

RDF

Grecia García García
grecia@matem.unam.mx

El Web

- El web ha cambiado el mundo:
 - la forma en que nos comunicamos...
 - acceso a millones de recursos...
 - realización de transacciones...
- Principales problemas:
 - sobrecarga de información
 - heterogeneidad en la información (problemas de interoperabilidad)

Sobre XML....

```
<?xml version="1.0"?>
```

```
<grupo>
```

```
  <integrantes>
```

```
    <alumno>
```

```
      <nombre>Ana Karina</nombre>
```

```
    </alumno>
```

```
  </integrantes>
```

```
</grupo>
```

```
<?xml version="1.0"?>
```

```
<grupo>
```

```
  <integrantes>
```

```
    <gato>
```

```
      <nombre>Rosco</nombre>
```

```
    </gato>
```

```
  </integrantes>
```

```
</grupo>
```

Web Semántico

Así, en 1998 surge un proyecto por Tim Berners-Lee.

(Presidente del World Wide Web Consortium, W3C)

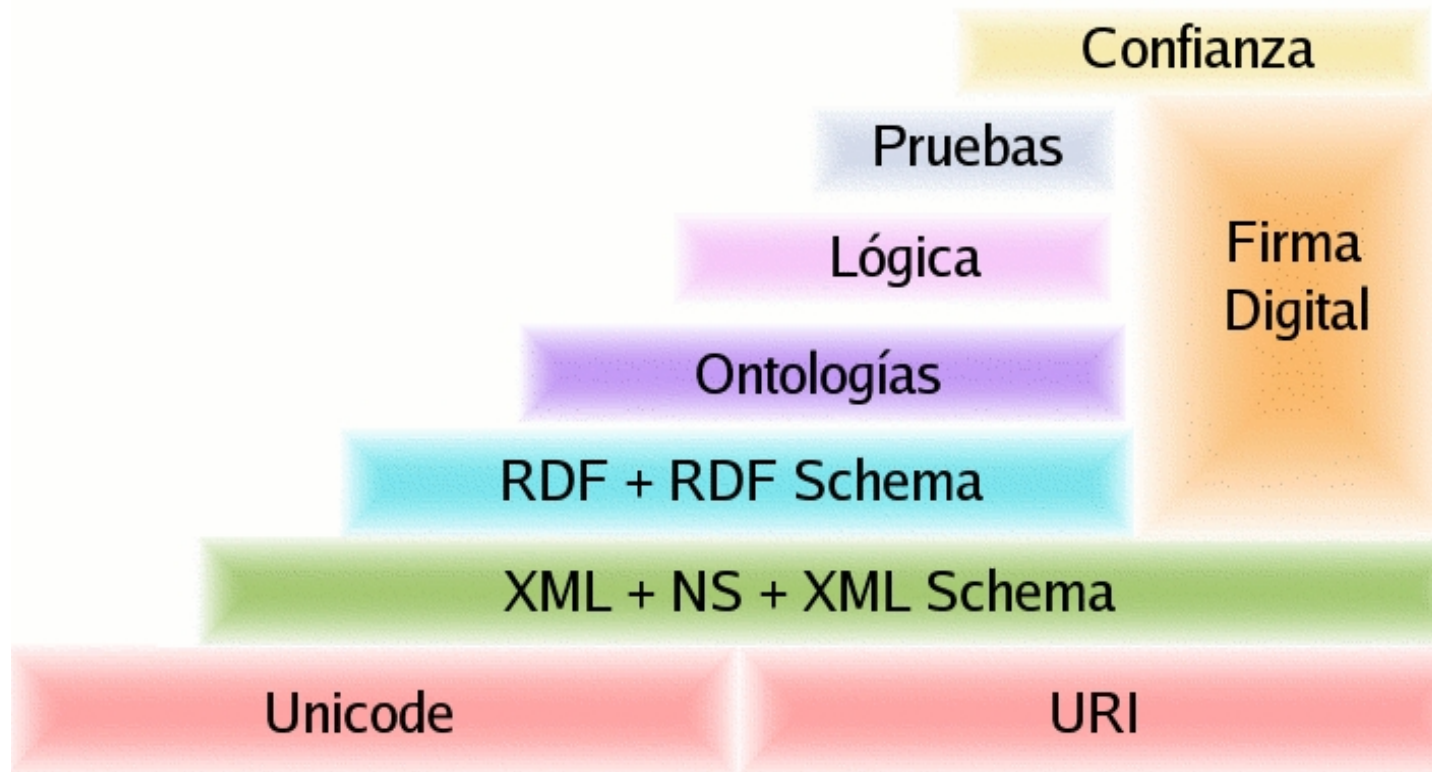


El Web Semántico

Permite a los usuarios delegar las tareas al software:
procesar contenido, razonarlo, realizar deducciones lógicas

Web Semántico

Extensión del Web actual con un significado bien definido para la información, habilitando a la gente y computadoras a trabajar en cooperación.



