

AI

Web Semántica en AI.

- ## ¿Por qué le interesa a los ambientes inteligentes?

SW: una visión desde la Documentación

- No es otra versión de la Web: **es la Web**.
- **Implica**
 - Crear una interfaz común a todos los conceptos y bases de conocimiento en la Web.
 - Crear sistema de información de acceso integrado a los datos, términos y conceptos (=información).
 - Aunar distintos recursos que existen en diferentes sistemas o servicios de información digital en la Web (diccionarios, tesauros, ontologías, etc.)
 - Facilitar la búsqueda por materias y temática

¿Qué es realmente la Web?

La expresión de la interoperabilidad de la información y el conocimiento a nivel mundial

Dato, información y conocimiento

- **Dato**: información que incluye aspectos útiles y también irrelevantes. Debe ser procesado para darle sentido.
- **Información**: Conocimiento obtenido de la investigación, estudio o instrucción y expresado en algún lenguaje.
- **Conocimiento**: cuerpo organizado de información. Afirmaciones probadas como verdaderas.

Definiciones de la Web Semántica

W3C:

La Web Semántica es una Web extendida, *dotada de mayor significado* en la que cualquier usuario en Internet podrá encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a *una información mejor definida*. Al dotar a la Web de *más significado* y, por lo tanto, *de más semántica*, se pueden obtener soluciones a problemas habituales en la *búsqueda de información*.

SW: Definición

- **T. Berners-Lee, J. Hendler, O. Lassila**

The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.

SW: Definición enriquecida

“La Web Semántica es una extensión de la actual Web en la que a la información disponible se le otorga (anota o marca con) una **semántica** bien definida.

Está basada en la idea de proporcionar en la Web **datos** definidos y enlazados, permitiendo que aplicaciones heterogéneas descubran, integren, razonen, utilicen la información presente

Anotación

Ontologías

Los dos almas de la WS

- **La Inteligencia Artificial (IA) > *Ontologías***
- **Las bases de datos (SGBD) > *XML + Metadatos***

Conceptos y Estructura Fundamental

- **La Web semántica se basa en dos puntos fundamentales**
 - La descripción del significado
 - La manipulación automática de estas descripciones
- **La *descripción* del significado se articula con**
 - Semántica
 - MetaDatos
 - Ontologías
- **La *manipulación* se efectúa mediante**
 - Lógica
 - Motores de inferencia

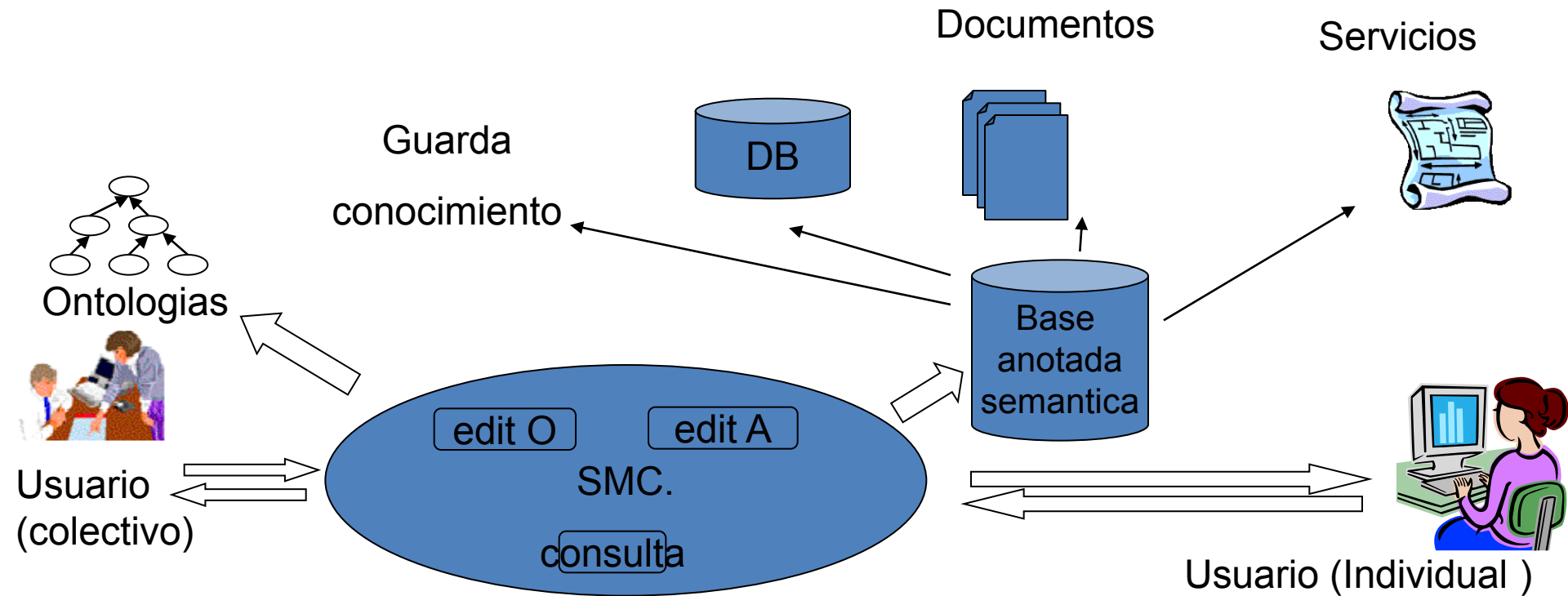
Inspiración

*“The objective of a knowledge management structure is to promote knowledge growth, promote **knowledge communication**, and in general preserve knowledge within the organisation” (Steels L., 93)*

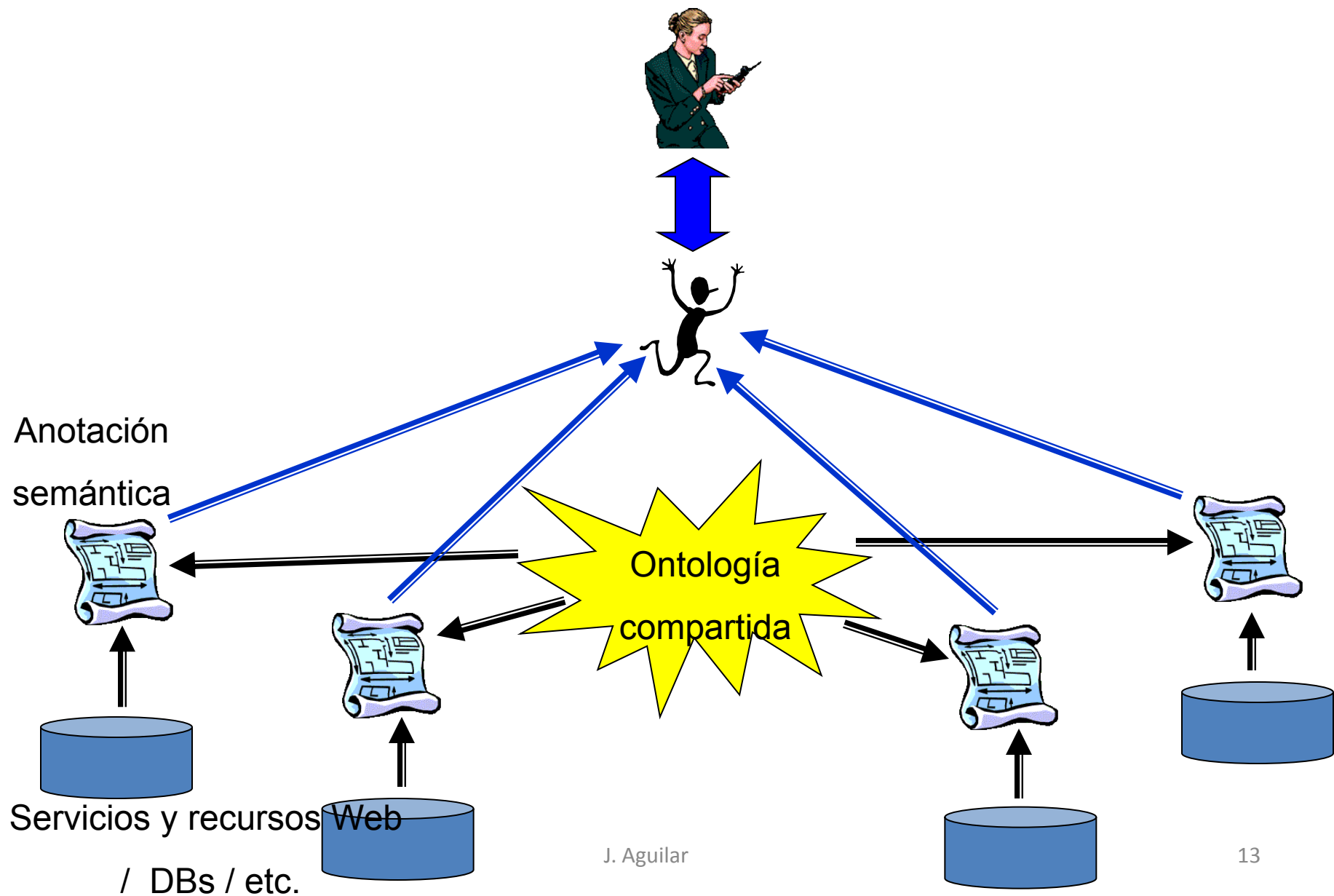
- **Estándares:** RDF, RDFS, OWL, SPARQL, KIF

W3C Standards expressing knowledge

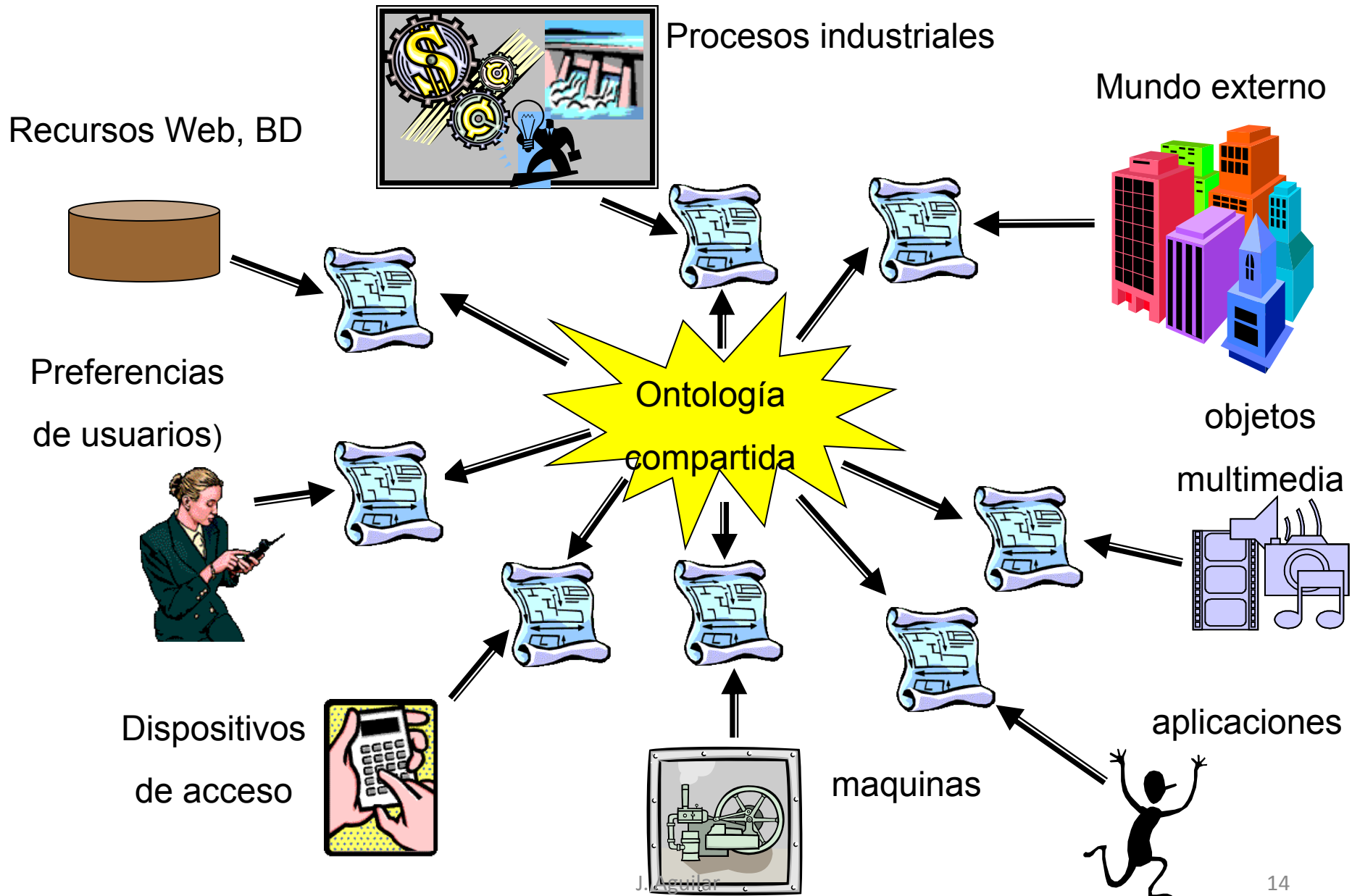
WS



WS: integración



WS: recursos a anotar?



SW implica...

- **Lenguajes formales** como: (DAML+OIL), RDF, OWL, etc.
- **Metadatos** y esquemas de metadatos.
- **Ontologías**: tanto específicas de un dominio, como ontologías de alto nivel.

SW implica...

