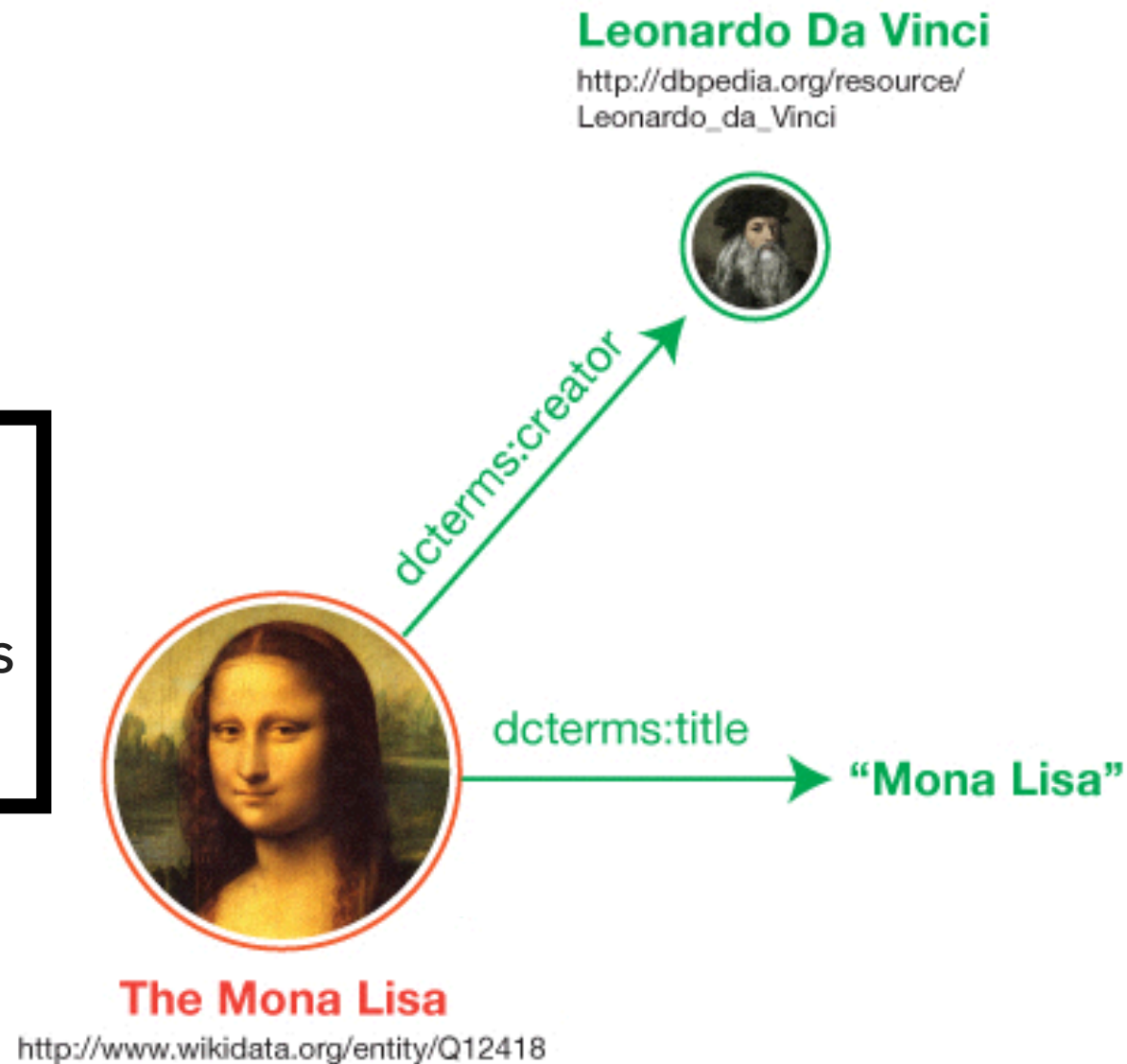


```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
wd:Q12418 dcterms:title "Mona Lisa".
wd:Q12418 dcterms:creator dbpedia:Leonardo_da_Vinci.
```

TRIPLAS EN RDF

Nota:

En el modelo RDF las URIs son completas, son las serializaciones que permiten usar prefijos



```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
wd:Q12418 dcterms:title "Mona Lisa".
wd:Q12418 dcterms:creator dbpedia:Leonardo_da_Vinci.
```