RDF

- ▶ *Resource Description Format*
- ▶ Un lenguaje para *representar información* a cerca de los recursos del Web.
- ▶ Particularmente para representar *metadatos* a cerca de los recursos.
- ▶ Información *procesable* por computadora.

RDF

▶ *Resource Description Format*

- ▶ Identificar Recursos usando identificadores llamado *URIref* (URI references)

URI + identificador

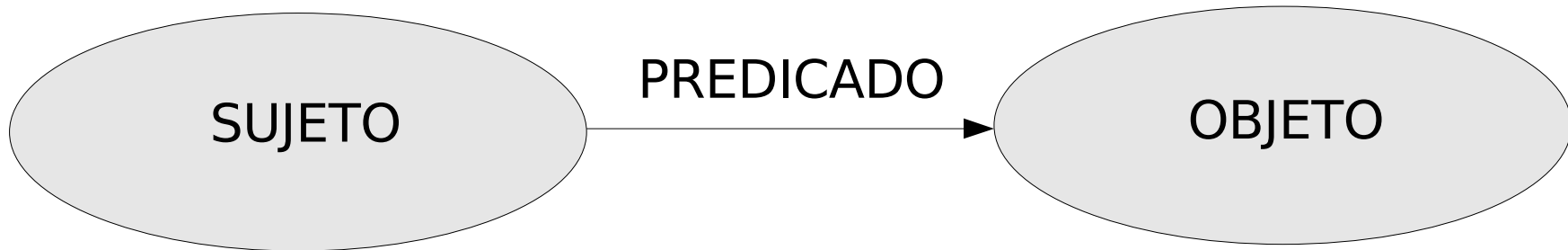
Por ejemplo:

<http://ejemplo.org/colores.html#rojo>

RDF define un **recurso** como cualquier cosa identificable por un URIref

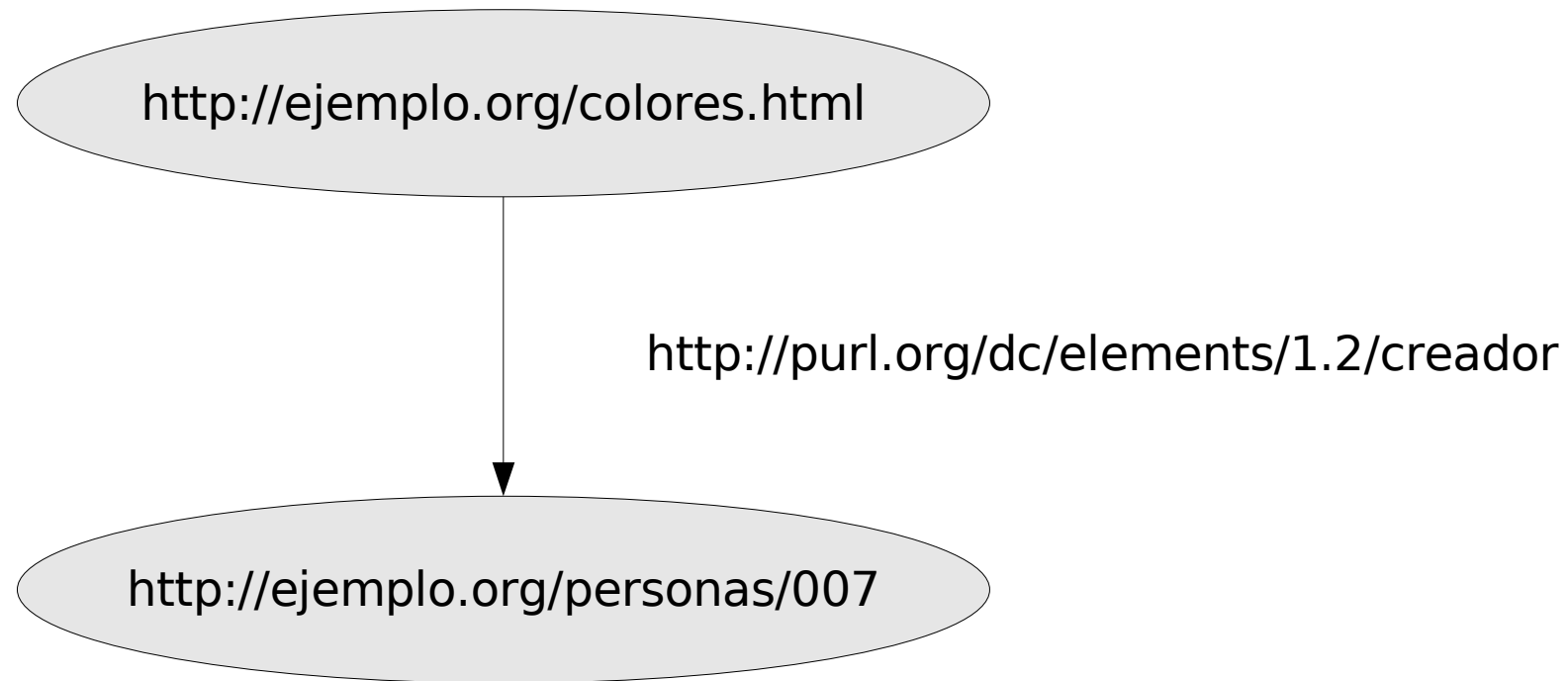
Modelo de Datos

- ▶ Proporciona un modelo de datos de gráficas para dar información sobre los recursos.
- ▶ Describir los recursos en términos de enunciados simples formados por: ***sujeto***, ***predicado*** y ***objeto***.



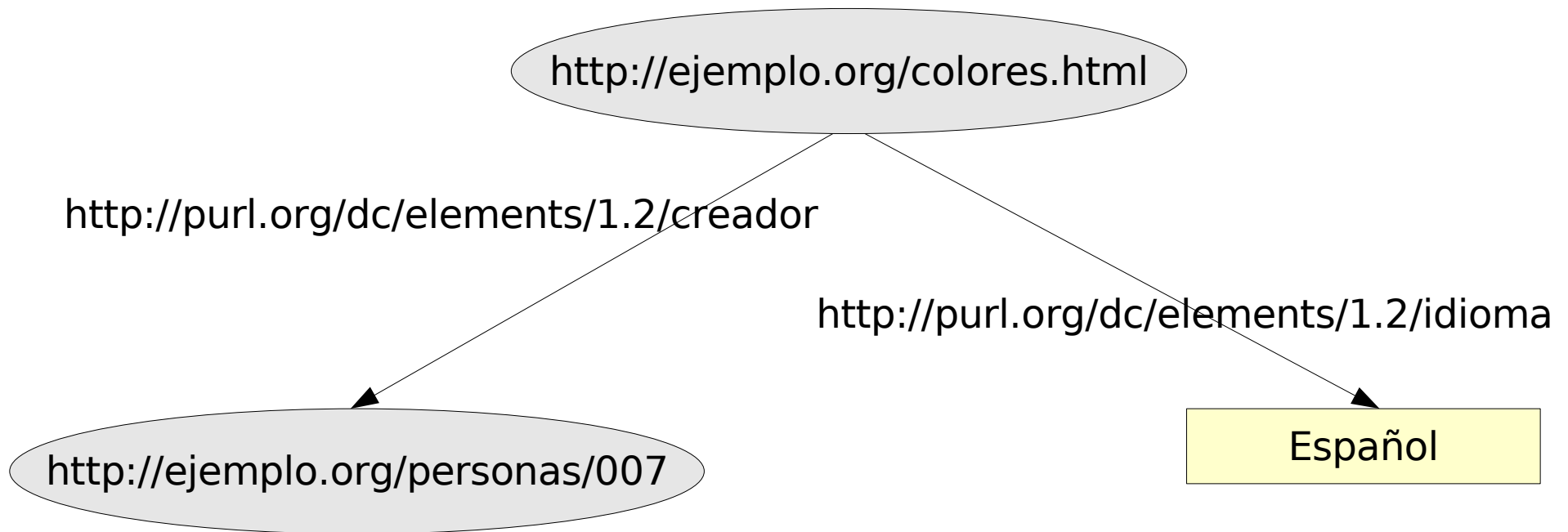
Ejemplos

La página **<http://ejemplo.org/colores.html>** tiene como **creador** a **Ana**



Ejemplos

La página **<http://ejemplo.org/colores.html>** tiene como **creador** a **Ana**
La página **<http://ejemplo.org/colores.html>** tiene como **idioma** el **Español**



Presentación de información

Por medio de **Tripletas**:

```
<http://ejemplo.org/colores.html>  
    <http://purl.org/dc/elements/1.2/creador>  
        <http://ejemplo.org/personas/007> .
```

```
<http://ejemplo.org/colores.html>  
    <http://purl.org/dc/elements/1.2/idioma>  
        "Español" .
```



Demasiado largo...