Descripciones

Semántica

Metadatos

Ontologías

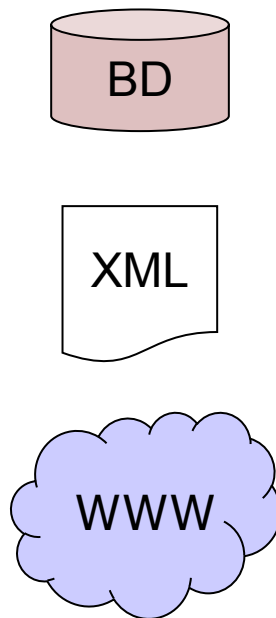
**Procesamiento
automático**

Lógica

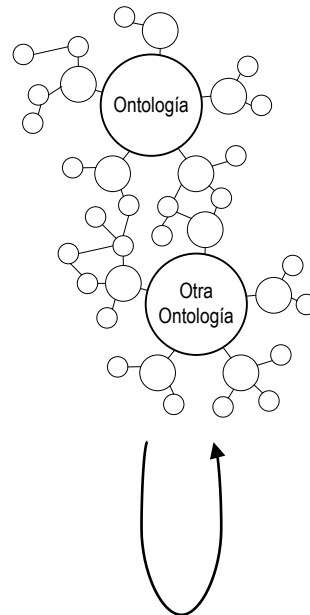
Motores de inferencia
semántica

Ciclo de Vida WS

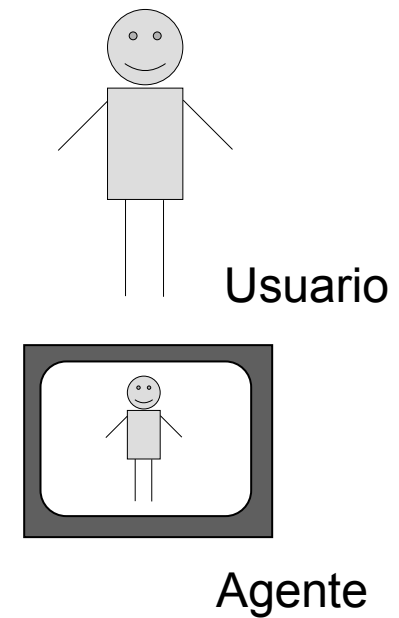
Fuentes de información



Web Semántica

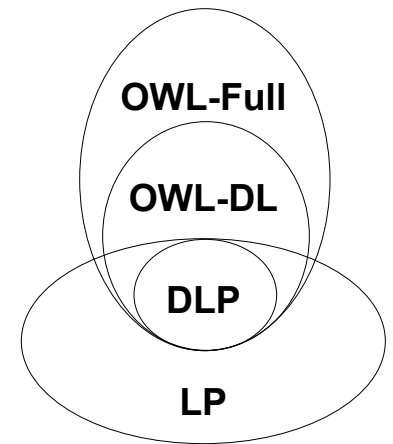


Consumidores de conocimiento



Procesamiento en la WS

- **Semántica explícita:** en descripciones de procesos y objetos
- **Inferencia lógica:**
 - LPO
 - Razonadores (Lógica Descriptiva)
 - Programación lógica
 - Motores de reglas
- **Ontologías:**
 - inferencia + estructura + lingüística + ...
- **Integración de datos**
- **Servicios Web Semánticos**

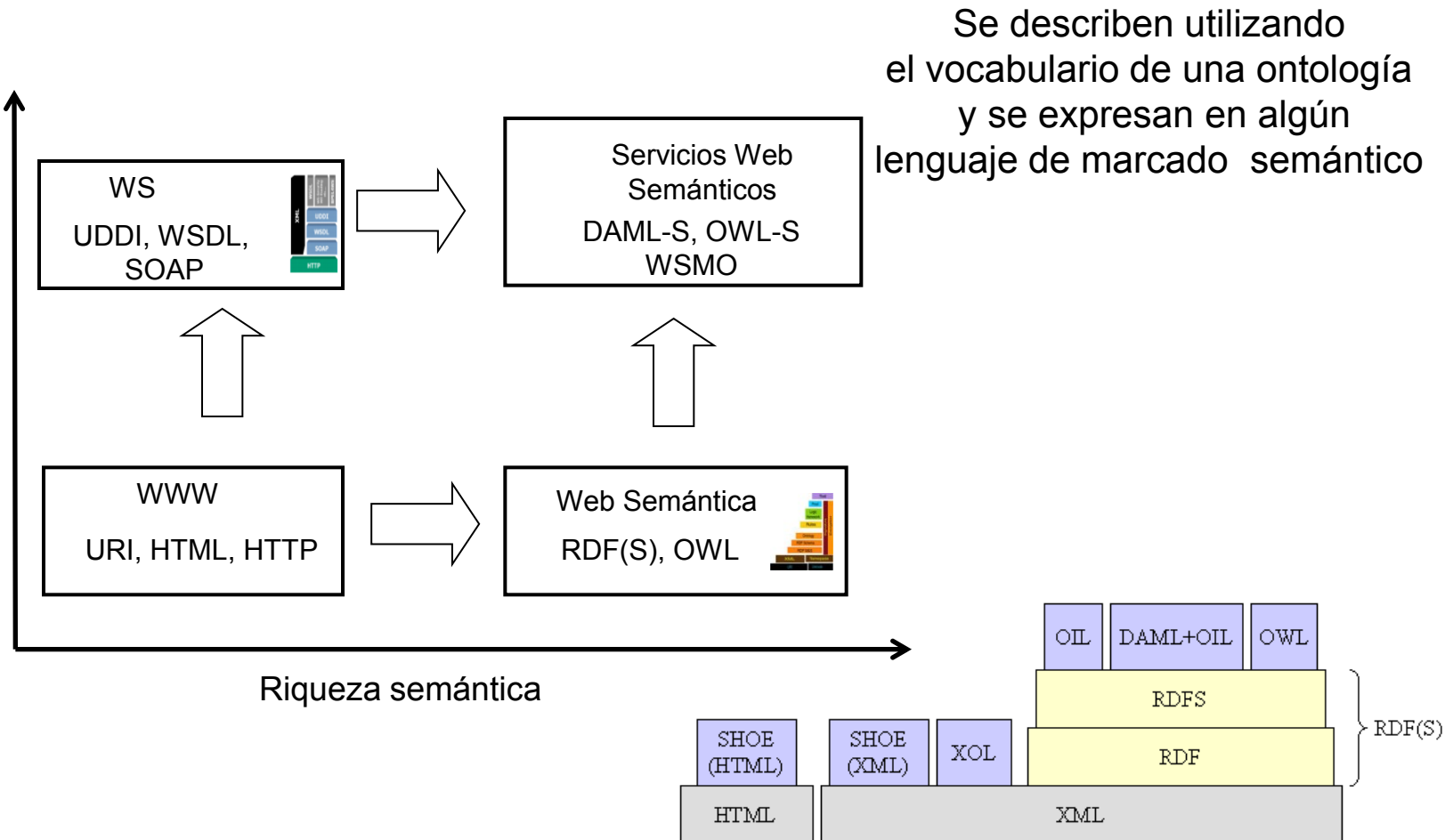


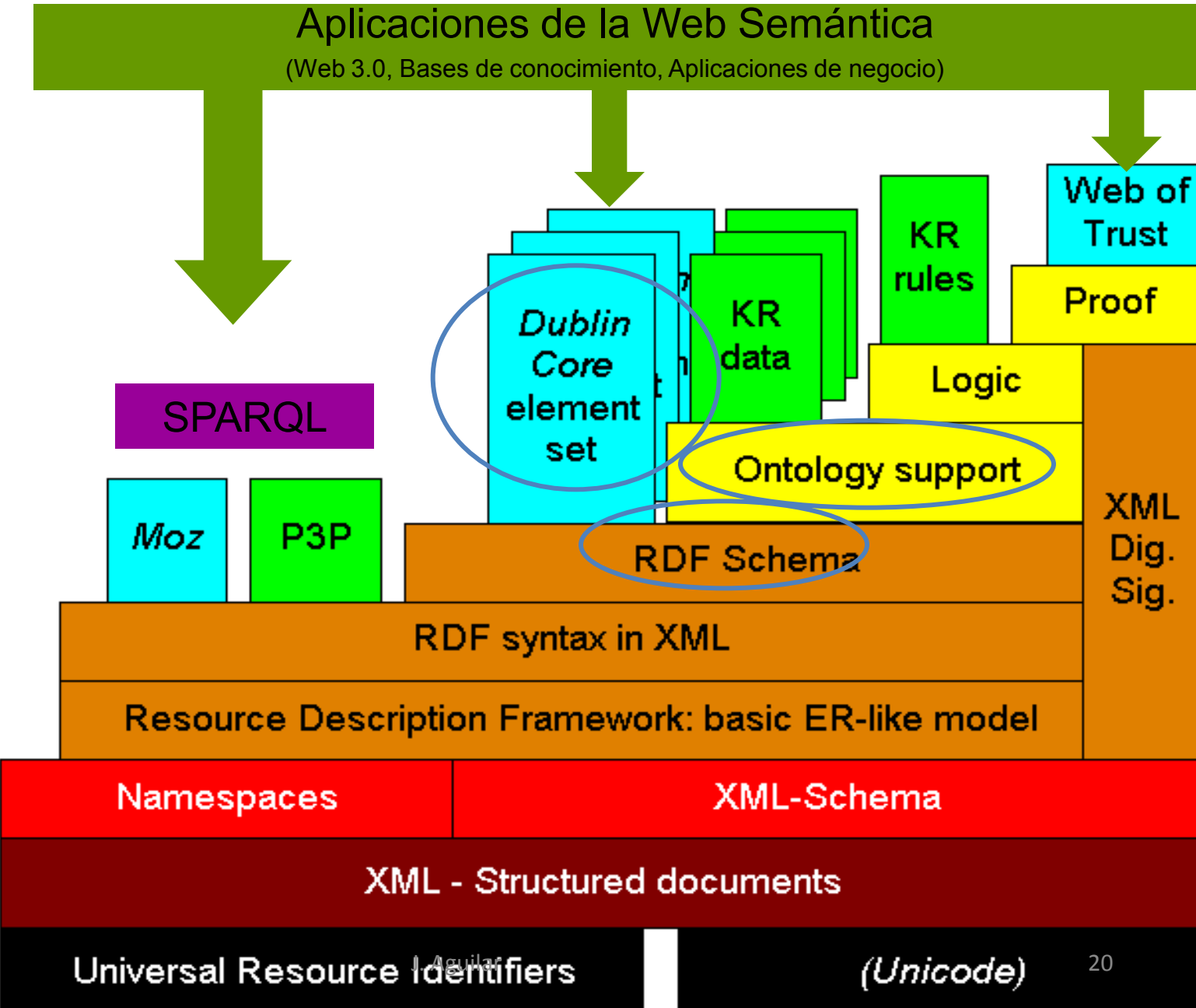
La Web Semántica y Servicios Web Semánticos (SW+WS)

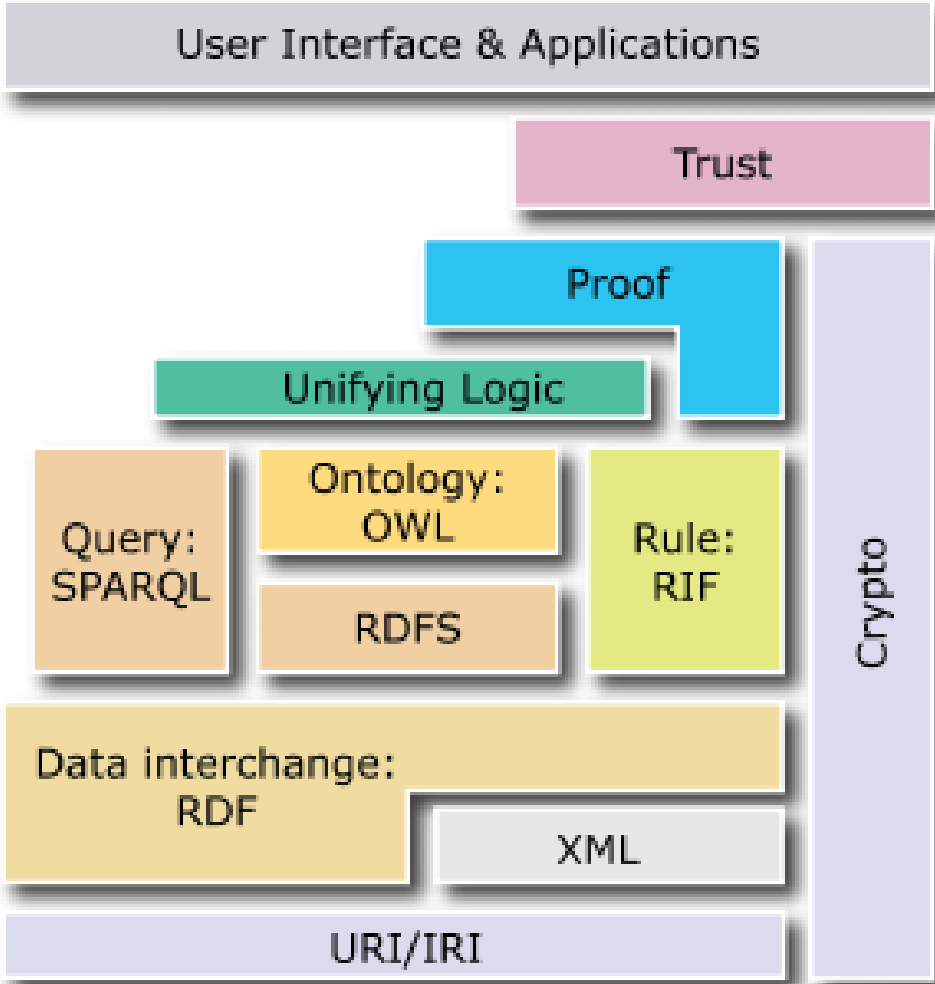


Dinámica

Estática







The Semantic Web Technology Stack (not a piece of cake...)