LITERALES

- ▶ **Texto "simple"**

"Mona Lisa"

- ▶ **Literal con Tipo**

"42"^^xsd:integer (simplificado 42)

"3"^^xsd:decimal (simplificado 3.0)

"Mona Lisa"^^xsd:string (lo mismo que "Mona Lisa")

"1990-07-04"^^xsd:date

- ▶ **Texto con Idioma**

"La Gioconda"@it

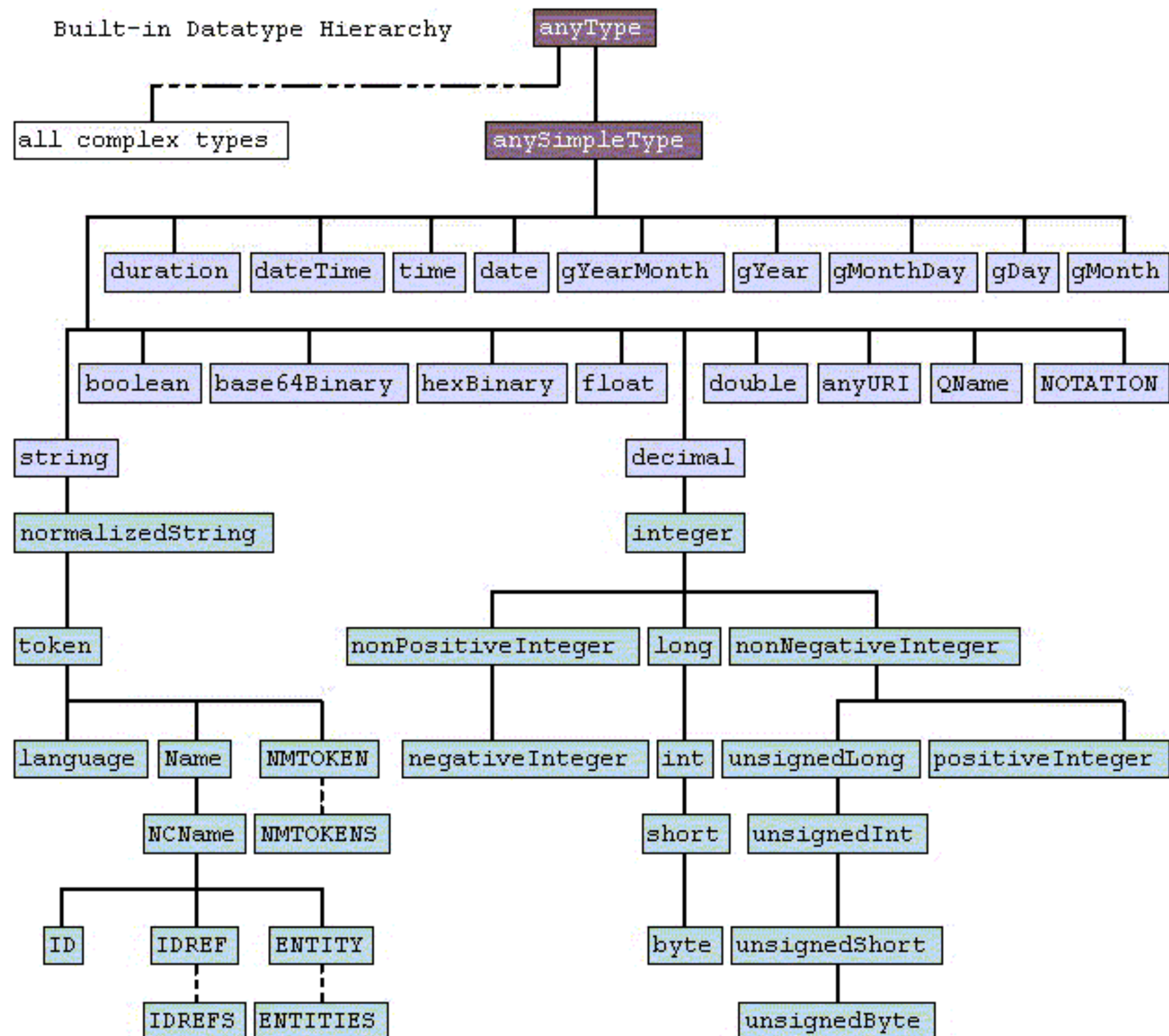
"La Joconde"@fr

TIPOS DE LOS LITERALES

- ▶ Usa los tipos primitivos de XML Schema (**X**ML **S**chema **D**atatype, ejemplos a la izquierda)
- ▶ Descritos con IRIs, nuevos tipos pueden ser introducidos libremente