Abreviación de la información

- **Qualified name** (*QName*)

Contiene un *prefijo* que identifica al *espacio de nombres*, seguido de “:” y luego el **nombre local**

Por ejemplo:

<http://purl.org/dc/elements/1.2/idioma>

Y para el espacio de nombres:

<http://purl.org/dc/elements/1.2/>

asignamos el prefijo **dc**, entonces tenemos:

[dc:idioma](#)

Presentación de información

Tripletas:

<http://ejemplo.org/colores.html>
 <http://purl.org/dc/elements/1.2/creador>
 <http://ejemplo.org/personas/007> .

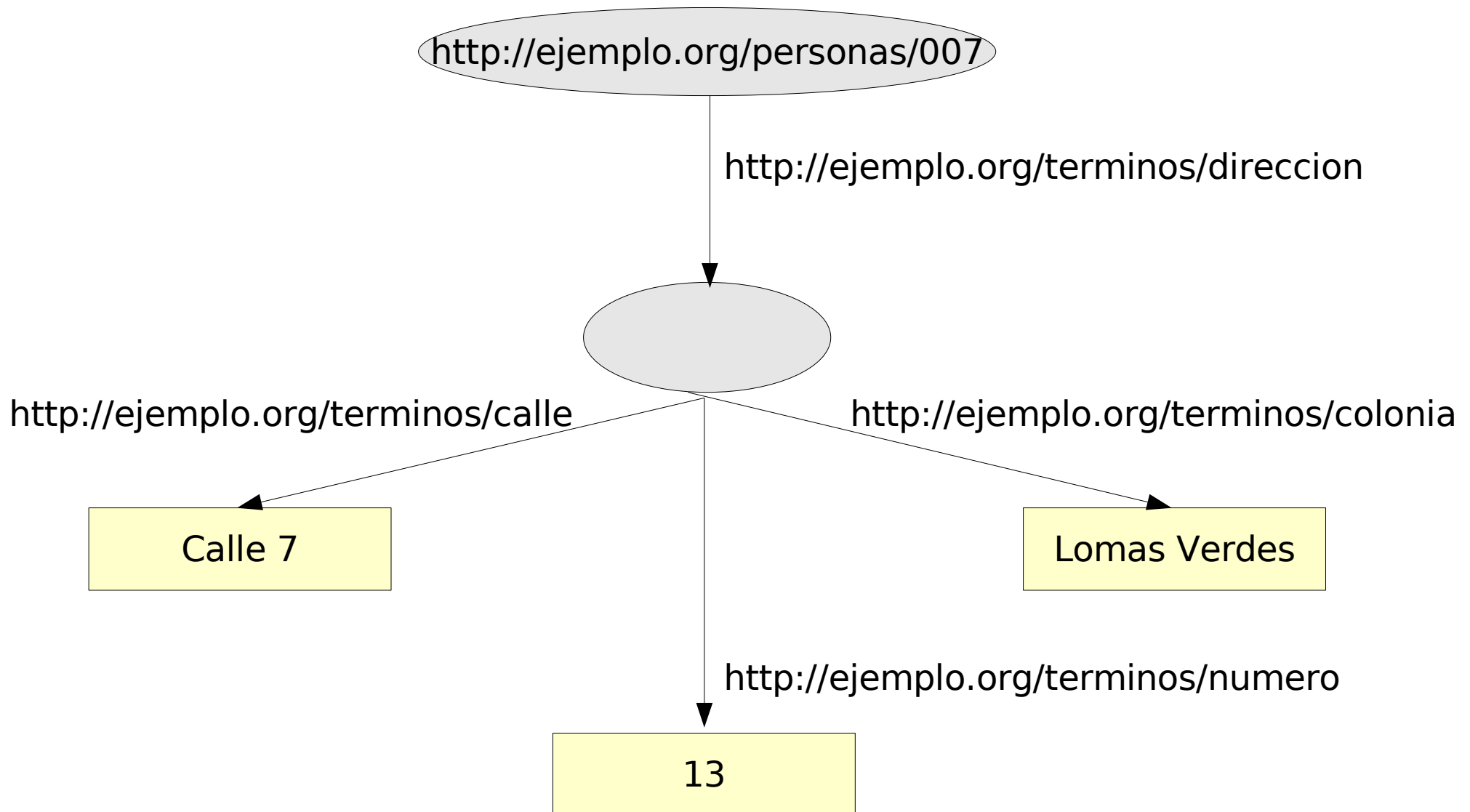
<http://ejemplo.org/colores.html>
 <http://purl.org/dc/elements/1.2/idioma>
 “Español” .

