

Fundamentos de la Web Semántica

Modelos Semánticos y XML

Modelos semánticos

Modelos de datos que permiten representar y asociar semántica a los datos en diferentes escenarios del mundo real.

Existen diferentes modelos semánticos con diferente expresividad.

¿Para qué sirven los modelos?

- **Para la comunicación entre las personas**
 - Describen una situación de forma tal que otras personas puedan comprenderla
- **Para exponer ideas y entender un fenómeno)**
 - Relacionan fenómenos más simples con otros más complejos
- **Para obtener diferentes vistas de un mismo fenómeno**
 - Permiten visualizar los aspectos comunes y las diferencias en dos puntos de vista diferentes del mismo fenómeno
 - Representar el mismo fenómeno con diferente grado de detalle

¿Para qué sirven los modelos?

- **Para la comunicación entre las personas**

- Describen una situación de forma tal que otras personas puedan comprenderla

Ejemplo: Interpretación de una ley

El texto de la ley es un **modelo** de la situación que aborda la ley.

¿La interpretación es única?

¿Para qué sirven los modelos?

- **Para la comunicación entre las personas**

- Describen una situación de forma tal que otras personas puedan comprenderla

Ejemplo: Interpretación de una ley

El texto de la ley es un **modelo** de la situación que aborda la ley.

¿La interpretación es única?

Datos + Interpretación = Conocimiento

¿Para qué sirven los modelos?

Ejemplo:

$$2 + 2 = 4$$

¿Siempre el resultado es 4? ¿De qué depende?

¿Para qué sirven los modelos?

Ejemplo:

$$2 + 2 = 4$$

¿Siempre el resultado es 4? ¿De qué depende?

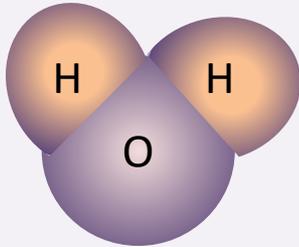
De llegar a un acuerdo en el significado de “2”, “4”, “+” y “=”

Esto permite aplicar la suma en otras situaciones

Reusar el modelo

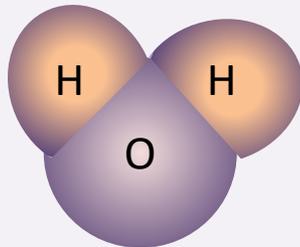
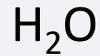
¿Para qué sirven los modelos?

H₂O



¿En qué difieren estos modelos?

¿Para qué sirven los modelos?



¿En qué difieren estos modelos?

- **Para obtener diferentes vistas de un mismo fenómeno**
 - Permiten visualizar los aspectos comunes y las diferencias en dos puntos de vista diferentes del mismo fenómeno
 - Representar el mismo fenómeno con diferente grado de detalle