<code>xsd:string</code>
<code>xsd:boolean</code>
<code>xsd:integer</code>
<code>xsd:decimal</code>
<code>xsd:double</code>
<code>xsd:boolean</code>
<code>xsd:date</code>
<code>xsd:time</code>
<code>xsd:dateTime</code>
<code>xsd:duration</code>

TIPOS DE LOS LITERALES



ur types



built-in primitive types



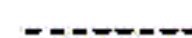
built-in derived types



complex types



derived by restriction



derived by list

derived by extension or
restriction

SERIALIZACIONES RDF (FORMATOS)

- ▶ **RDF/XML**: basada en XML
- ▶ **N-Triples**: una tripla cada renglón
- ▶ **Turtle**: sintética para escribir y leer
- ▶ **RDFa**: RDF incorporado en HTML
- ▶ **JSON-LD**: basada en JSON

NEGOCIACIÓN DE CONTENIDO HTTP (CONTENT NEGOTIATION)

- ▶ **Content-Type:** formato del contenido del mensaje
- ▶ **Accept:** formato(s) aceptado(s) en respuesta

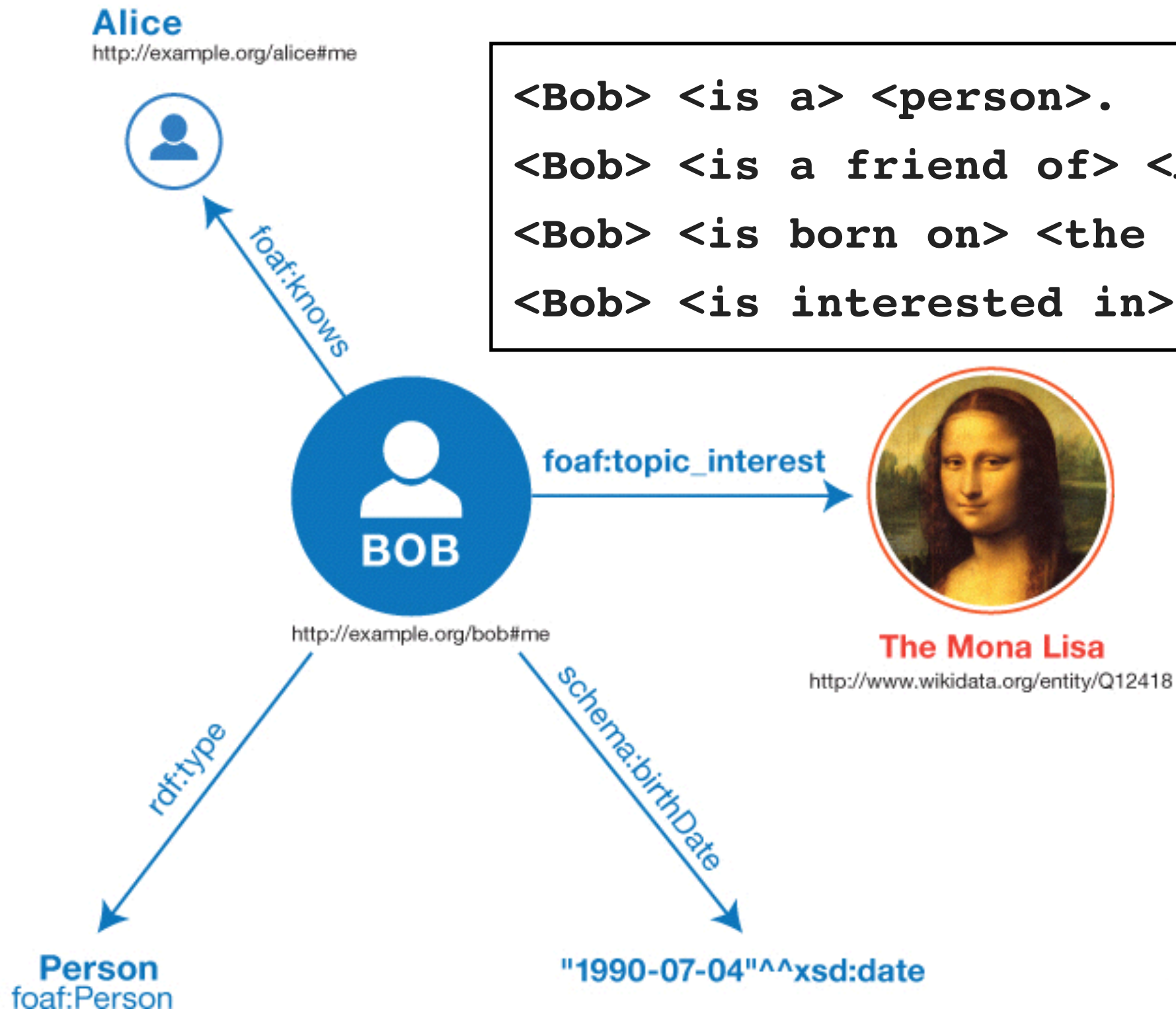
NEGOCIACIÓN DE CONTENIDO HTTP (CONTENT NEGOTIATION)