O también:

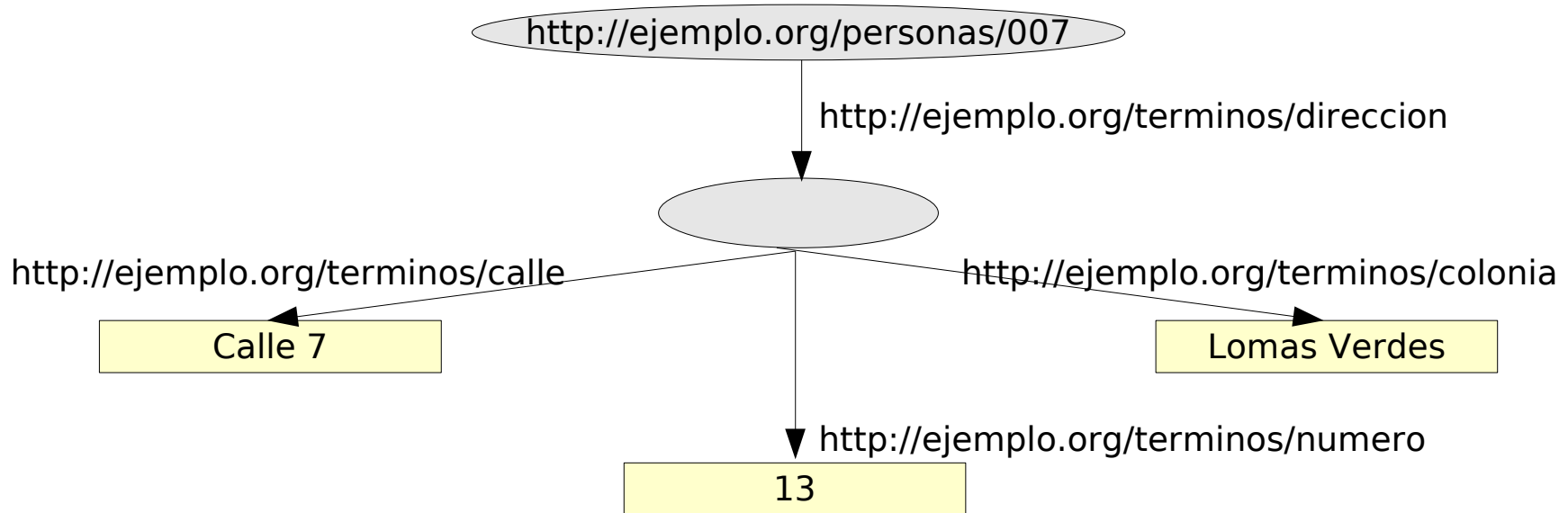
ej:colores.html **dc**:creador **ejp**:007 .
ej:colores.html **dc**:idioma “Español” .

donde	ej	corresponde a	http://ejemplo.org/
	dc	corresponde a	http://purl.org/dc/elements/1.2/
	ejp	corresponde a	http://ejemplo.org/personas/

Nodos en blanco o recursos anónimos



Nodos en blanco



ejp:007
_ :dirAna
_ :dirAna
_ :dirAna

ejt:direccion
ejt:calle
ejt:nreumero
ejt:colonia

"_ :dirAna" .
"Calle 7" .
"13" .
"Lomas Verdes" .

Con:

ejp
ejt

como
como

`http://ejemplo.org/personas/`
`http://ejemplo.org/terminos/`

Nodos en blanco

Permiten hacer enunciados sobre recursos que no tienen un URI (como las personas).

<code>_:ana</code>	<code>ejt:correo</code>	<code><mailto:ana@ejemplo.org> .</code>
<code>_:ana</code>	<code>rdf:type</code>	<code>foaf:Person .</code>
<code>_:ana</code>	<code>foaf:name</code>	<code>"Ana Liveira" .</code>
<code>_:ana</code>	<code>ejt:empID</code>	<code>"23748" .</code>
<code>_:ana</code>	<code>ejt:edad</code>	<code>"26" .</code>

Literales con tipo

¿Cómo saber cómo debemos interpretar los valores?

En tripletas:

```
<http://ejemplo.org/personas/007>  
  <http://ejemplo.org/terminos/edad>  
    "27"^^<http://www.w3.org/2001/XMLSchema#integer> .
```

O también:

```
ejp:007      ejt:edad      "27"^^xsd:integer .
```

Otro ejemplo:

```
ej:colores.html  ejt:fecha-creacion  "2005-09-06"^^xsd:date .
```

Presentación en RDF/XML

[ej:colores.html](#) [ejt:fecha-creacion](#) "Septiembre 6, 2005" .