Most apps use only a subset of the stack

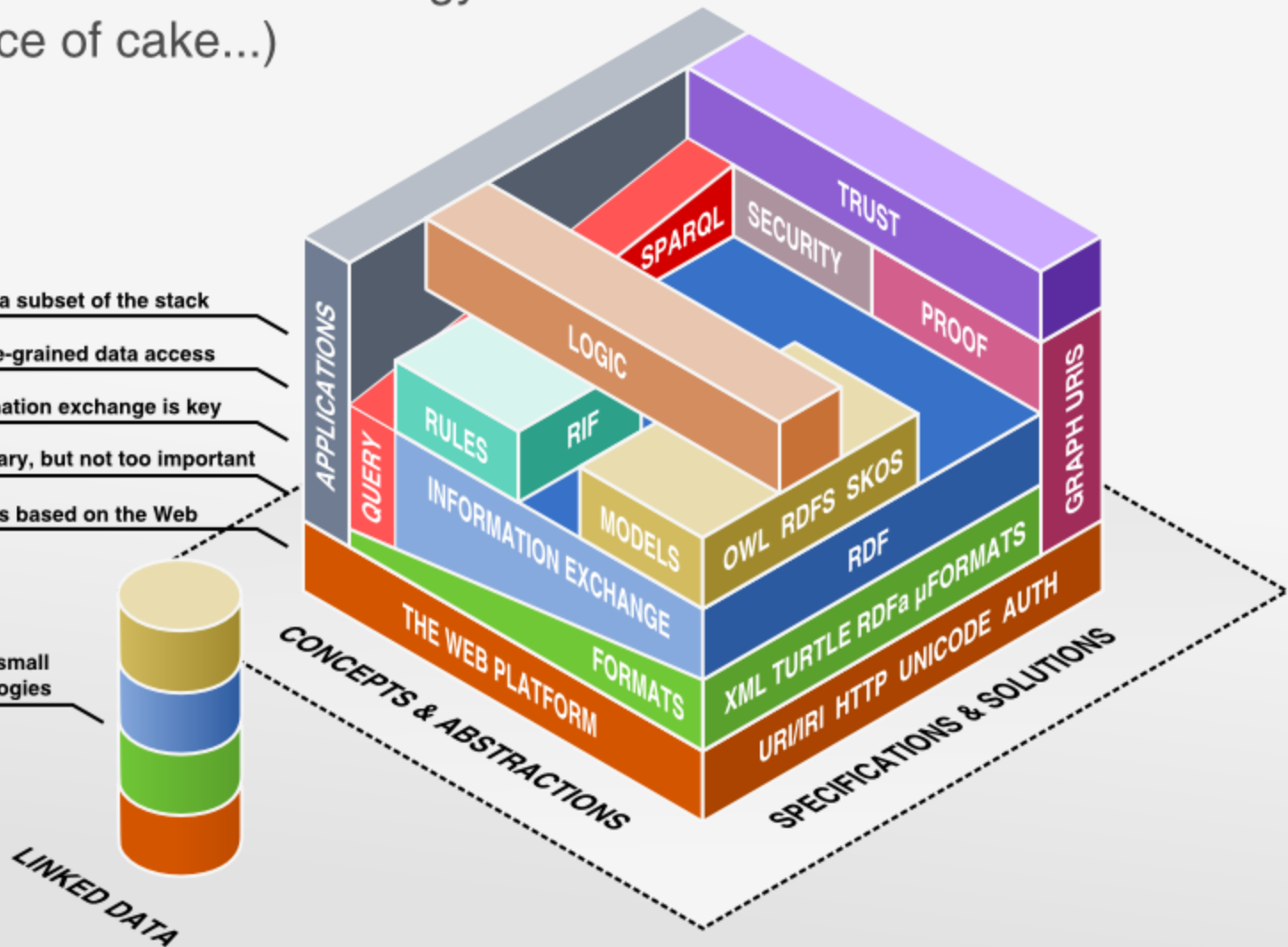
Querying allows fine-grained data access

Standardized information exchange is key

Formats are necessary, but not too important

The Semantic Web is based on the Web

Linked Data uses a small
selection of technologies



Componentes principales de la WS

- XML
- RDF
- OWL
- Agentes de usuario

Ontologías

SW: algo más que XML → Semántica

- **Significados legibles por máquina:** codificar la “semántica de mundo”.
- ***XML (eXtended Markup Language).***
 - Estándar de representación, metalenguaje para el intercambio de datos/información en la web.
 - Desde un punto de vista estrictamente informático: XML no añade semántica.
 - Por ello, necesita además: metadatos y ontologías (semántica).



XML

- XML: lenguaje para definir sentencias con etiquetas semánticamente ricas:

<autor>Umberto Eco</autor>

vs.

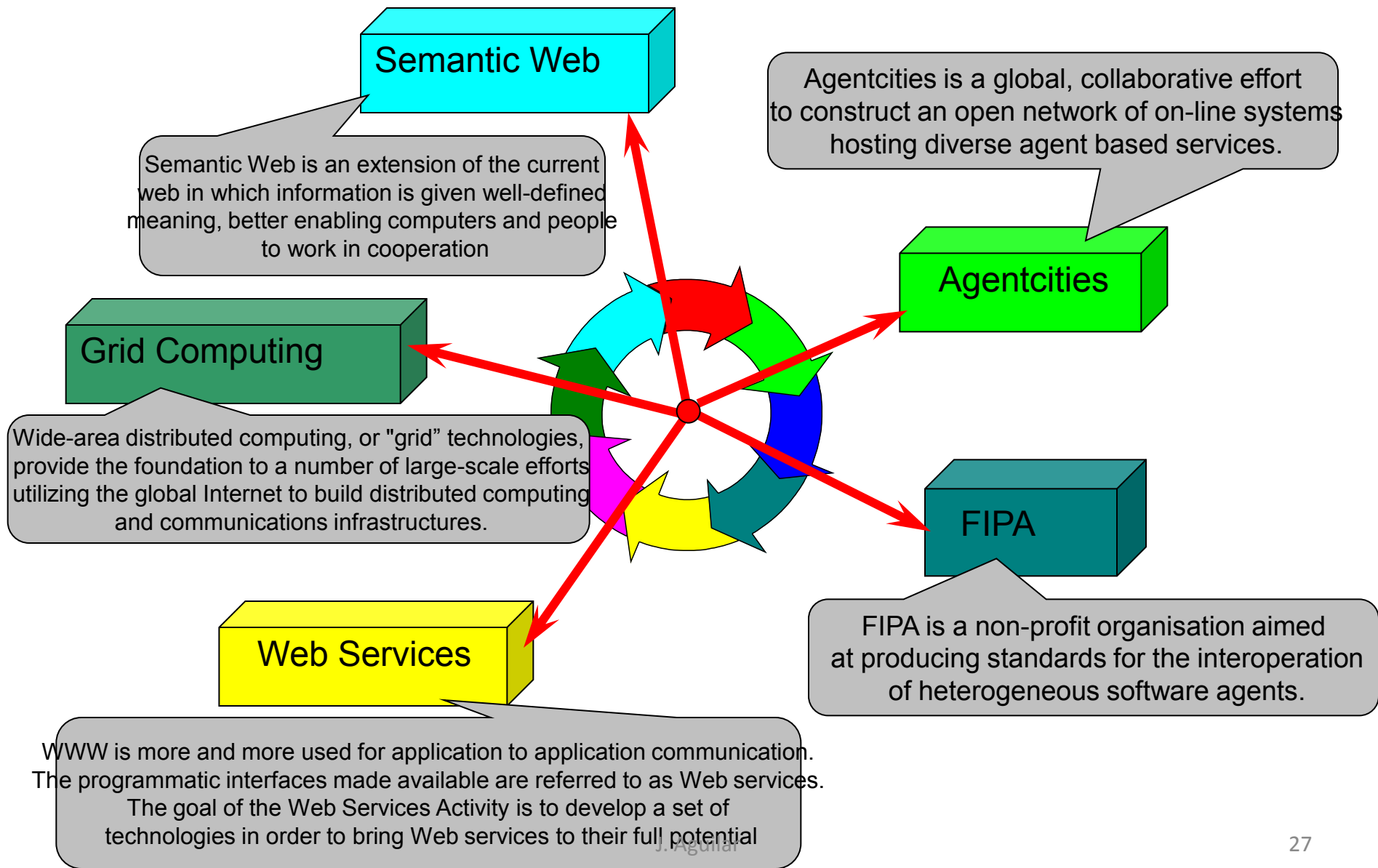
Umberto Eco

- XML Schema:
 - Especificación para asignar tipos de datos, dominios, rangos de valores y restricciones a las etiquetas XML

Aplicaciones XML

- SGBD
- Editores de sitios web
- Navegadores

Actividades en la web



Problemas de los motores de búsqueda basada en palabras clave

- Los resultados son **muy sensibles** a vocabulario
- Participación humana **es necesaria para interpretar y combinar** los resultados
- Los resultados de las búsquedas en Internet **no son fácilmente accesibles** por otras herramientas de software

Significado páginas Web

- El significado de las páginas Web **no es fácilmente accesible a máquinas**: falta de semántica
- **Es difícil distinguir el significado** entre estos dos frases:
Soy un profesor de ciencias de la computación.
usted puede pensar que soy un profesor de ciencias de la computación . . .
- **Procesamiento del lenguaje natural no puede ser la respuesta** en una escala tan grande

WS

- Representar un contenido en la Web en una forma que es **más fácilmente procesable por máquinas.**
- Utilizar técnicas inteligentes para **aprovechar estas representaciones.**
- La Web Semántica evolucionará gradualmente de la web existente

Evolucionar hacia una Web de conocimiento

Ejemplo: reservación automática de vuelo

- Su sistema de reserva de vuelos automática **sabe** acerca de sus preferencias
- **Construye** la base de conocimientos mediante su pasado
- **Combina** el conocimiento local con servicios remotos:
 - preferencias de aerolíneas
 - necesidades dietéticas
 - calendario , etc.
- **Se comunica** con la información a distancia (es decir, en la Web!)

Hacia la WS

- Las tareas a menudo requieren **combinar los datos en la Web:**

información del hotel y los viajes, que pueden provenir de diferentes sitios

- Los seres humanos **combinan esta información con facilidad**, incluso si se utilizan diferentes terminologías, la información es incompleta, o enterrados en las imágenes, videos, ...

Ejemplo: redes sociales

- Los sitios sociales **están en todas partes** estos días (LinkedIn, Facebook, Dopplr, Digg, Plexo, Zyb, ...)
- **Los datos no son intercambiables**: ¿cuántas veces usted tiene que añadir sus contactos?
- **Las aplicaciones deben ser capaces de llegar a esos datos** a través de medios convencionales, por supuesto, nos son cuestiones de privacidad ...

Qué se requiere?

- Los datos deben **estar disponibles** para las máquinas
- Los datos deben ser eventualmente **combinados**, se fusionan en una escala Web
- Las máquinas también pueden **tener que razonar** sobre esos datos

Crear una web de datos (más allá de la Web de Documentos)

El impacto de la Web Semántica - Integración de Datos

- Algunas aplicaciones requieren la gestión de varias bases de datos (heterogéneo). Por ejemplo
 - después de fusiones de compañías
 - Grupos de empresas con actividades similares (por ejemplo, de reservas de hoteles)
 - combinación de datos: e-Gobierno
datos científicos (por ejemplo, en la bioinformática)

La mayor parte de los datos públicos está disponible en la Web

Integración de Datos

1. Mapa de los distintos datos en una representación de datos abstractos
2. Hacer los datos independiente de su representación interna ...
3. Combinar las representaciones resultantes
4. Comience a hacer preguntas sobre el todo!

**consultas que no se podrían haber hecho en los
conjuntos de datos individuales**

Integración de Datos

- ¿Qué se necesita?
 - Datos disponibles para procesar
 - Datos posibles a gran escala para combinar (web)
 - Mecanismos de razonamiento sobre los datos

Web de Datos

Ejemplo de integración

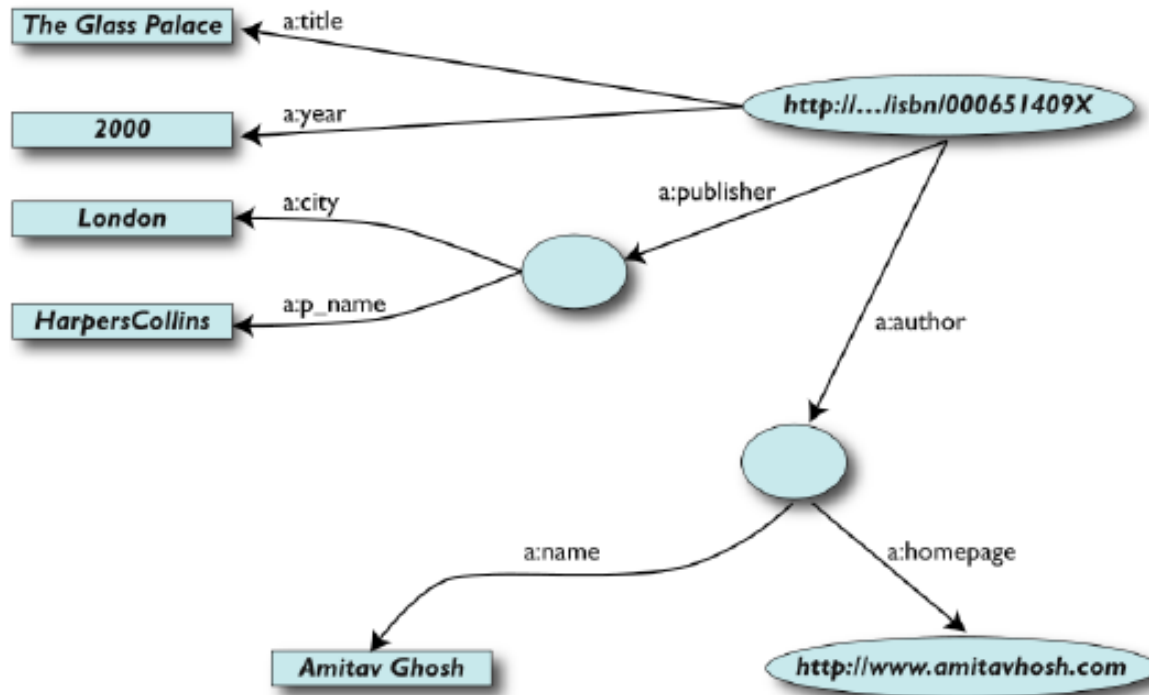
- Datos de una tienda de libros

ID	Author	Title	Publisher	Year
ISBN 0-00-651409-X	id_xyz	The Glass Palace	id_qpr	2000