Formato	Mime Type	Extensión
Turtle	text/turtle	.ttl
N-Triples	application/n-triples	.nt
RDF/XML	application/rdf+xml	.rdf
JSON-LD	application/ld+json	.jsonld
TriG	application/trig	.trig
N-Quads	application/n-quads	.nq

TURTLE

- ▶ Prefijos
- ▶ Base URI para URIs relativos
- ▶ Uso de ';' cuando el sujeto es el mismo
- ▶ Uso de ',' cuando el sujeto y el predicado son los mismos
- ▶ Uso de 'a' en vez de `rdf:type`
- ▶ Sintaxis simple para **Números y Booleanos**
(4.0 en vez de `"4"^^xsd:decimal`)

GRAFO Y TRIPLAS YA CONOCIDAS (2)



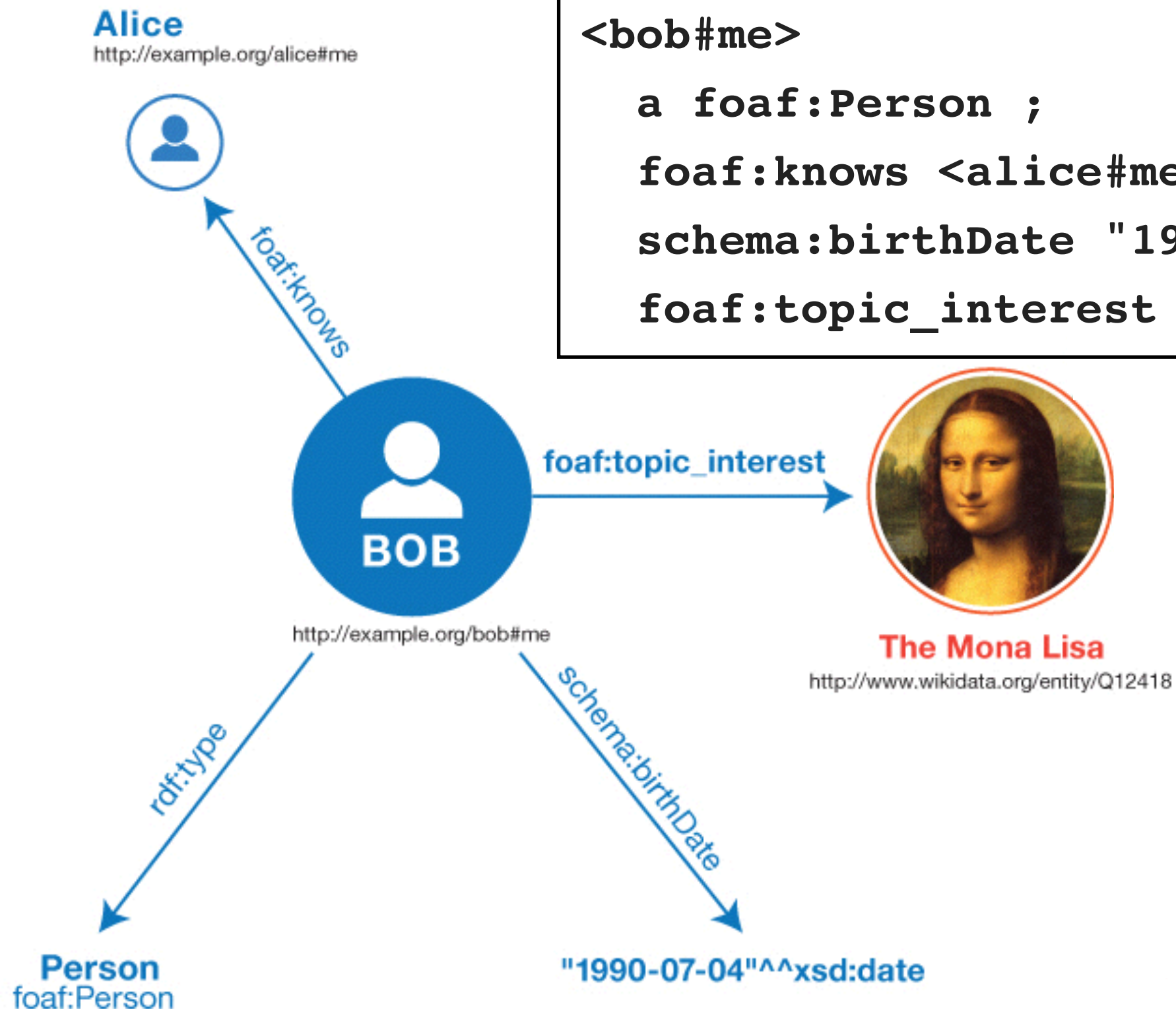
`<Bob> <is a> <person>.`

`<Bob> <is a friend of> <Alice>.`

`<Bob> <is born on> <the 4th of July 1990>.`

`<Bob> <is interested in> <the Mona Lisa>.`

TURTLE (FALTA DECLARACIÓN DE PREFIJOS)



<bob#me>

a foaf:Person ;

foaf:knows <alice#me> ;

schema:birthDate "1990-07-04"^^xsd:date;

foaf:topic_interest wd:Q12418 .

TURTLE (COMPLETO)

```
BASE    <http://example.org/>
PREFIX  foaf: <http://xmlns.com/foaf/0.1/>
PREFIX  xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX  schema: <http://schema.org/>
PREFIX  dcterms: <http://purl.org/dc/terms/>
PREFIX  wd: <http://www.wikidata.org/entity/>
```

```
<bob#me>
```

```
  a foaf:Person ;
  foaf:knows <alice#me> ;
  schema:birthDate "1990-07-04"^^xsd:date ;