1. `<?xml version="1.0"?>`
2. `<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:ejt="http://ejemplo.org/terminos/">`
3. `<rdf:Description rdf:about="http://ejemplo.org/colores.html">`
4. `<ejt:fecha-creacion>Septiembre 6, 2005</ejt:fecha-creacion>`
5. `</rdf:Description>`
6. `</rdf:RDF>`

[ej:colores.html](#) [ejt:idioma](#) "Español" .

```
<rdf:Description rdf:about="http://ejemplo.org/colores.html">  
  <ejt:idioma>Español</ejt:idioma>  
</rdf:Description>
```

Presentación en RDF/XML

ej:colores.html ejt:fecha-creacion "2005-09-06"^^xsd:date .
ej:colores.html ejt:idioma "Español" .

1. `<?xml version="1.0"?>`
2. `<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">`
3. `xmlns:ejt="http://ejemplo.org/terminos/">`
4. `<rdf:Description rdf:about="http://ejemplo.org/colores.html">`
5. `<ejt:fecha-creacion rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2005-09-06`
`</ejt:fecha-creacion>`
6. `<ejt:idioma>Español</idioma>`
7. `</rdf:Description>`
8. `</rdf:RDF>`

Presentación en RDF/XML

1. `<?xml version="1.0"?>`
2. `<!DOCTYPE rdf:RDF [<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">]>`
3. `<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">`
4. `xmlns:ejt="http://ejemplo.org/terminos/">`
5. `<rdf:Description rdf:about="http://ejemplo.org/colores.html">`
6. `<ejt:fecha-creacion rdf:datatype="&xsd:date">2005-09-06`
7. `<ejt:creador rdf:resource="http://ejemplo.org/personas/007"/>`
8. `</rdf:Description>`
9. `</rdf:RDF>`

¿Y los nodos en blanco cómo se representan?

Presentación en RDF/XML

Asignando un identificador (*blank node identifier*) usando `rdf:nodeID`

Otras presentaciones:

A relative URIref #item10245. The relative URIref would be understood as being a URIref defined relative to the base URIref of the catalog.

*Using a similar abbreviation, the URIref of the tent could also be given by specifying `rdf:about="#item10245"` in the catalog entry (i.e., by specifying the relative URIref directly) instead of `rdf:ID="item10245"` synonyms: the full URIref formed by RDF/XML is the same in either case:
<http://www.example.com/2002/04/products#item10245>.*

Presentación en RDF/XML

<code>_:ana</code>	<code>ejt:correo</code>	<code><mailto:ana@ejemplo.org></code> .
<code>_:ana</code>	<code>rdf:type</code>	<code>foaf:Person</code> .
<code>_:ana</code>	<code>foaf:name</code>	<code>"Ana Liveira"</code> .
<code>_:ana</code>	<code>ejt:empID</code>	<code>"23748"</code> .
<code>_:ana</code>	<code>ejt:edad</code>	<code>"26"</code> .

1. `<?xml version="1.0"?>`
2. `<!DOCTYPE rdf:RDF [<!ENTITY foaf "http://www.w3.org/2001/XMLSchema#" >]>`
3. `<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">`
4. `xmlns:ejt="http://ejemplo.org/terminos/">`
5. `<rdf:Description rdf:nodeID="_:ana">`
6. `<ejt:correo rdf:resource="mailto:ana@ejemplo.org"/>`
7. `<rdf:type rdf:resource="&foaf;Person"/>`
8. `<foaf:name>Ana Liveira</foaf:name>`
9. `<ejt:empID>23748</ejt:empID>`
10. `<ejt:edad>26</ejt:edad>`
11. `</rdf:Description>`
12. `</rdf:RDF>`

Presentación en RDF/XML

1. `<?xml version="1.0"?>`
2. `<!DOCTYPE rdf:RDF [<!ENTITY foaf "http://www.w3.org/2001/XMLSchema#">]>`
3. `<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">`
4. `xmlns:ejt="http://ejemplo.org/terminos/">`
5. `<foaf:Person rdf:nodeID="_:ana">`
6. `<ejt:correo rdf:resource="mailto:ana@ejemplo.org"/>`
7. `<foaf:name>Ana Liveira</foaf:name>`
8. `<ejt:empID>23748</ejt:empID>`
9. `<ejt:edad>26</ejt:edad>`
10. `</foaf:Person>`
11. `</rdf:RDF>`

Otras capacidades

Contenedores:

- * rdf:Bag
- * rdf:Seq
- * rdf:Alt

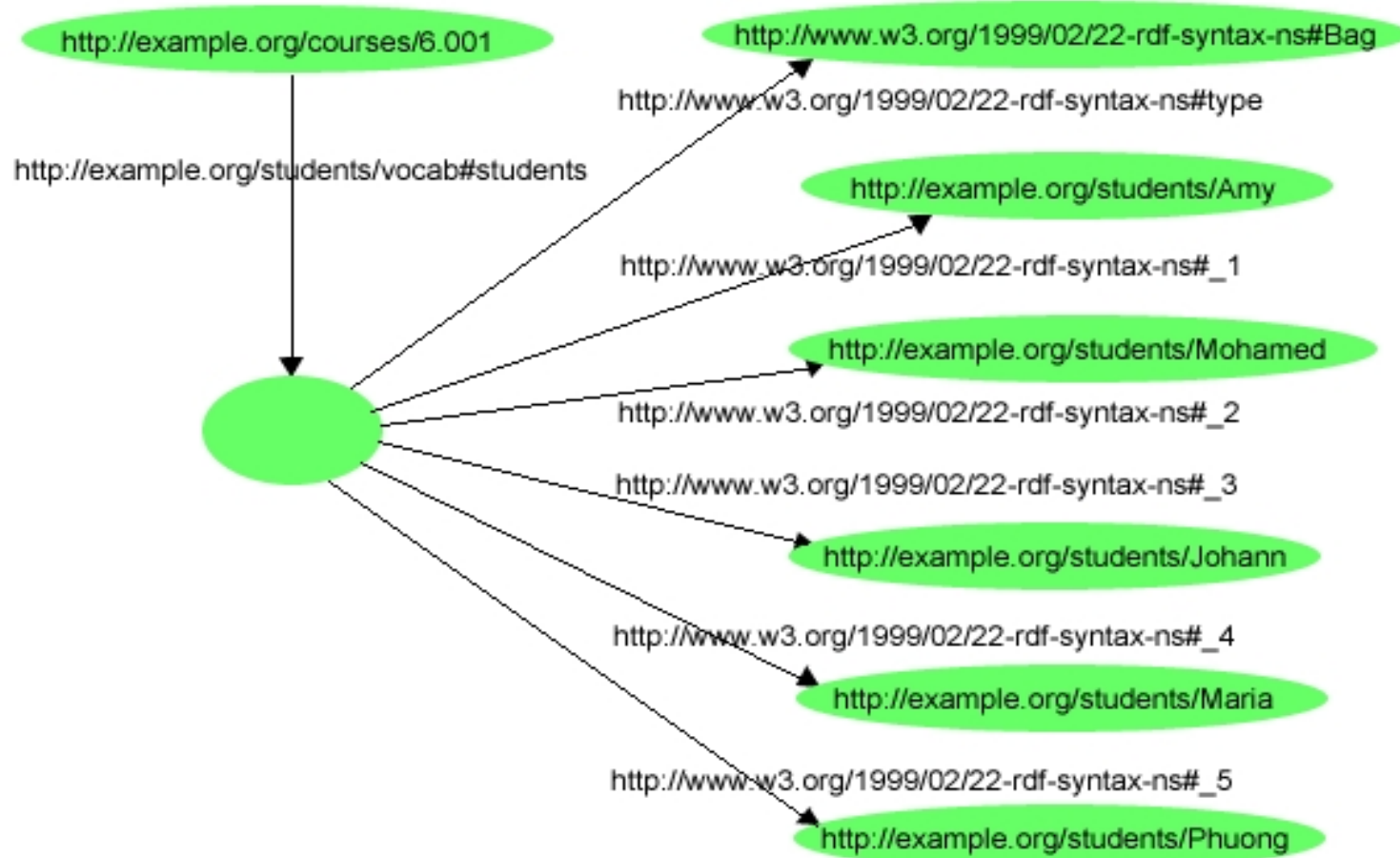
Colecciones:

- * Una forma de indicar estos son todos los miembros de un conjunto. En RDF es una colección de cosas en una estructura de lista.