ID	Name	Home page
id_xyz	Amitav Ghosh	http://www.amitavghosh.com/

ID	Publisher Name	City
id_qpr	Harper Collins	London

Ejemplo de integración

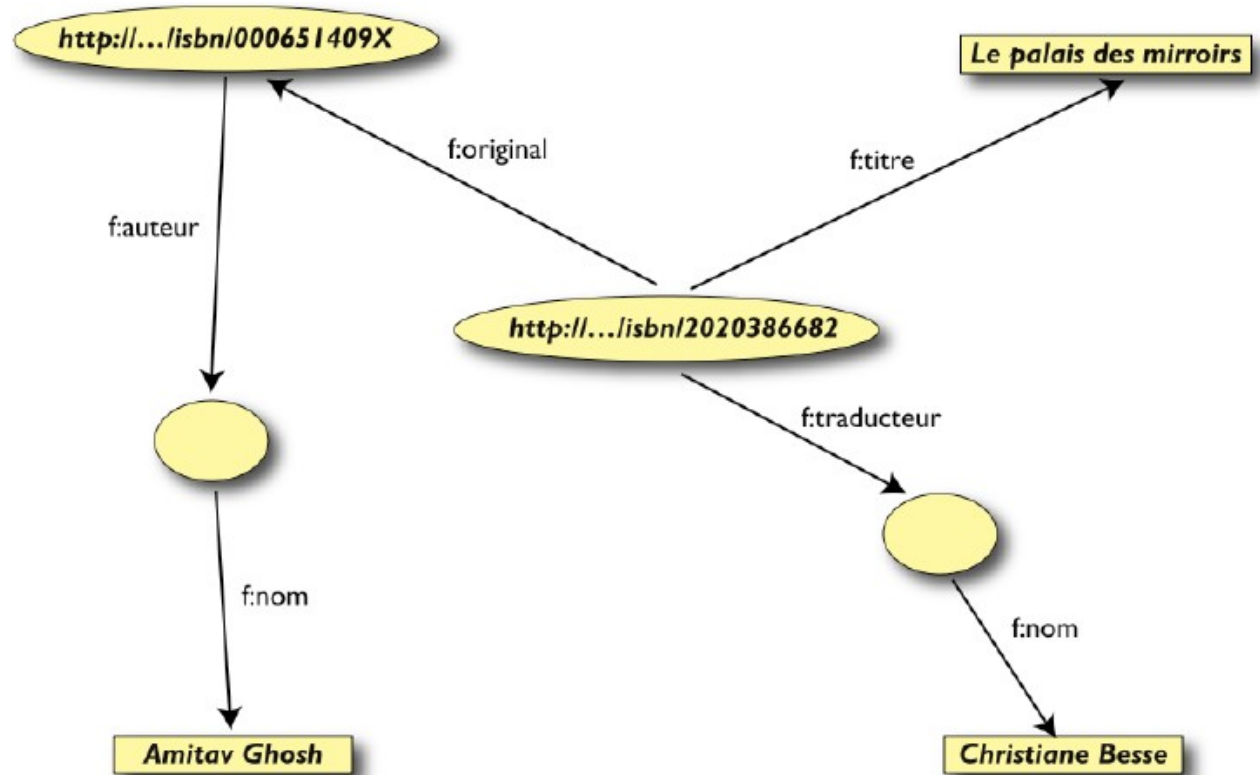


Ejemplo de integración

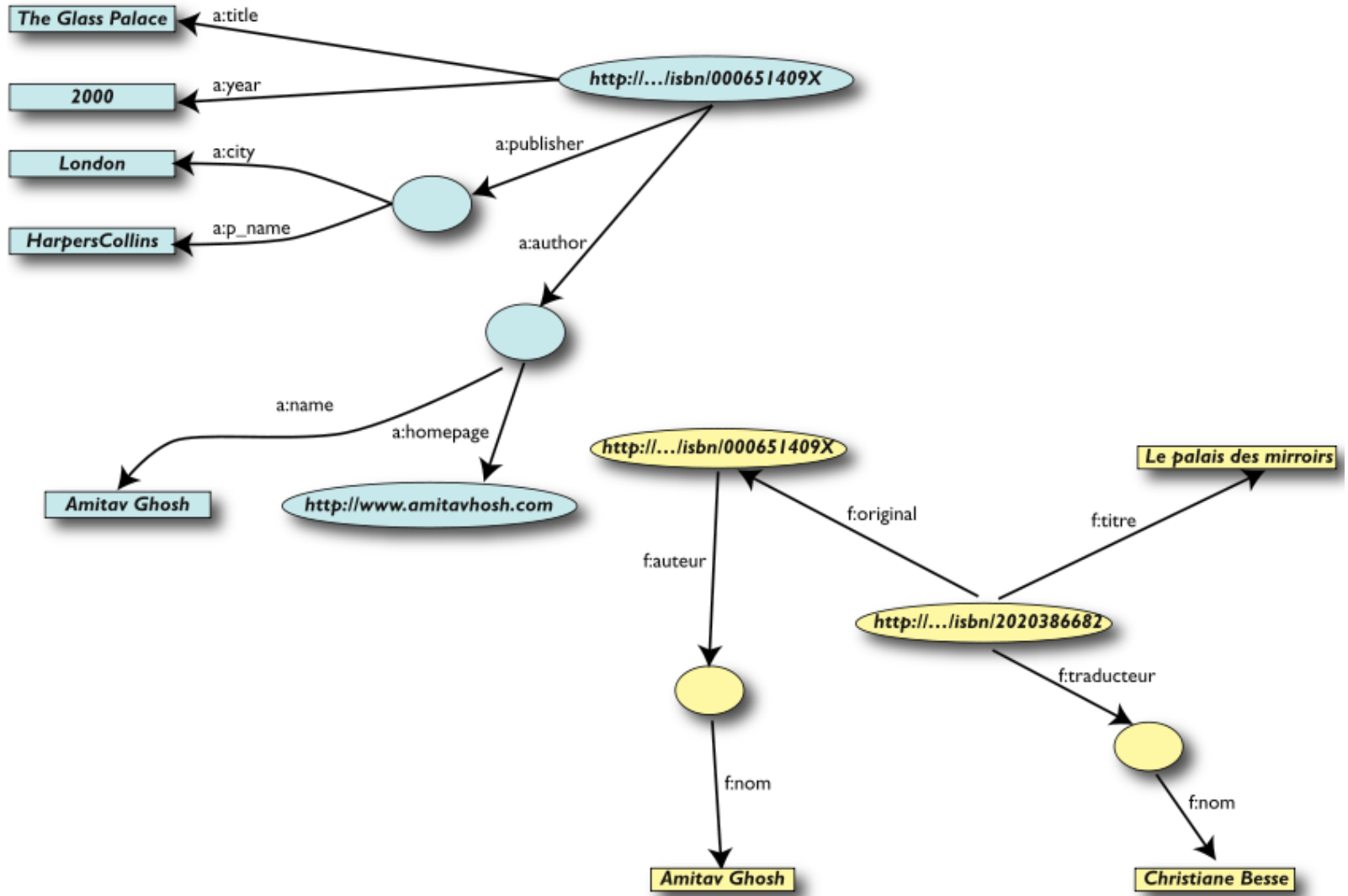
- Datos en otra tienda

ID	Titre	Auteur	Traducteur	Original
ISBN 2020386682	Le Palais des miroirs	i_abc	i_qrs	ISBN 0-00-651409-X

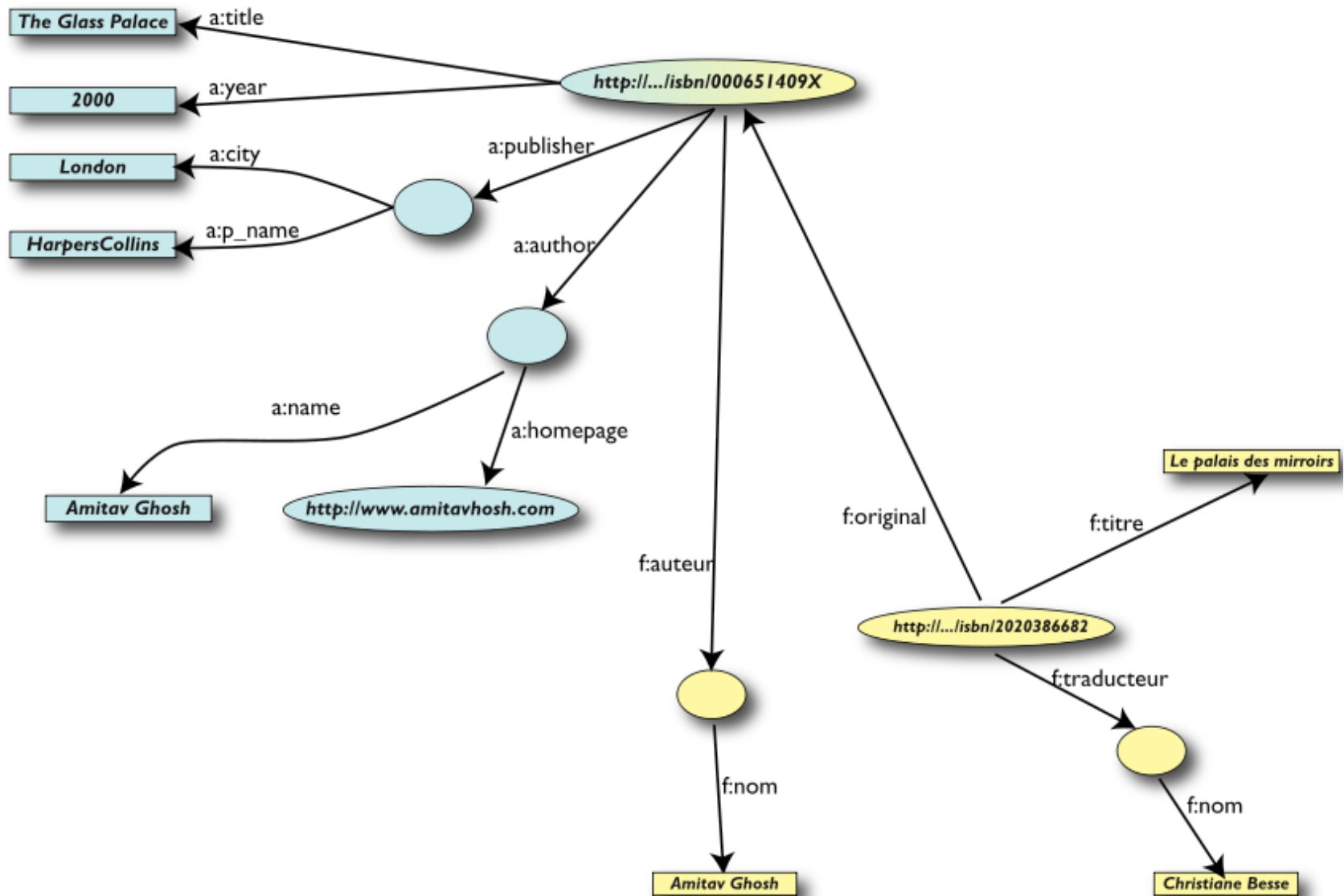
ID	Nom
i_abc	Amitav Ghosh
i_qrs	Christiane Besse



Ejemplo de integración



Ejemplo de integración



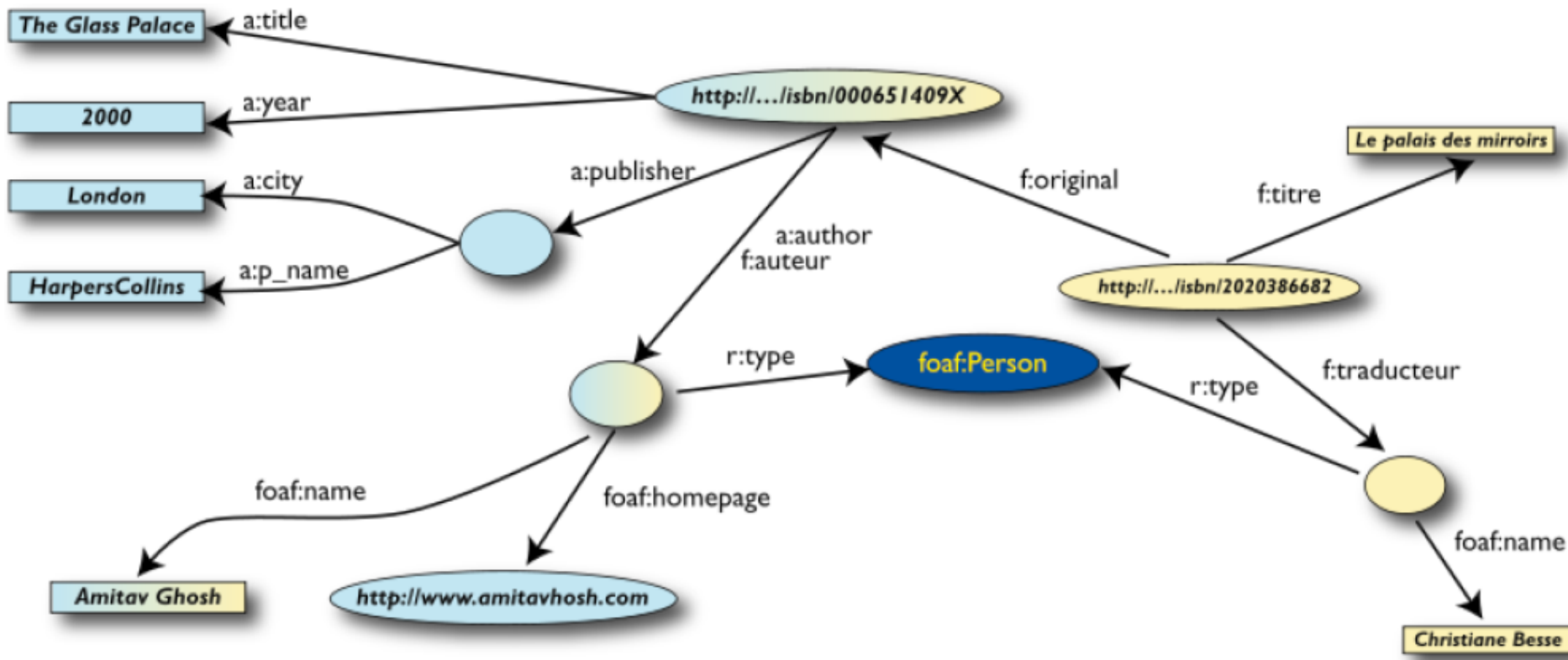
Ejemplo de integración

¿Qué pasaría si tuviéramos conocimiento adicional?