  foaf:topic_interest wd:Q12418 .
```

ALGUNAS PROPIEDADES DE USO COMÚN

IRI	Asocia un recurso a...
<code>rdf:type</code>	su tipo (otro recurso)
<code>rdfs:label</code>	un nombre (texto)
<code>rdfs:comment</code>	una descripción (texto)
<code>rdfs:member</code>	un recurso contenido por este
<code>rdfs:seeAlso</code>	un recurso genericamente asociado
<code>owl:sameAs</code>	otro recurso con el mismo significado
<code>foaf:primaryTopic</code>	el argumento primario representado
<code>foaf:depiction</code>	una imagen que lo muestra
<code>dc:creator</code>	su creador/autor

ACTIVIDAD

- ▶ Descargar varios formatos RDF de DBpedia y similares
- ▶ Comparar los distintos formatos, sobretodo N-Triples y Turtle
- ▶ Buscar RDFa y microformat en paginas HTML
(OpenLink Structured Data Sniffer)
- ▶ Usar la negociación de contenidos HTTP
(DHC by Restlet, Advanced Rest Client Application...) `text/turtle`, `application/n-triples`,
`application/rdf+xml`, `application/ld+json`
- ▶ Crear un account Dydra (para actividades siguientes)

DECLARACIONES DE DECLARACIONES

Para decir algo sobre un hecho o relativizarlo:

- ▶ **Tiempo:** tiene validez en un intervalo temporal
- ▶ **Origen:** por ejemplo para la credibilidad
- ▶ y otros usos

DECLS. DE DECLS. 1: REIFICACIÓN (REIFICATION)