Diferente expresividad

Modelos en sistemas de información

Modelos conceptuales

Modelos de flujos de datos

Modelos lógicos

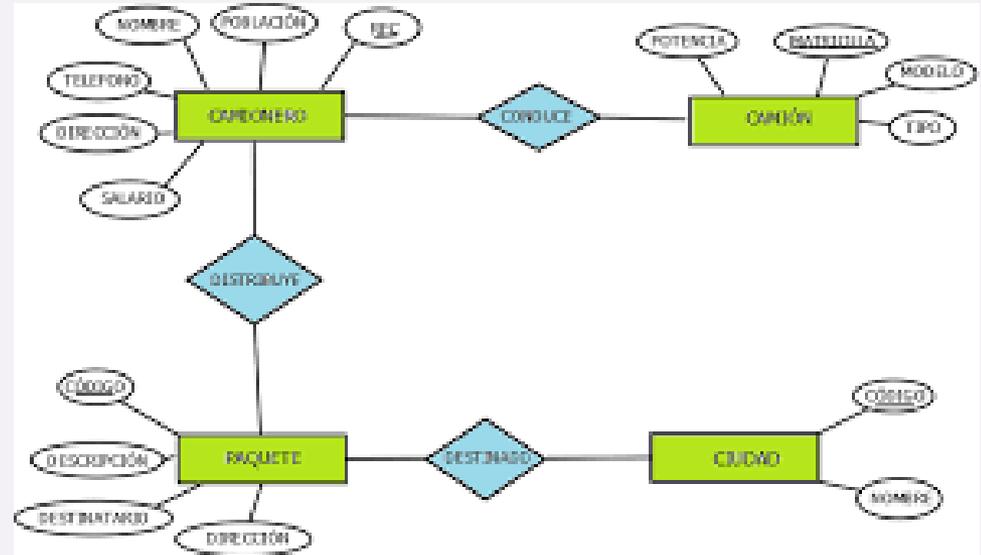
Modelos físicos

Bases de datos

Modelos conceptuales

Modelos lógicos

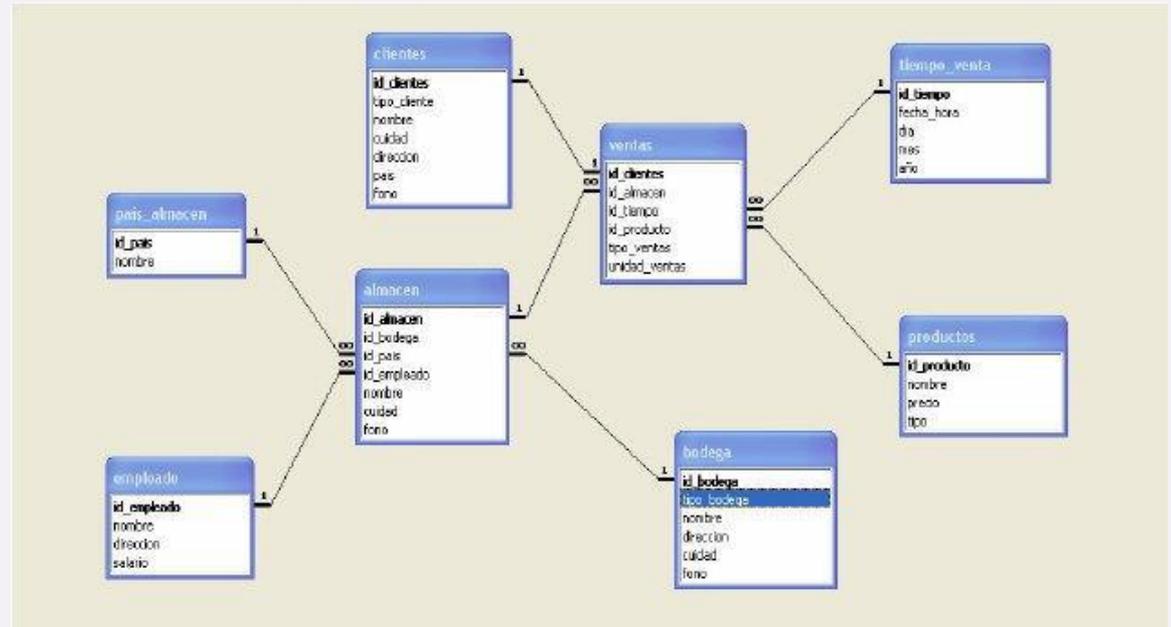
Modelos físicos



Bases de datos

Modelos conceptuales

Modelos lógicos



Modelos físicos

Bases de datos

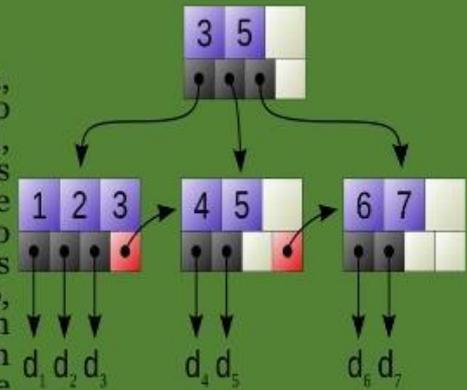
Modelos conceptuales

Modelos lógicos

Modelos físicos

Arboles B+

- En ciencias de la computación, un **árbol B+** es un tipo de estructura de datos de árbol, representa una colección de datos ordenados de manera que se permite una inserción y borrado eficientes de elementos. Es un índice, multinivel, dinámico, con un límite máximo y mínimo en el número de claves por nodo. Un árbol B+ es una variación de un árbol B.