- por ejemplo, podríamos saber que a: el autor es el mismo que f: auteur
- Podríamos saber que en ambos casos representan personas

**podríamos tener ontologías que describen los
conceptos y relaciones**

Ejemplo de integración

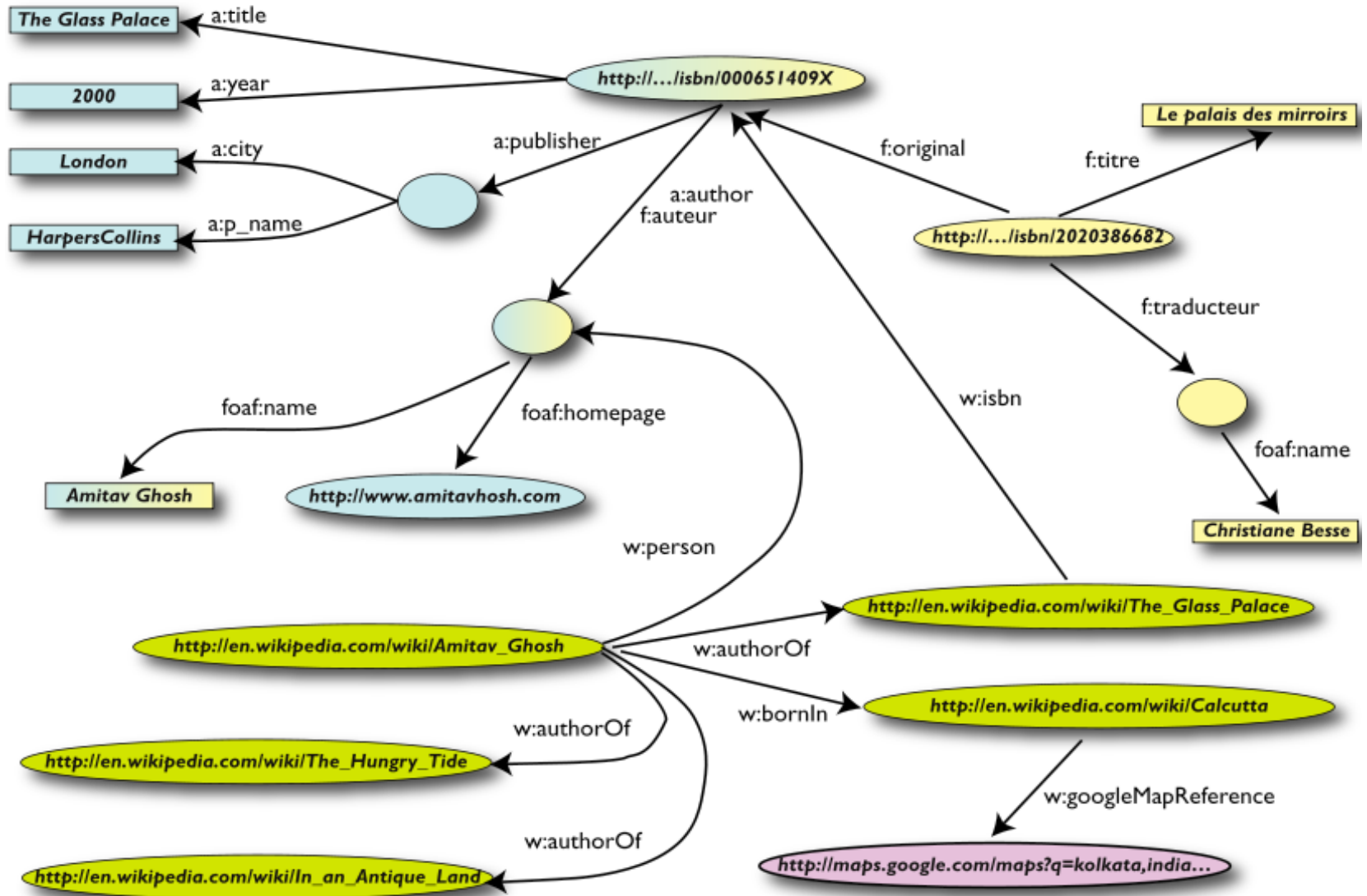


Ejemplo de integración

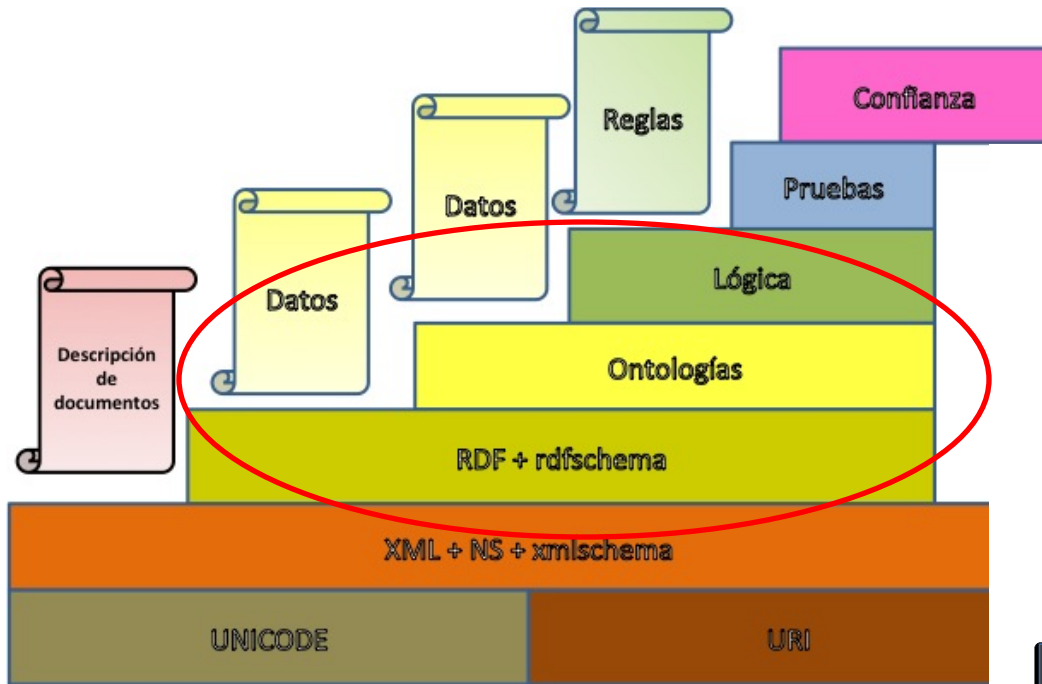
Con esta información resultante de la fusión, consultas más ricas son posibles:

- La segunda librería puede consultar cuál es la página web de la autora del libro
- Posiblemente puede preguntar acerca de otra información sobre el autor, o un traductor
- La primera librería puede consultar la información sobre el traductor

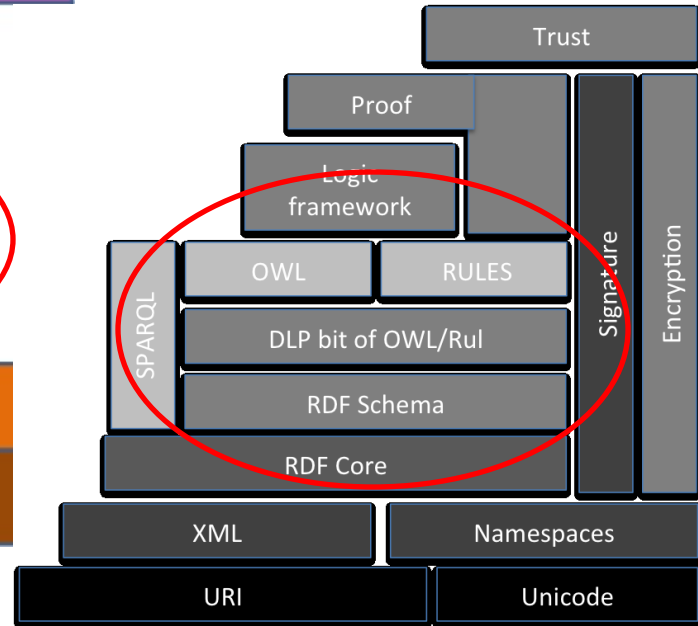
Web de Datos



Web Semántica



Arquitectura de la Web Semántica. (adaptada de [Fuente: <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide11-0.html>])



Arquitectura alternativa de la Web Semántica. [Antoniou, Harmelen]

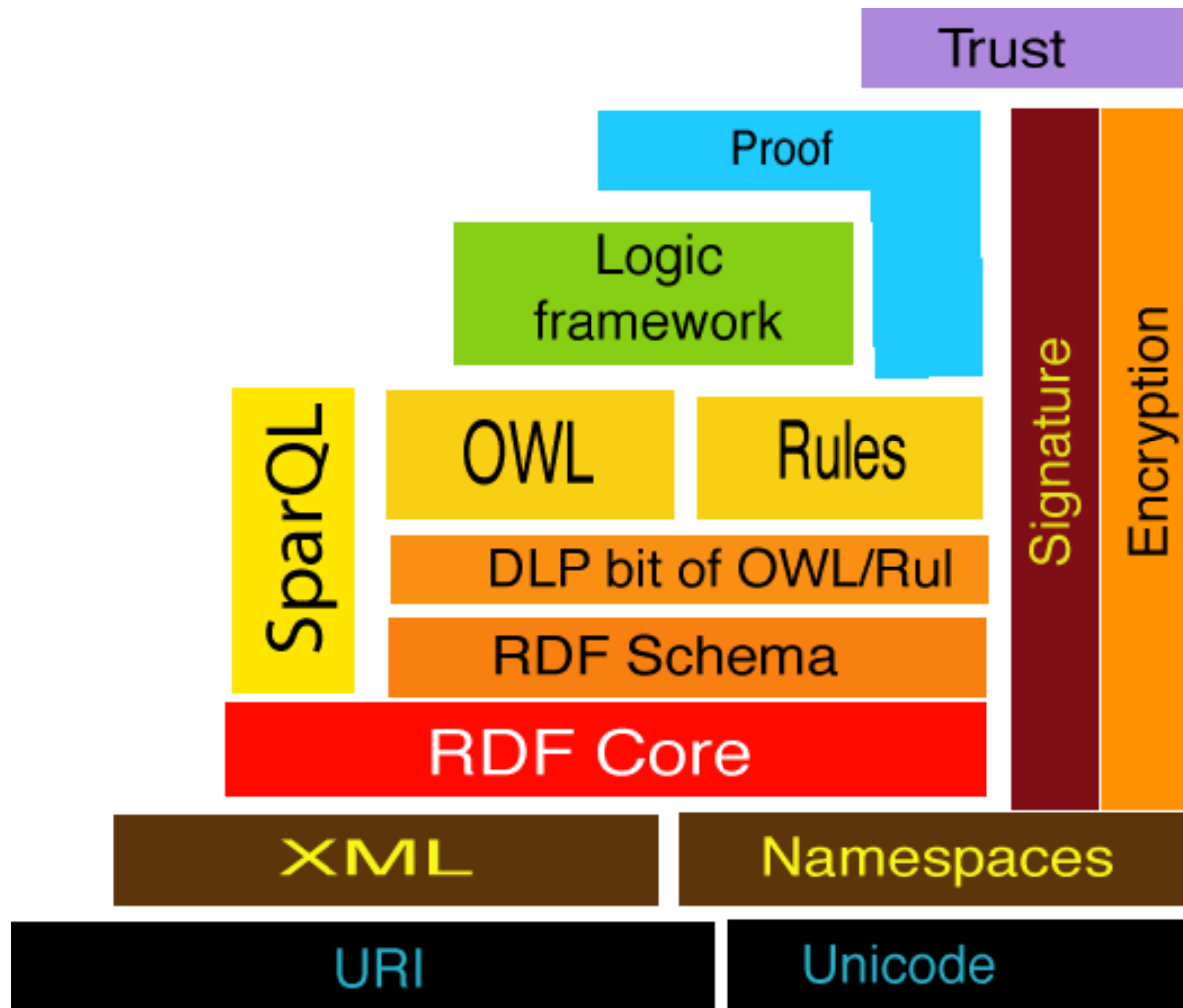
Con los lenguajes se pueden realizar:

- ✓ Anotaciones
- ✓ Integraciones
- ✓ Inferencias
- ✓ **Compartir (Interoperabilidad)**



J. Aguilan

Web Semántica



WS

- **Capa XML**
 - Base sintáctica de datos estructurados
- **Capa RDF**
 - RDF modelo de datos básico para hechos
 - Esquema RDF lenguaje simple ontología
- **Capa Ontológica**
 - Más expresivo que el Esquema RDF
 - Estándar Web actual: OWL

WS

- **Capa lógica**
 - mejora aún más lenguajes de ontologías,
 - conocimiento declarativo específico de la aplicación
- **Capa de prueba**
 - pruebas, validación
- **Capa de confianza**
 - firmas digitales, recomendaciones, agencias de calificación

Lenguajes de Representación

- Para describir la semántica se requiere un lenguaje apropiado (llamado lenguaje de representación)
- Tienden a estar basados en XML
- Ejemplos:
 - OML (Ontology Markup Language)
 - XOL (Ontology Exchange Language)
 - SHOE una extensión de HTML
 - RDF y RDFS impulsados por el W3C consortium
 - Mapas Temáticos (Topic Maps) estándar ISO
- RDF y Topic Maps son los más comunes

Web de datos: HTML

<h1> Bibliography </h1>

<p> <i> Foundations of Databases </i>

Abiteboul, Hull, Vianu

 Addison Wesley, 1995

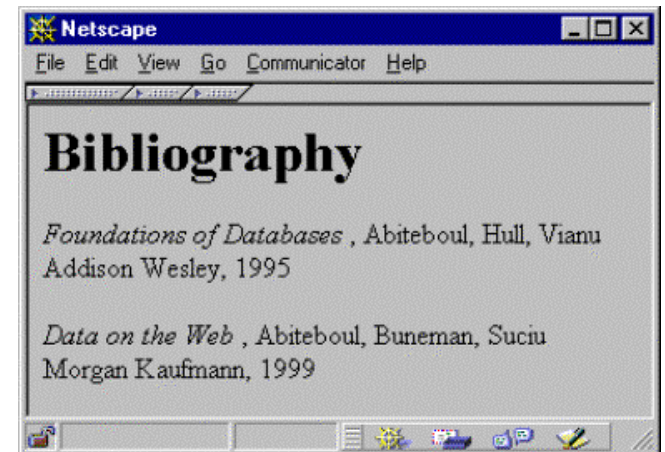
<p> <i> Data on the Web </i>

Abiteoul, Buneman, Suciu

 Morgan Kaufmann, 1999

¡lenguaje de terror!...pero todos tenemos alguna virtud:

HTML es fácil de usar, fácil de usar, fácil de usar,... y robusto.