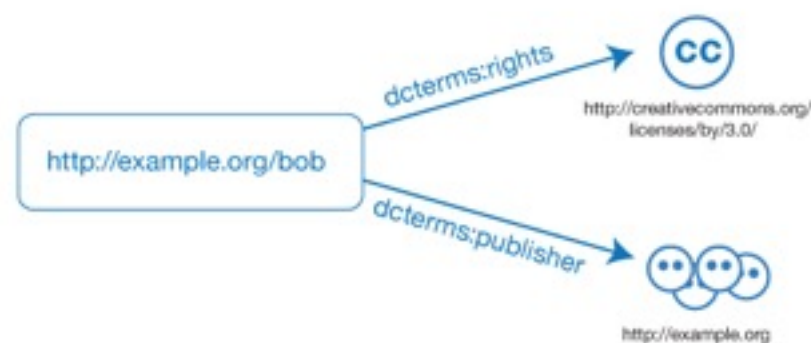
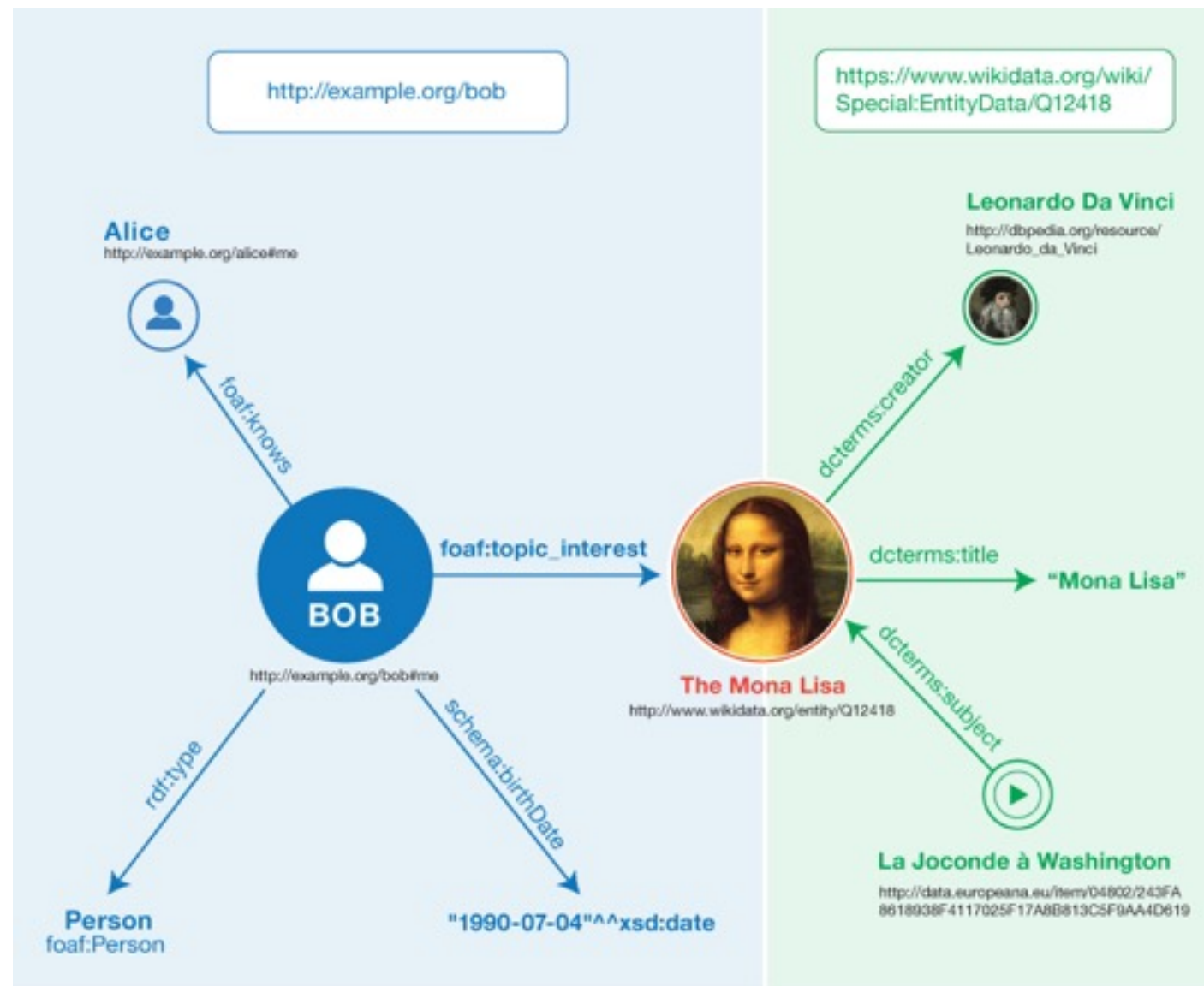
- ▶ El artículo item10245 pesa 2.4Kg

```
exproducts:item10245  exterms:weight  "2.4"^^xsd:decimal.
```