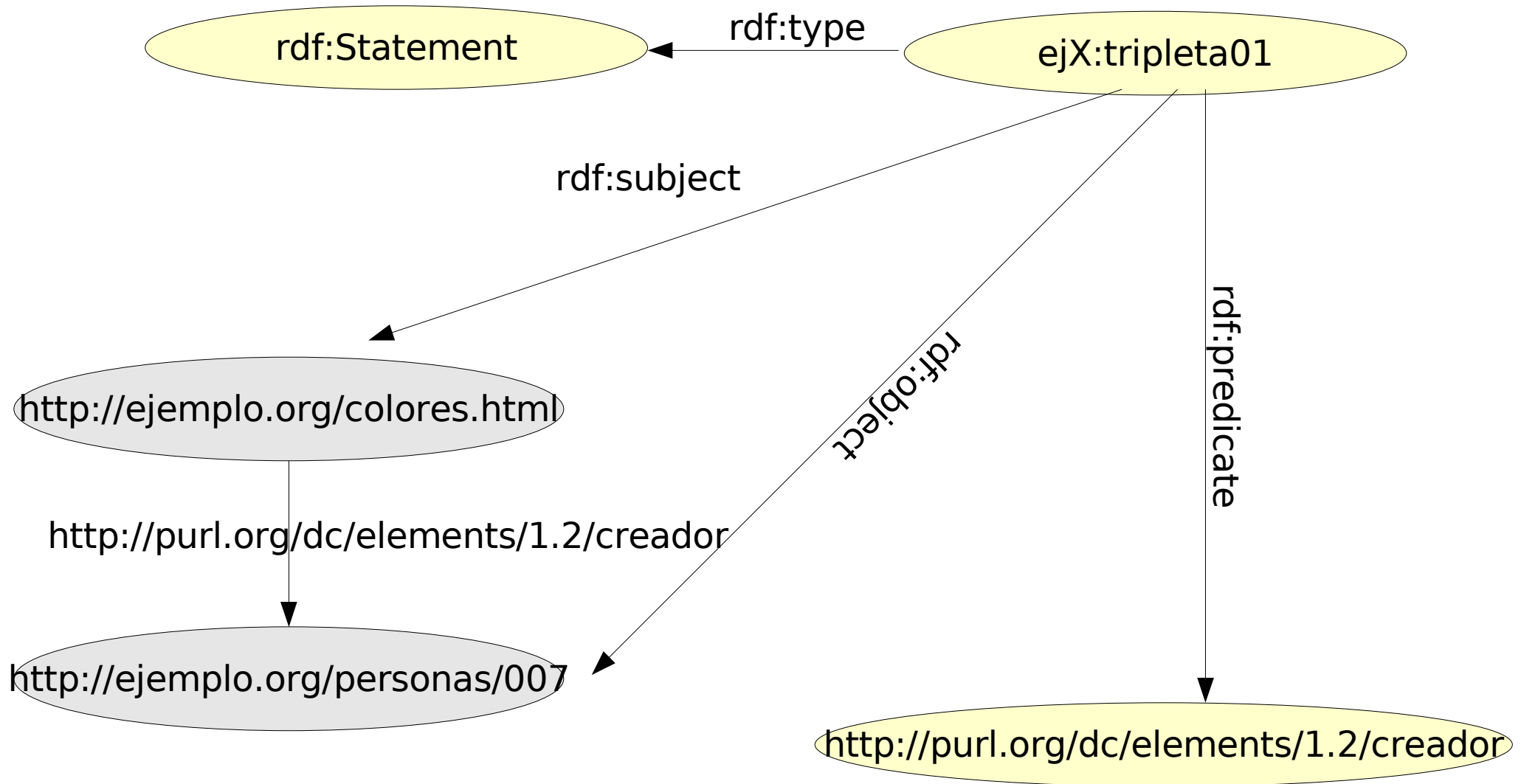
Reification:

- * Propiedades sobre los enunciados RDF.

Otras capacidades



Otras capacidades



RDFS

▶ RDF Schema

- ▶ Permite definir nuevos vocabularios.
- ▶ Se pueden crear **clases** de objetos y definir nuevas **propiedades**

RDFS

- ▶ Clases definidas con: `rdfs:Class` y `rdf:Property`
- ▶ Propiedades: `rdf:type`, `rdfs:subClassOf`, `rdfs:domain` y `rdfs:range`

Con *rdfs* como `http://www.w3.org/2000/01/rdf-schema#`
rdf como `http://www.w3.org/1999/02/22-rdf-syntax-ns#`

Ejemplo: (escuela)

`ej:Materia`

`ej:Seminario`

`rdf:type`

`rdfs:SubClassOf`

`rdfs:Class`

`ej:Materia`

RDFS

ej:Persona
ej:profesor
ej:profesor

rdf:type
rdf:type
rdfs:range

rdfs:Class
rdf:Property
ej:Persona

ej:Libro
ej:autor
ej:autor

rdf:type
rdf:type
rdfs:domain

rdfs:Class
rdf:Property
ej:Persona

¿En qué uso RDF?

- RSS

Permite catalogar información sobre noticias y eventos para encontrar información adaptada a las preferencias de los usuarios

- FOAF

Permite crear páginas para describir personas y las relaciones entre ellas

- En interfaces gráficas, en documentación, en control de acceso, *ien lo que quieras!*

FIN