HTML es limitado

- Orientado a **visualización** de datos
- No permite describir datos
- No es extensible: congela pequeños pedazos de etiquetas
- No describe “contenido” (objetos)

Web de datos: XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<curso>
```

```
  <profesor> Claudio Gutierrez </profesor>
```

```
  <nombre> Bases de Datos </nombre>
```

```
  <semestre anho="2005"> Primavera </semestre>
```

```
  <horario>
```

```
    <modulo tipo="catedra"> 2.1 </modulo>
```

```
    <modulo tipo="catedra"> 3.1 </modulo>
```

```
    <modulo tipo="auxiliar"> 1.5 </modulo>
```

```
  </horario>
```

```
</curso>
```

Web de datos: XML

- Posibilidad de crear *nuevos vocabularios*
- *Universalidad* via espacios de nombres
- *Modularidad* via espacios de nombres y tipos de datos
- Versatilidad via *datos semi-estructurados*
- *Sólidos fundamentos* via SGML
- Amplio soporte y distribución
- Estándar de interoperabilidad documental

se requiere más...

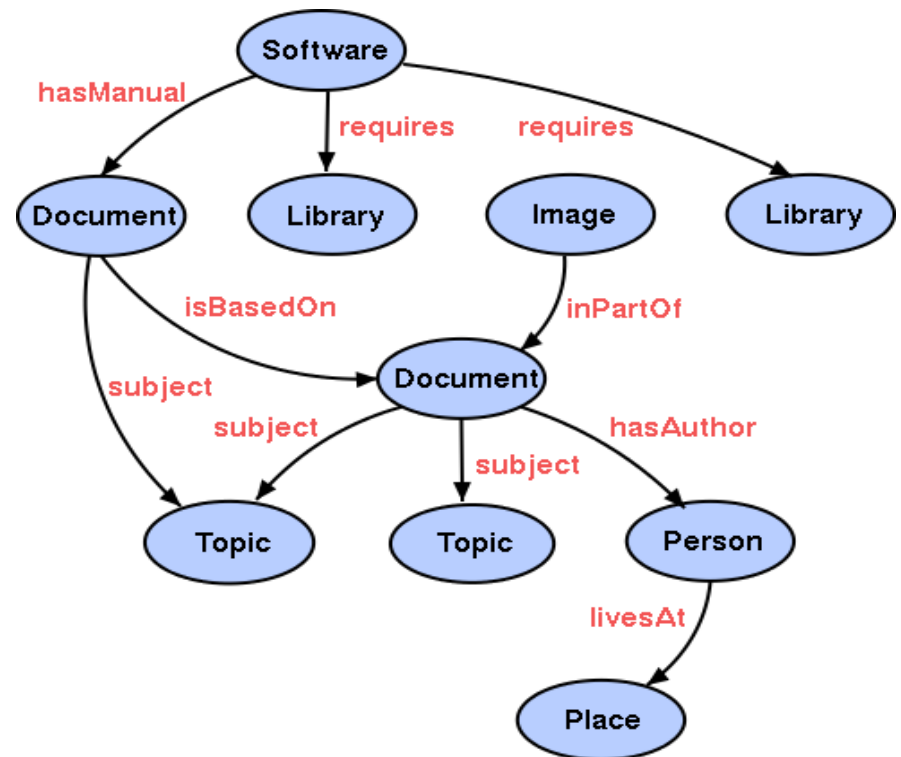
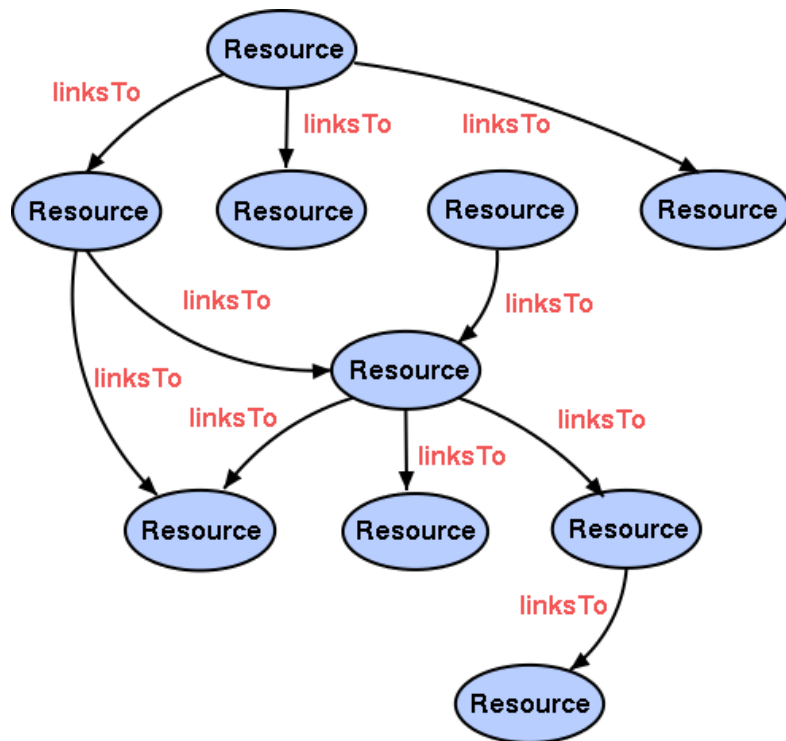
Web de información

un modelo común tiene que ser proporcionado a las máquinas para describir, consulta, etc., los datos y sus conexiones

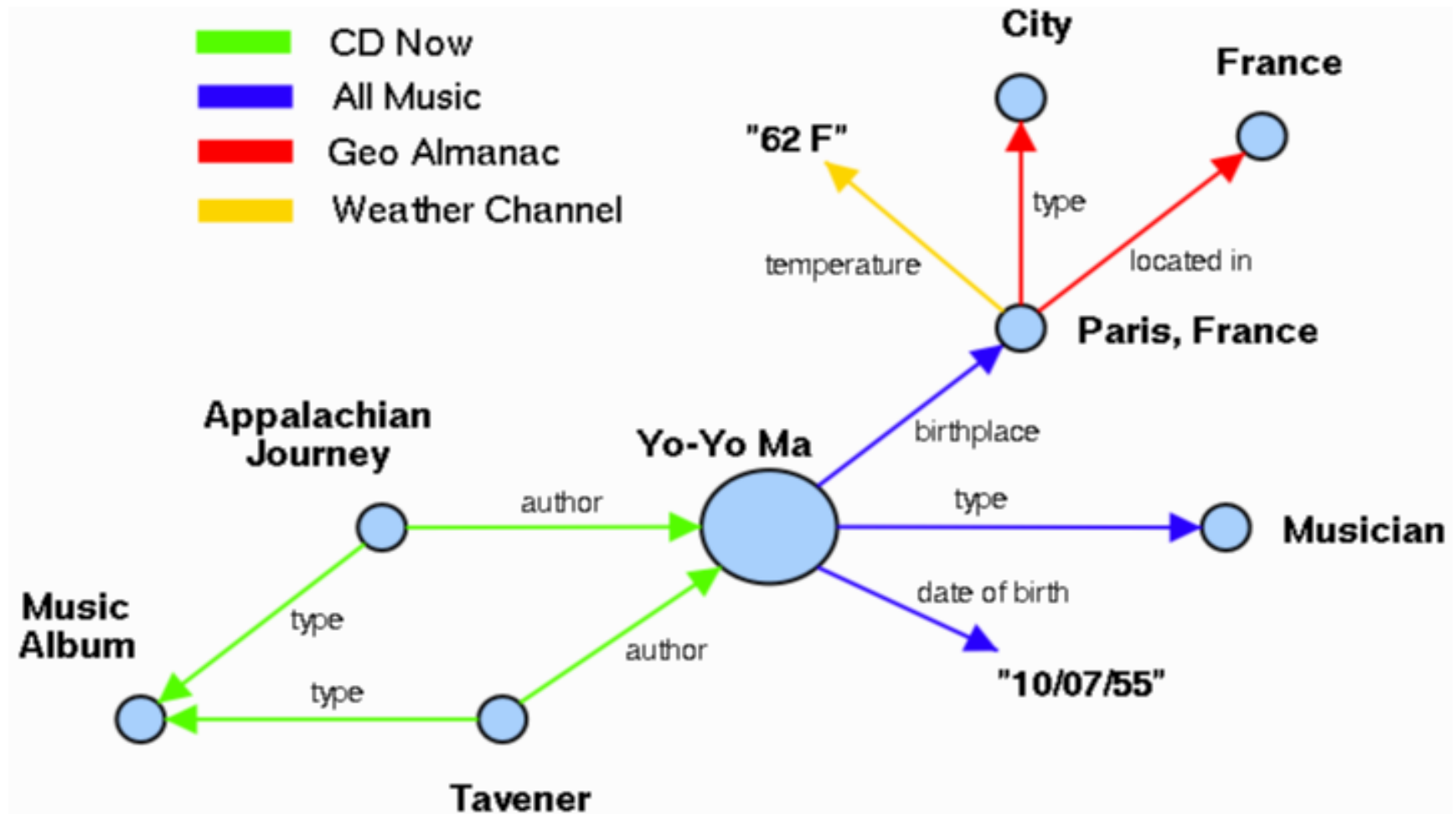
la "clasificación" de los términos puede llegar a ser muy complejo para áreas específicas del conocimiento: aquí es donde las ontologías, tesauros, etc., entran en el juego



Web de datos vs. Web de información



Web de información



Modelo de datos: Requerimientos

1. **Todo es un recurso:** páginas web, bases de datos, ampolleta del patio, dirección de la empresa, etc.
2. **Identificadores únicos:** cada recurso tiene un único identificador
3. **Vocabularios compartidos:** predicados deben ser comunes para todos en un área
4. **Estructura extensible y distribuída:** todos pueden agregar su información y nadie debe ser dueño de toda

Modelo de datos: XML no es suficiente

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<rukangma>
```

```
  <chaw> Aukan </chaw>
```

```
  <nguke> Peyeché </nguke>
```

```
  <yall>
```

```
    <pichi s="wentru"> mari </pichi>
```

```
    <pichi s="wentru"> epu </pichi>
```

```
    <pichi tipo="domo"> regle </pichi>
```

```
  </yall>
```

```
</rukangma>
```

Modelo de datos: XML no es suficiente

- Modelo es un árbol (no un grafo)
- Hijos están ordenados
- Esquemas no permiten extensibilidad distribuída

pero sobretodo:

XML modela documentos, y el mundo real no es un documento, sino una red de relaciones

(aunque no olvidemos que hay muchos documentos)

Principales estándares

- **SPARQL**: *Simple Protocol and RDF Query Language* (protocolo y lenguaje de interrogación para RDF)
- **RIF**: Rule Interchange Format
- **GRDDL**: *Gleaning Resource Descriptions from Dialects of Languages*, para transformar XML y XHTML a RDF
- **SAWSDL**: *Semantic Annotations for Web Service Definitions* (WSDL) típicamente utilizado en modelizaciones RDF y OWL.
- **RDFa**: *RDF Attributes*, para insertar datos estructurados en RDF páginas XHTML

Modelo de datos: la solución

Resource Description Framework (RDF)

La información es un **grafo dirigido etiquetado** que modela las **relaciones entre** objetos

- **Nodos**: recursos
- **Arcos**: propiedades

Las oraciones tienen de la forma:



RDF

- [RDF](#) es un estándar propuesto por el [W3C](#) para definir la arquitectura necesaria para soportar metadatos (descripciones) sobre recursos web. Es la aplicación XML recomendada por el consorcio para codificar, intercambiar y reutilizar metadatos estructurados.
- [RDF](#) es una base para procesar metadatos; proporciona interoperabilidad entre aplicaciones que intercambian información legible por máquina en la Web.



RDF: Definición

- RDF destaca por la facilidad para permitir un procesamiento legible por máquina de representaciones de recursos Web.
- RDF hereda todos los conceptos de XML que permiten especificar propiedades de los elementos (por ejemplo, los espacios de nombre o namespaces).

Modelo de datos: propiedades de RDF

- **Expresividad**: fragmento conjuntivo de la lógica de primer orden

$$\exists, \wedge, p(X, Y), c_1, c_2, \dots$$

- **Complejidad**: tratable bajo condiciones “razonables”
- **Soporte teórico**: modelo de grafos en Bases de Datos, Redes semánticas

Metadatos

- Son datos que describen otros datos
- En este contexto
 - Datos que describen recursos de la Web.
- La distinción entre datos y metadatos es relativa
 - Depende de la aplicación.
 - Les metadatos de una aplicación pueden ser los datos que maneja otra aplicación.

RDF - Metadatos

- RDF: Un sistema de descripción de entidades (recursos) con una base lógico/lingüística
- RDF relaciona recursos con propiedades y valores
- Puede considerarse un sistema de expresión de metadatos

Metadatos

- Ejemplos "tradicionales"
 - Las recetas de la estructura de base de datos y el contenido.
 - Nombres y permisos en un sistema de archivos de archivos.
- Los metadatos pueden ser consultados para conocer las características de los datos que describe.
- Sistemas de metadatos tradicionales están fuertemente acoplados funcionalmente a los datos que describen.
- Prescriptiva, necesaria para interactuar directamente con los datos.

Metadatos y Web

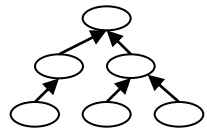
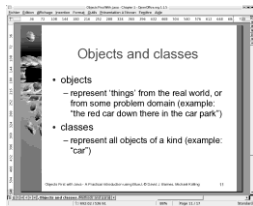
- Conceptos de metadatos tradicionales deben ampliarse cuando los sistemas se vuelven más distribuidos, porque la información se vuelve más amplio
- Metadatos utilizado para la información, se vuelve descriptivo.
- Los metadatos pueden necesitar para describir los recursos, no sólo datos.
- Todo es un recurso : gente, computadoras, software, presentaciones en congresos, conferencias, actividades, proyectos.