- ▶ El empleado 85740 dice que el artículo item10245 pesa 2.4Kg

```
exproducts:triple12345  rdf:type          rdf:Statement.  
exproducts:triple12345  rdf:subject       exproducts:item10245.  
exproducts:triple12345  rdf:predicate     exterms:weight.  
exproducts:triple12345  rdf:object        "2.4"^^xsd:decimal.  
  
exproducts:triple12345  dc:creator        exstaff:85740.
```

DECLS. DE DECLS. 2: GRAFOS CON NOMBRE (NAMED GRAPHS)



DECLS. DE DECLS. 2: GRAFOS CON NOMBRE (NAMED GRAPHS)

```
GRAPH <http://example.org/bob> {  
  <bob#me>  
    a foaf:Person ;  
    foaf:knows <alice#me> ;  
    schema:birthDate "1990-07-04"^^xsd:date ;  
    foaf:topic_interest wd:Q12418 .  
}
```

```
GRAPH <https://www.wikidata.org/wiki/Special:EntityData/Q12418> {  
  wd:Q12418  
    dcterms:title "Mona Lisa" ;  
    dcterms:creator dbpedia:Leonardo_da_Vinci .  
  <http://data.europeana.eu/...D619> dcterms:subject wd:Q12418 .  
}
```

```
<http://example.org/bob>  
  dcterms:publisher <http://example.org> ;  
  dcterms:rights <http://creativecommons.org/licenses/by/3.0/> .
```


DECLS. DE DECLS. 2: GRAFOS CON NOMBRE (NAMED GRAPHS)

Dataset RDF:

- ▶ un grafo RDF de default (default graph)
- ▶ zero o más grafos RDF con nombre (named graphs)

DECLS. DE DECLS. 2: GRAFOS CON NOMBRE (NAMED GRAPHS)

Sintaxis:

- ▶ Turtle → TriG
- ▶ N-Triples → N-Quads
- ▶ JSON-LD (OK)

GRAPH STORE (O TRIPLE STORE)

- ▶ Mantiene un conjunto de datos RDF
- ▶ Permite importación/exportación en varios formatos
- ▶ Permite interrogar los datos
- ▶ Puede permitir la integración con datos externos

GRAPH STORE: PROTOCOLOS DE INTERACCIÓN

- ▶ **SPARQL 1.1 Graph Store HTTP Protocol**
usa GET, PUT, POST para descargar, crear, editar grafos
- ▶ **SPARQL 1.1 Update**
usa variante de SPARQL para modificar (INSERT, DELETE)
- ▶ **Linked Data Platform (LDP)**
usa REST, centrado en recursos (basado en principios LD)

DYDRA: TRIPLE STORE ONLINE Y GRATUITO*!

- ▶ Mantiene varios repositorios RDF por cada usuario
- ▶ Permite importación/exportación en muchos formatos RDF
- ▶ Permite interrogar los datos con SPARQL
- ▶ Permite la integración con datos externos (pero solo pagando)

* Gratuito para fines de evaluación

FRONT-END (EN UNA APLICACIÓN DE LA WEB S.)

- Permite visualizar/crear/editar datos RDF
- Ofrece una Interfaz al Usuario (humano)
- Puede ser
 - ▶ Especifico (para un dataset o tipo de datos) o
 - ▶ Genérico (para cualquier dataset RDF)

UN FRONT-END GENERICO: OPENLINK RDF EDITOR

- Bajo Nivel: se editan directamente Triplas
- Se conecta a un Back-End (Graph Store) con varios protocolos:
 - ▶ SPARQL 1.1 Graph Store HTTP Protocol
 - ▶ SPARQL 1.1 Update
 - ▶ Linked Data Platform (LDP)

<http://linkeddata.uriburner.com/rdf-editor/>

ACTIVIDAD

- ▶ Importar datos en Dydra
<http://dydra.com/miguel/ontologias-web/>
- ▶ Importar/editar datos con OpenLink RDF Editor
<http://linkeddata.uriburner.com/rdf-editor/>
- ▶ Crear y importar nuevos datos a partir de los descargados
- ▶ Visualizarlos (en Query usar la query SPARQL de default)
- ▶ Probar a Explorarlos (con Graph Rover de Dydra)

PARA LA PRÓXIMA VEZ

- ▶ Activar un account en Dydra
- ▶ Instalar OpenRDF Sesame Workbench

RDF

EMAIL

mceriani@itba.edu.ar