Modelos – Nivel de expresividad

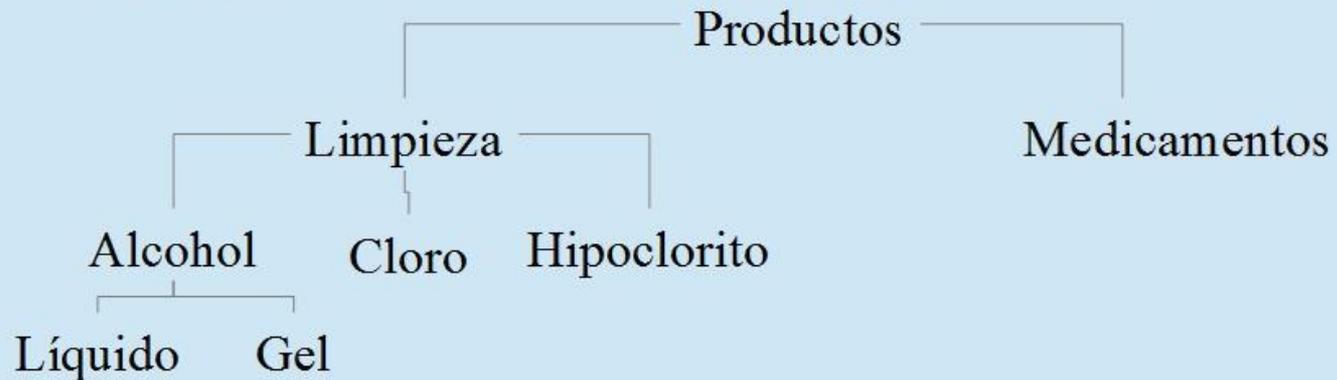


CÓDIGOS

productos de limpieza	01
Alcohol líquido	01011
Alcohol en gel	01012
Cloro	0102
Hipoclorito	0103

Vocabulario controlado

TAXONOMÍA



Modelos - Expresividad

- **Vocabulario Controlado**

Lista de términos de un área de interés.

- **Taxonomía**

Su función es clasificar elementos. Vocabulario controlado que tiene una **jerarquía** (términos más amplios y más específicos).

- **Tesauro**

Taxonomía que además provee información sobre:

- Sinónimos
- Término más utilizado, término referente, traducciones

- **Modelo Conceptual**

Permite además representar atributos y relaciones entre términos (conceptos), para un dominio de conocimiento.

- **Modelo orientado a objetos**

Se agrega herencia, etc.

- **Ontología**

Permite además representar relaciones transitivas, componer relaciones, etc.

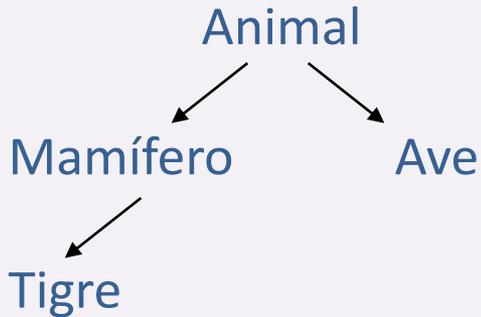
**Menos
Expresivo**



**Más
Expresivo**

Modelos Semánticos - Expresividad

Ejemplo: Taxonomía



Ejemplo: Axiomas lógicos (Ontología)

Todos los mamíferos son animales

Todas las aves son animales

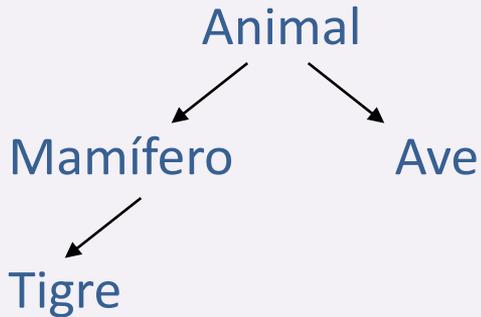
Ningún ave es mamífero

Todos los tigres son mamíferos

Los tigres son aves?