Anotación



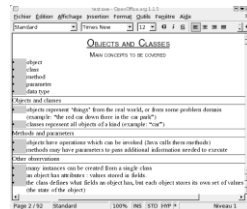
- Expresar conocimiento adicional
 - Con base en la experiencia y la visión del usuario
- 3 pasos
- Pre-procesamiento
 - Manual de anotación
 - extracción automática

recurso a
user (XML)



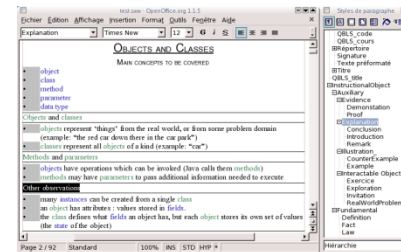
pre-procesar

“anotable”
versión



anotación

anotada
versión



transformar.

content
(XHTML)

annotation
(RDF)

Ontologías
(OWL, RDFS)

Preprocesar

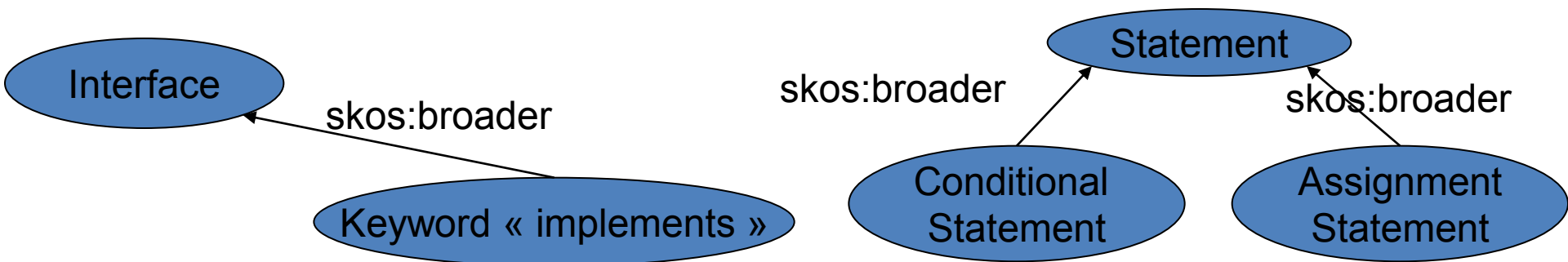


- Identificación características del contenido
- Separación de las entidades pequeñas
- anotación automática
- Vocabulario utilizado → conceptos de dominio,
- la anotación automática con la ontología de dominio
- Funciones de Recursos → ontología

Anotar Manualmente

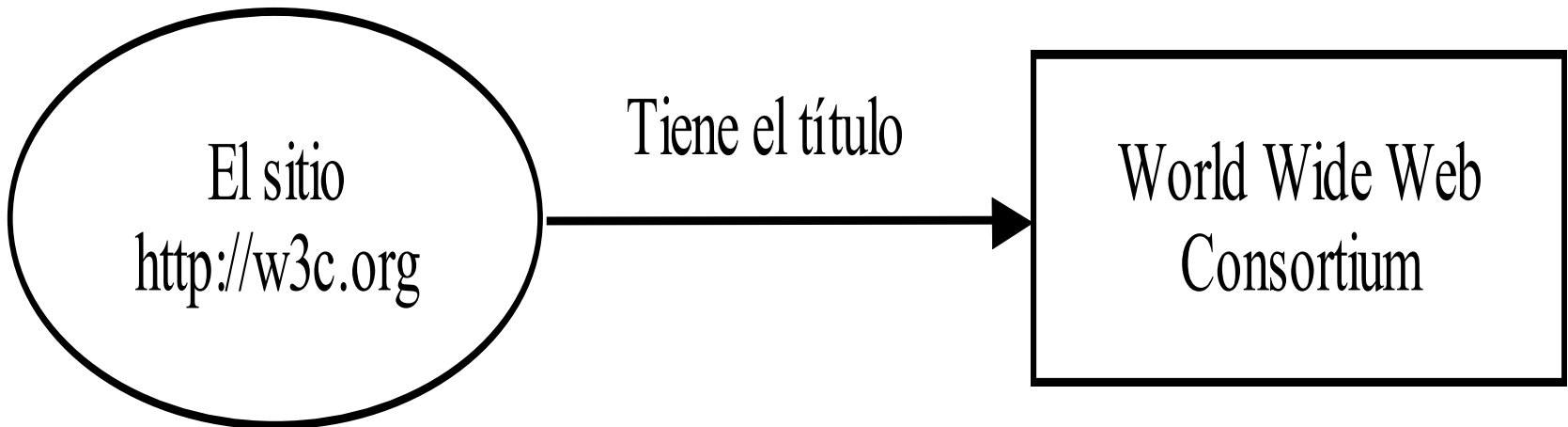


- Explotación de herramientas
- Evolución / enriquecimiento / creación de la ontología de dominio correspondiente
- Caminos de conexión de navegación:



Ejemplo RDF

En modo nativo (gráfico):



En modo serializado (RDF/XML)

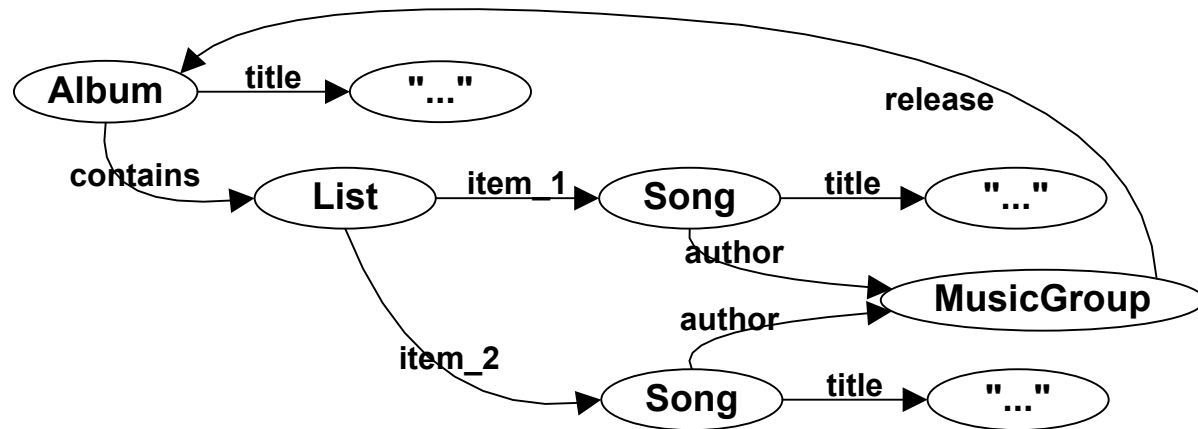
...

```
<rdf:Description rdf:about="http://www.w3.org/">  
  dc:title>World Wide Web Consortium</dc:title>  
</rdf:Description>
```

...

Principio: Modelo Grafo

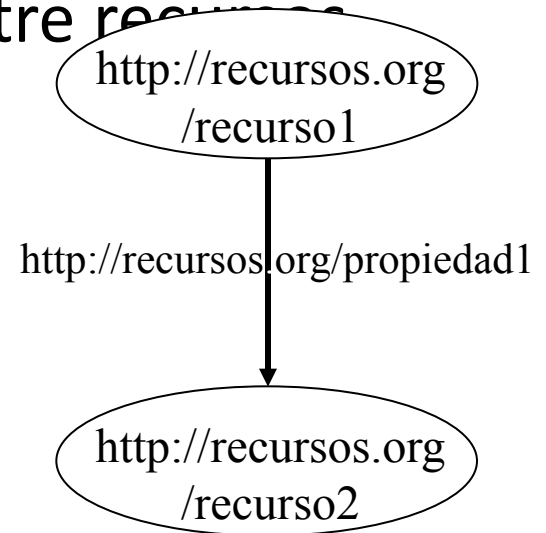
- **Ejemplo grafo RDF:**



- **Grafo puede modelar árbol, tabla, ...**

Principio: Modelo Grafo

- Modelo común datos bajo nivel, en el aplicar mapeos.
- Tripletas. Elementos:
 - **Recurso**: algo identificable por una URI
 - **Propiedad**: relaciones binarias entre recursos
 - **Objeto**: URI o literal
- Conjunto tripletas: **Grafo**

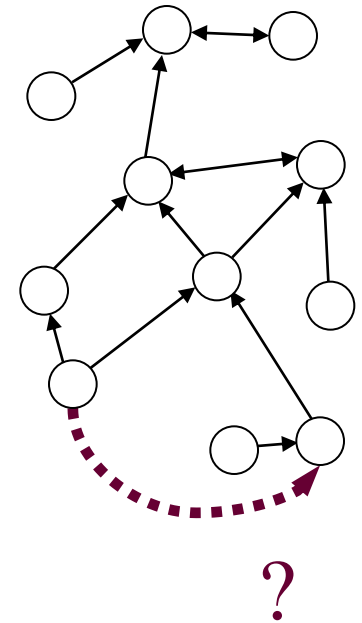


Principio: Modelo Grafo

- **Problema interoperabilidad vocabularios**
 - Definir un conjunto de “**etiquetas**”
 - Formalizar su **semántica** para facilitar interoperabilidad
 - Mecanismos para **mapear** vocabularios

Principio: Diseño Minimalista

- Estandarización mínima
- Proporcionar **marco global** con **reglas de mapeo** para interoperabilidad a nivel semántico
 - Subclase
 - Disjunto
 - Clase equivalente
 - Propiedad equivalente
 - Mismo individuo
 - ...



RDF

- Permite la descripción y el procesamiento de metadatos
 - No hace ninguna suposición sobre el dominio o campo de aplicación
 - No define ninguna semántica a priori
 - Tiene la capacidad de describir metadatos de cualquier dominio
- Sintaxis y estructura similar a la de los lenguajes orientados a objetos
 - Clases y subclases

RDF

- Les clases y subclases se disponen en una jerarquía.
 - Les subclases pueden heredar propiedades de las clases
 - Es posible la herencia múltiple, que permite
 - La mezcla de diferentes esquemas semánticos
 - Que los agentes enfrentados con una semántica desconocida la puedan trazar hasta encontrar elementos comunes a partir de los cuales deducir el resto
- Un conjunto de clases que definen un dominio o aplicación se llama *esquema*

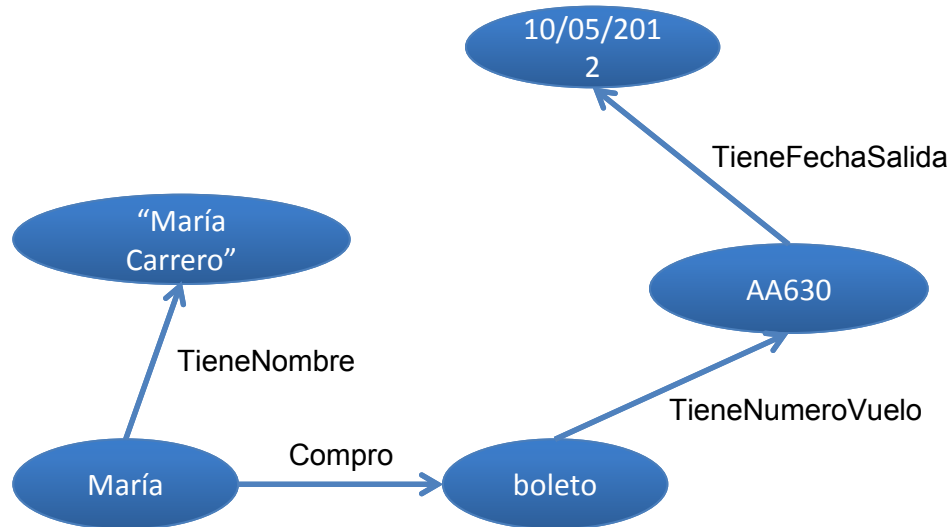
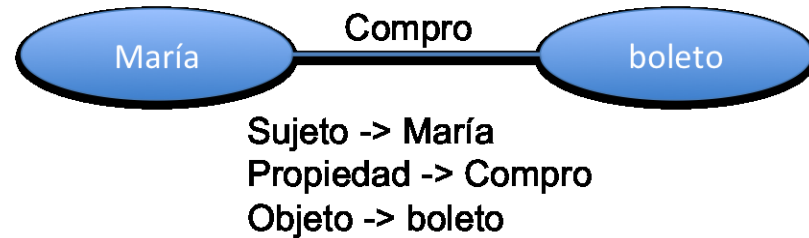
RDF: Características

- Basado en XML
- Es una Infraestructura de Descripción de recursos.
- RDF propone un modelo de datos coherente y lenguaje de esquema (RDFS) para definir vocabularios.
- Componentes de RDF:
 - Un modelo de datos formal.
 - Una sintaxis para intercambiar datos.
 - Un esquema.
 - Una sintaxis legible por máquina del esquema.
 - Protocolos de interrogación.

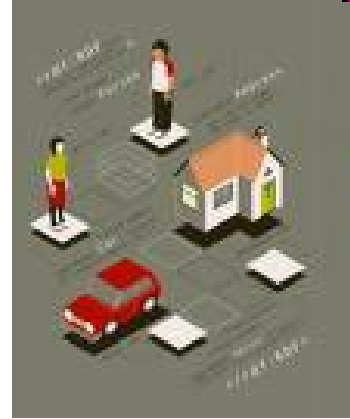
<http://www.w3.org/RDF>

RDF (Resource Description Framework)

- ✓ Sentencia en RDF es la tripleta
<sujeito, predicado, objeto>
 - ✓ <María, Compro, boleto>
- ✓ Se puede representar como un grafo



Componentes de RDF



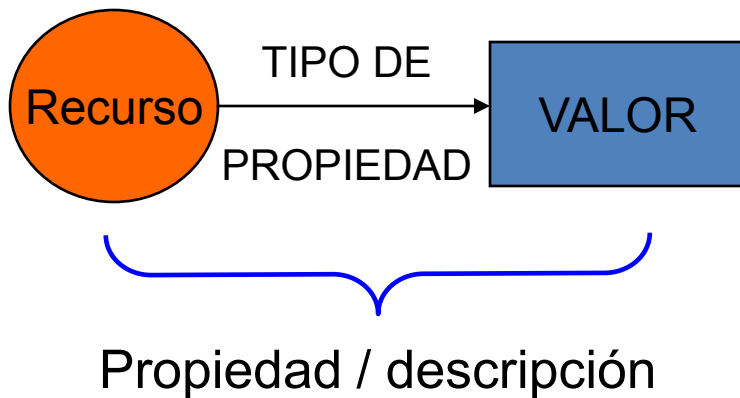
- **Modelo RDF:**
 - asocia tipos de propiedades (atributos) con sus valores.
- **Esquema RDF (RDFS)**
 - proporciona los mecanismos para crear vocabularios y relacionar propiedades y recursos.
- **Namespaces XML:**
 - identifica cada tipo de propiedad de forma unívoca.