Modelos Semánticos - Expresividad

Ejemplo: Taxonomía



No se puede afirmar que los tigres no son aves

Ejemplo: Axiomas lógicos (Ontología)

Todos los mamíferos son animales

Todas las aves son animales

Ningún ave es mamífero

Todos los tigres son mamíferos

Se puede afirmar que los tigres no son aves

Los tigres son aves?

Modelos Semánticos - Lenguajes

- Permiten expresar los metadatos con diferente nivel de expresividad
- Proveen un marco formal
- Sirven como interfaz para compartir la información

Modelos Semánticos - Lenguajes

XML – Sintaxis para representar documentos estructurados

XML Schema – Agrega restricciones a la estructura de documentos XML

RDF – Lenguaje para representar recursos y relaciones entre ellos. Provee una semántica simple.

RDF Schema – Lenguaje que describe clases y propiedades de recursos RDF. Provee una semántica para representar jerarquía de clases y propiedades.

OWL – Lenguaje que permite describir relaciones y propiedades con múltiples restricciones (clases disjuntas, cardinalidad de propiedades) y características (propiedades simétricas, funcionales, transitivas, compuestas).
Está basado en el formalismo Lógica Descriptiva.

Menos
Expresivo



Más
Expresivo

Modelos Semánticos VS Lenguajes

Semántica fuerte

Logica Modal

Lógica de Primer Orden

Lógica Descriptiva

DAML+OIL, OWL

Relac. con propiedades transitivas

UML

Modelo Conceptual

RDF Schema

Relac. con propiedades (herencia) Interoperabilidad Semántica

Thesauros

"tiene-significado-mas-cercano-a"

DB Schemas, XML-S

Interoperabilidad Estructural

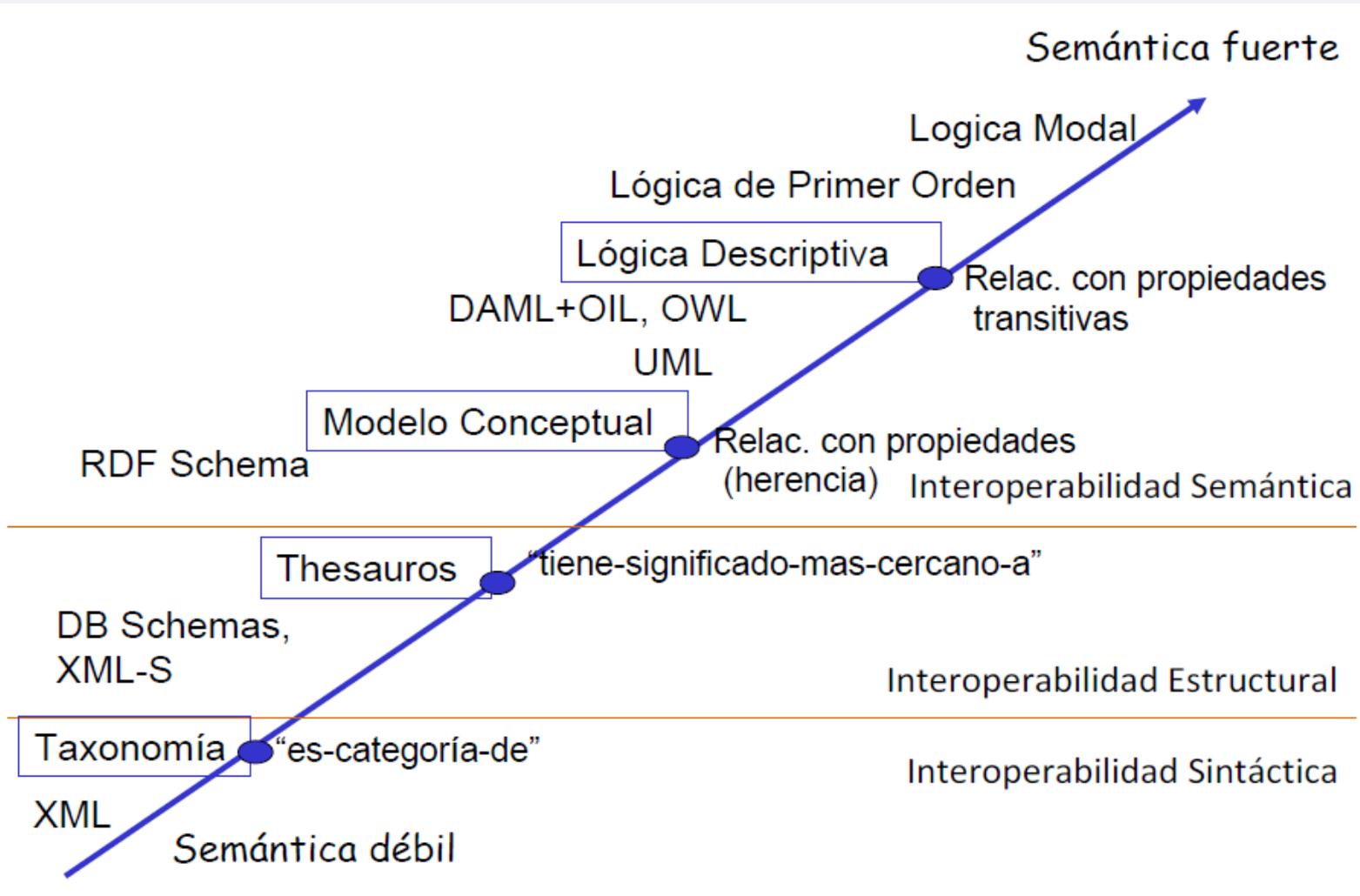
Taxonomía

"es-categoría-de"

XML

Interoperabilidad Sintáctica

Semántica débil



XML – eXtensible Markup Language



XML – eXtensible Markup Language

Tim Bray: “XML will be the ASCII of the Web - basic, essential, unexciting.”

Lenguaje para describir documentos

- Estándar **W3C** (World Wide Web Consortium)
- **Intercambio** de documentos
- Comercio Electrónico
- Gerenciamiento de documentos

XML vs HTML

<h2> A Semantic Web Primer </h2>
<i> por Gregoris Antoniou and Frank van Harmelen </i>

 The MIT Press 2004
 ISBN 0-262-01210-3

```
<libro>
  <titulo>
    A Semantic Web Primer
  </titulo>
  <autor>
    Gregoris Antoniou
  </autor>
  <autor>
    Frank van Harmelen
  </autor>
  <editor>
    The MIT Press
  </editor>
  <año>
    2004
  </año>
  <ISBN>
    0-262-01210-3
  </ISBN>
</libro>
```

Ambos usan tags

¿Para qué sirven las tags ?

XML vs HTML

<h2> A Semantic Web Primer **</h2>**
<i> por **** Gregoris Antoniou **** and **** Frank van Harmelen **** *</i>*
**
** The MIT Press 2004 **
** ISBN 0-262-01210-3

<libro>
 <titulo>
 A Semantic Web Primer
 </titulo>
 <autor>
 Gregoris Antoniou
 </autor>
 <autor>
 Frank van Harmelen
 </autor>
 <editor>
 The MIT Press
 </editor>
 <año>
 2004
 </año>
 <ISBN>
 0-262-01210-3
 </ISBN>
</libro>

Para mostrar información

Ambos usan tags

¿Para qué sirven las tags ?

El usuario decide:

- Para qué usa las tags
- Qué tags usar

XML vs HTML

```
<h2> A Semantic Web Primer </h2>
<i> por <b> Gregoris Antoniou </b> and <b> Frank van Harmelen </b> </i>
<br> The MIT Press 2004 <br> ISBN 0-262-01210-3
```

```
<libro>
  <titulo>
    A Semantic Web Primer
  </titulo>
  <autor>
    Gregoris Antoniou
  </autor>
  <autor>
    Frank van Harmelen
  </autor>
  <editor>
    The MIT Press
  </editor>¿¿
  <año>
    2004
  </año>
  <ISBN>
    0-262-01210-3
  </ISBN>
</libro>
```

Para que un agente recupere el título del libro:

Del documento XML puede hacerlo fácilmente ¿por qué?

XML vs HTML

```
<h2> A Semantic Web Primer </h2>  
<i> por <b> Gregoris Antoniou </b> and <b> Frank van Harmelen </b> </i>  
<br> The MIT Press 2004 <br> ISBN 0-262-01210-3
```

```
<libro>  
  <titulo>  
    A Semantic Web Primer  
  </titulo>  
  <autor>  
    Gregoris Antoniou  
  </autor>  
  <autor>  
    Frank van Harmelen  
  </autor>  
  <editor>  
    The MIT Press  
  </editor>¿¿  
  <año>  
    2004  
  </año>  
  <ISBN>  
    0-262-01210-3  
  </ISBN>  
</libro>
```

Para que un agente recupere el título del libro:

Del documento XML puede hacerlo fácilmente ¿por qué?



Porque las tags describen la información

Debe haber acuerdo en el vocabulario utilizado

XML

Comunidades de diferentes dominios han definido vocabularies especializados o “extensiones”:

- **MathML** → Matemáticas
- **Xlink** → Links entre recursos
- **XMLNS** → Espacios de nombres
- **Xpointer** → Fragmentos de documentos XML
- **XSL** → Formato de documentos XML
- **DOM** → Modelo para manipular documentos XML

XML

Un documento XML contiene:

- Declaración y referencia a documento de estructura (opcional)
- Elementos: contenido descrito por el documento
 - Definiciones de etiquetas
 - Estructuras se pueden anidar con profundidad arbitraria

XML

- **Elementos:** contenido descrito por el documento
 - Definiciones de etiquetas
 - Estructuras se pueden anidar con profundidad arbitraria

Etiquetas anidadas

**Propiedades con atributos:
Nombre, valor**

```
< persona >  
  < Nombre >  
    < PrimerNombre idioma="español" > Pablo </PrimerNombre >  
    < PrimerNombre idioma="inglés" > Paul </PrimerNombre >  
    < Apellido > Martinez </Apellido >  
  </Nombre >  
  < edad > 32 </edad >  
  < e-mail > jjj@abc.com </e-mail >  
< /persona >
```

XML – Documentos bien formados

Cumplen las **reglas de sintaxis, gramática y estructura de XML**
un único elemento raíz,
los elementos deben estar correctamente anidados,
los nombres de etiquetas deben seguir la sintaxis establecida

```
< persona >
```

```
  < 1nombre idioma="inglés" > Paul </1nombre >
```

```
  < edad > 32
```

```
  < e-mail > jjj@abc.com < /e-mail >
```

```
< /persona >
```

XML – Documentos bien formados

Cumplen las **reglas de sintaxis, gramática y estructura de XML**

un único elemento raíz,

los elementos deben estar correctamente anidados,

los nombres de etiquetas deben seguir la sintaxis establecida

```
< persona >
```

```
  < 1nombre idioma="inglés" > Paul </1nombre >
```

```
  < edad > 32
```

```
  < e-mail > jjj@abc.com < /e-mail >
```

```
< /persona >
```

¿Es suficiente para intercambio de documentos?

- **Poca consistencia:** las etiquetas las entiende quien construye el documento, son arbitrarias.
- Datos contenidos en un árbol rotulado

Elemento: contiene elementos y datos

Dentro de un elemento: el orden entre los sub-elementos y los datos es **relevante**

Documentos Bien Formados

No es lo mismo que

Documentos Válidos

Documento Válido

Su estructura y restricciones de valores conforman con un documento XML Schema.

Documentos XML y Esquemas

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

```
< a >
```

```
  < b id="x1" >
```

```
    < c > du < /c >
```

```
    < c > du < /c >
```

```
  < /b >
```

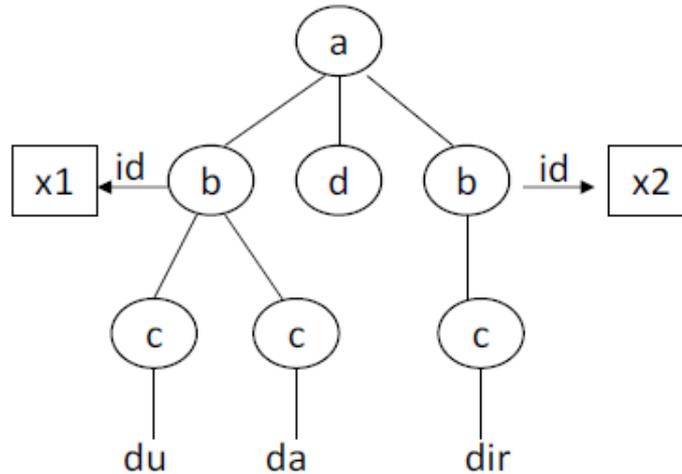
```
< d / >
```

```
  < b id="x2" >
```

```
    < c > dir < /c >
```

```
  < /b >
```

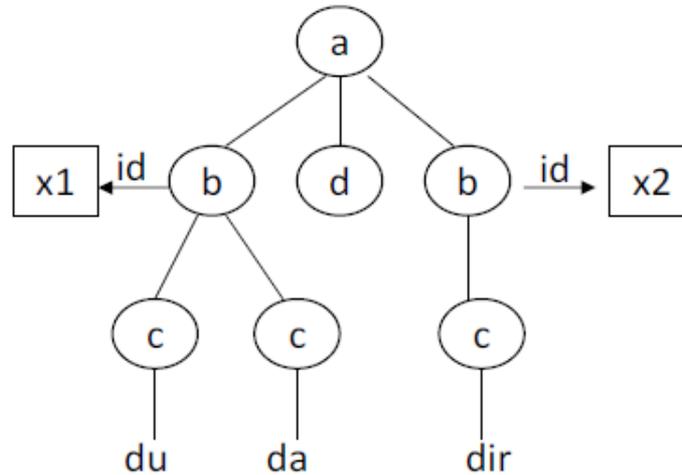
```
< /a >
```



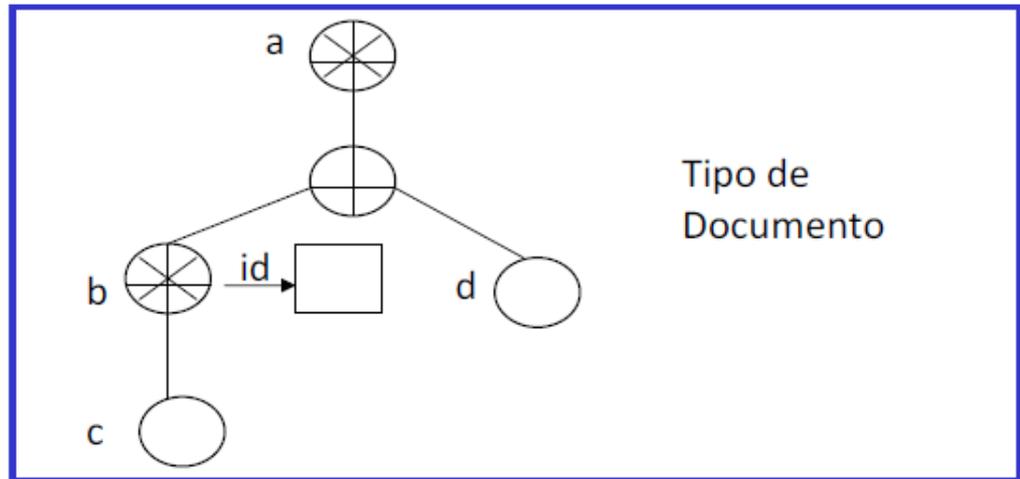
Estructura
Lógica

Documentos XML y Esquemas

```
< ?XML version="1.0"? >  
< a >  
  < b id="x1" >  
    < c > du < /c >  
    < c > du < /c >  
  < /b >  
  < d />  
  < b id="x2" >  
    < c > dir < /c >  
  < /b >  
< /a >
```



Estructura
Lógica



Tipo de
Documento

Documentos XML - Orden

Documentos estructurados:

Persona: {nombre: “Juan”, apellido : “Perez”}

Persona: {apellido : “Perez”, nombre: “Juan”}

Equivalentes

Documentos semi-estructurados XML:

```
< persona>  
  < nombre > Juan < /nombre>  
  < apellido > Perez < /apellido>  
< /persona>
```

```
< persona>  
  < apellido > Perez < /apellido>  
  < nombre > Juan < /nombre>  
< /persona>
```

NO Equivalentes

Documentos XML y Esquemas – XML Schema

```
<?xml version="1.0"?>
<hojaPedido fechaPedido="1999-10-20">
  <facturarA pais="EEUU">
    <nombre>Robert Smith</nombre>
    <calle>8 Oak Avenue</calle>
    <ciudad>Old Town</ciudad>
  </facturarA>
  <elementos>
    <elemento numProducto="872-AA">
      <nombreProducto>Cortacésped</nombreProducto>
      <cantidad>1</cantidad>
      <precioEEUU>148.95</precioEEUU>
      <comentario>Confirmar que es eléctrico</comentario>
    </elemento>
    <elemento numProducto="926-AA">
      <nombreProducto>Monitor para bebés</nombreProducto>
      <cantidad>1</cantidad>
      <precioEEUU>39.98</precioEEUU>
      <fechaEnvio>1999-05-21</fechaEnvio>
    </elemento>
  </elementos>
</hojaPedido>
```

Documentos XML y Esquemas – XML Schema

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
```

XML Namespace

```
<xsd:element name="hojaPedido" type="TipoHojaPedido"/>  
<xsd:element name="comentario" type="xsd:string"/>
```

Elementos

```
<xsd:complexType name="TipoHojaPedido">  
  <xsd:sequence>  
    <xsd:element name="facturarA" type="direccionEEUU"/>  
    <xsd:element name="elementos" type="Elementos"/>  
  </xsd:sequence>  
  <xsd:attribute name="fechaPedido" type="xsd:date"/>  
</xsd:complexType>
```

Definición de tipos

```
<xsd:complexType name="direccionEEUU">  
  <xsd:sequence>  
    <xsd:element name="nombre" type="xsd:string"/>  
    <xsd:element name="calle" type="xsd:string"/>  
    <xsd:element name="ciudad" type="xsd:string"/>  
  </xsd:sequence>  
  <xsd:attribute name="pais" type="xsd:NMTOKEN" fixed="EEUU"/>  
</xsd:complexType>
```

Documentos XML y Esquemas – XML Schema

```
<xsd:complexType name="Elementos">
  <xsd:sequence>
    <xsd:element name="elemento" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="nombreProducto" type="xsd:string"/>
          <xsd:element name="cantidad">
            <xsd:simpleType>
              <xsd:restriction base="xsd:positiveInteger">
                <xsd:maxExclusive value="100"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name="precioEEUU" type="xsd:decimal"/>
          <xsd:element ref="comentario" minOccurs="0"/>
          <xsd:element name="fechaEnvio" type="xsd:date" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="numProducto" type="SKU" use="required"/>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
.....
```

Documentos XML y Esquemas – XML Schema

```
<ejemplo xmlns:pasarela="http://www.fi.upm.es/PASARELA/"  
          xmlns:coches="http://www.fi.upm.es/COCHES/"  
          xmlns:coleccion="http://www.fi.upm.es/COLECCION/">  
  <pasarela:MODELO>Karolina Kurkova</pasarela:MODELO>  
  <coches:MODELO>BMW 530d</coches:MODELO>  
  <coleccion:MODELO>Americana de 3 botones</coleccion:MODELO>  
</ejemplo>
```

Ejemplo tomado de:

Curso básico de tecnologías XML,

Instituto Nacional de Administración Pública, España

XML y XML Schema

<https://www.w3.org/TR/REC-xml/>

<https://www.w3.org/TR/xmlschema11-1/>

Herramientas para validar

<https://www.liquid-technologies.com/online-xsd-validator>