RDF: Modelo

- Describe parejas de nombre y valores de una propiedad determinada
- Tres tipos de objetos
 - Recurso: cualquier objeto
 - se describe con un URI
 - Propiedad: característica, atributo, relación o aspecto que describe un recurso
 - Sentencia: el conjunto de
 - un recurso determinado
 - una propiedad con un nombre y un valor asociado al recurso

RDF: Modelo de datos

- El *modelo de datos* que propone RDF consiste en tres tipos de objetos:



Recursos

Tipos de propiedades

Descripciones

RDF: Modelo basado en declaraciones

- El modelo de datos de RDF está basado, en declaraciones, afirmaciones o aserciones (*statements*).
- Una afirmación / sentencia es un predicado de n componentes.
- Los valores pueden ser: *strings*, fechas y URIs.

S, P, O

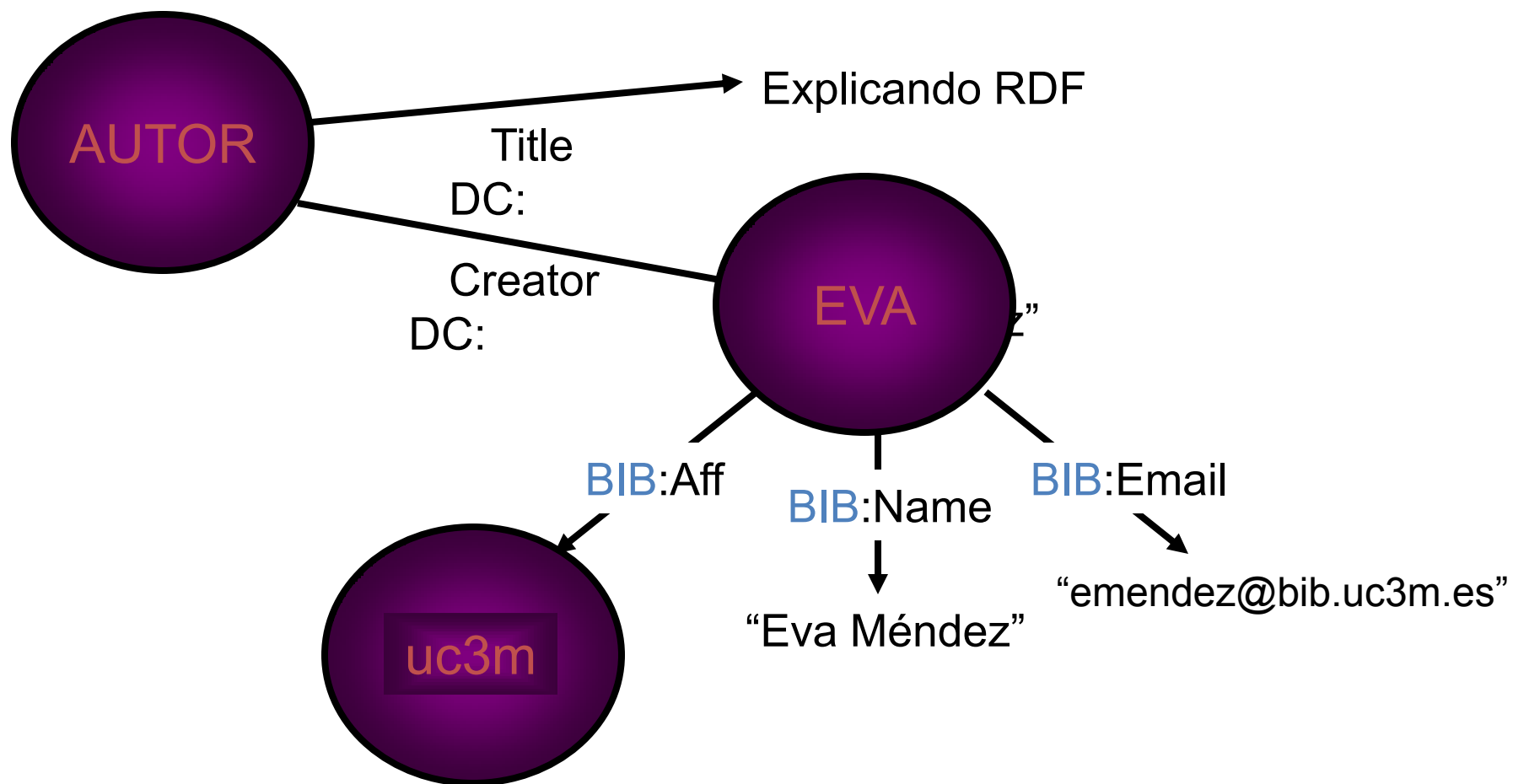
SUJETO (recurso)

PREDICADO (tipo de propiedad)

OBJETO (valor)

J. Aguilar

RDF: Representación gráfica



RDF: Recurso

- Recurso: cualquier objeto
 - Una Web entera
 - Una página web
 - Una casa.
- Sean de la naturaleza que sean
 - se describe con un URI
- URI (Universal Resource Identifier) abstracto
 - URL Localizador, una dirección concreta.
 - URN Nombre, permite cambiar la dirección y aún así encontrarla

RDF: Propiedad

- Propiedad: característica, atributo, relación o aspecto que describe un recurso
- Cada propiedad tiene
 - Significado
 - Define sus valores posibles
 - Define los tipos de recurso a los que es aplicable
 - Define la relación con otras propiedades

RDF: Sentencia

- Sentencia: el conjunto de
 - un recurso determinado
 - una propiedad con
 - un nombre
 - un valor asociado al recurso
- Cada elemento de la sentencia se llama
 - Sujeto: el recurso
 - Predicado: el nombre de la propiedad
 - Objeto: el valor de la propiedad

RDF Sintaxis básica

- El modelo define el marco abstracto
- Se requiere una sintaxis concreta
 - RDF usa XML como a lenguaje de base
 - También requiere la “XML namespace facility” para asociar cada propiedad con el esquema que la define
- La estructura de clases aparece en el esquema

RDF Sintaxis básica

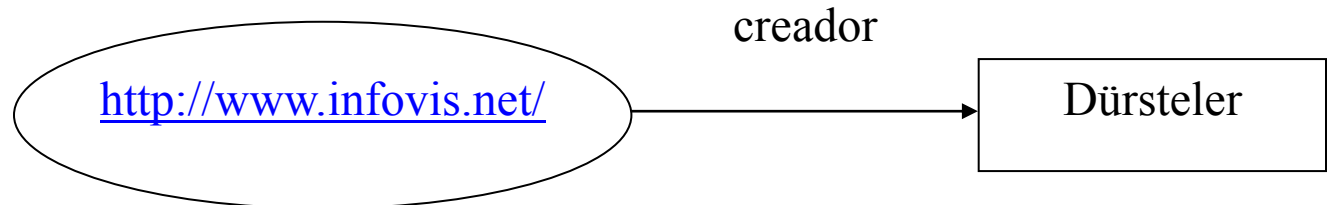
- Ejemplo

- Dürsteler es el creador de la web

<http://www.infovis.net>

- Sujeto: <http://www.infovis.net> (recurso)
 - Predicado: creador (propiedad, tiene un creador)
 - Objeto: Dürsteler (el valor de la propiedad)

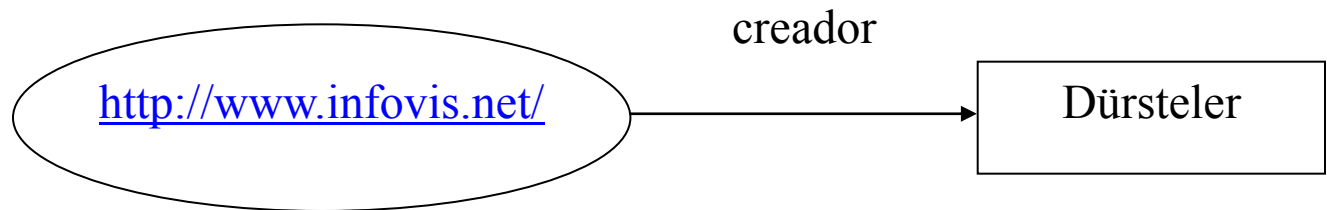
- En forma de grafo



RDF Sintaxis básica

- Ejemplo

- `<?xml version="1.0"?>`
- `<rdf:RDF`
- `xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
- `xmlns:s="http://description.org/schema/"`
- `<rdf:Description about="http://www.infovis.net">`
- `<s:Creador>Dursteler</s:Creador>`
- `</rdf:Description>`
- `</rdf:RDF>`



RDF Sintaxis básica serializada

- [1] RDF ::= ['<rdf:RDF>'] description* ['</rdf:RDF>']
- [2] description ::= '<rdf:Description' idAboutAttr? '>'
propertyElt*
- '</rdf:Description>'
- [3] idAboutAttr ::= idAttr | aboutAttr
- [4] aboutAttr ::= 'about="' URI-reference '"'
- [5] idAttr ::= 'ID="' IDsymbol '"'
- [6] propertyElt ::= '<' propName '>' value '</' propName '>'
| '<' propName resourceAttr '/>'
- [7] propName ::= QName
- [8] value ::= description | string
- [9] resourceAttr ::= 'resource="' URI-reference '"'
- [10] QName ::= [NSprefix ':'] name
- [11] URI-reference ::= string, interpreted para [URI]
- [12] IDsymbol ::= (any legal XML Name symbol)
- [13] name ::= (any legal XML name symbol)
- [14] NSprefix ::= (any legal XML namespace prefix)
- [15] string ::= (any XML texto, with "<", ">", and "&"
escaped)

RDF Sintaxis abreviada

- Ejemplo

- `<?xml version="1.0"?>`
- `<rdf:RDF`
- `xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
- `xmlns:s="http://description.org/schema/">`
- `<rdf:Description about="http://www.infovis.net">`
- `<s:Creador>Dursteler</s:Creador>`
- `</rdf:Description>`
- `</rdf:RDF>`

- Abreviadamente

- `<rdf:RDF>`
- `<rdf:Description about=http://www.infovis.net`
- `s:Creador="Dursteler" />`
- `</rdf:RDF>`

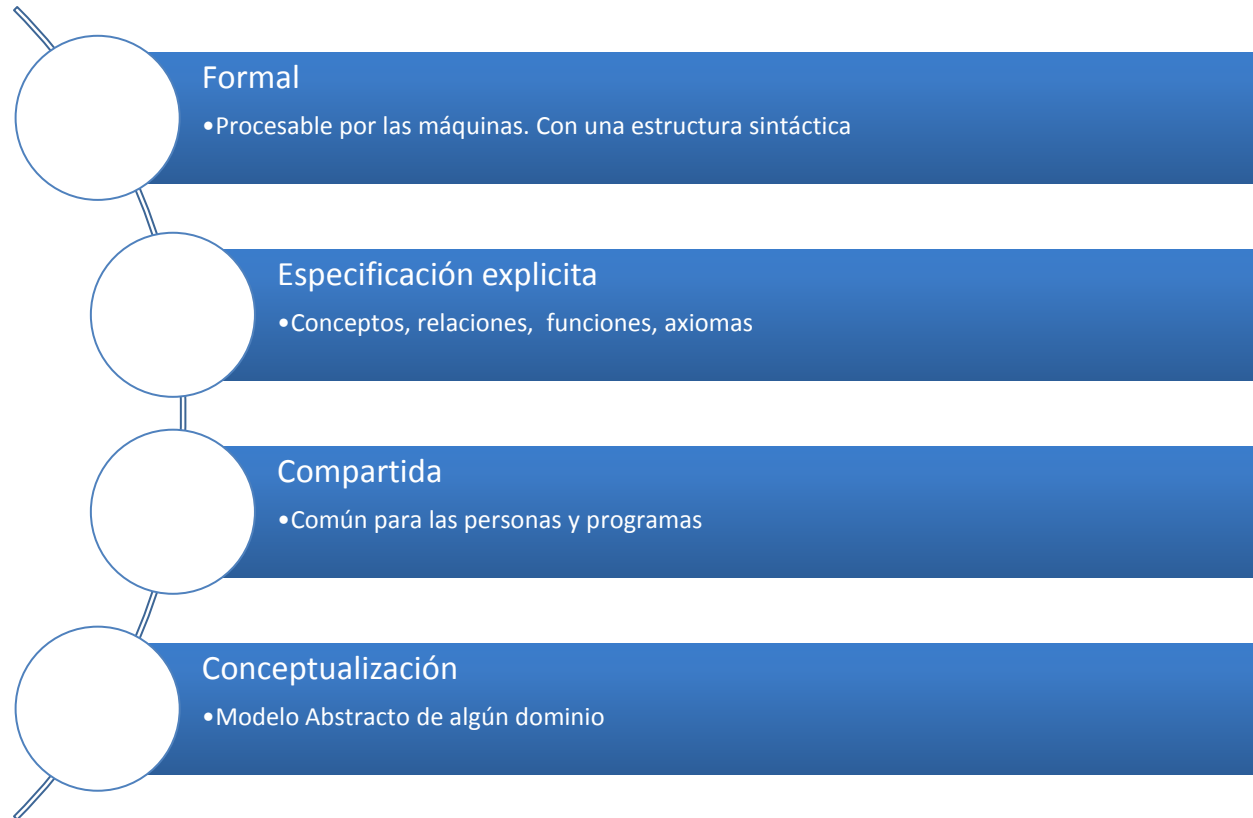
Esquemas

- Es una colección de recursos que se pueden utilizar para a
 - describir propiedades de otros recursos
 - que definen vocabularios específicos de una aplicación
- El esquema define una jerarquía de
 - clases,
 - propiedades
- con restricciones sobre las propiedades
 - como dominio, rango de aplicación, etc.

Esquemas

- El modelo proporciona una sintaxis que opera con unos objetos genéricos vacíos de semántica
- El modelo no proporciona ningún mecanismo para
 - declarar las propiedades
 - definir las relaciones entre propiedades y recursos
- El esquema es el que se encarga de ello
- El esquema RDF proporciona información sobre la interpretación de las sentencias de RDF

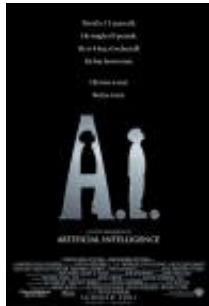
Ontologías



Ontología



Explicación sistemática de la Existencia



“Especificación explícita de una conceptualización” Gruber(1993)



“Base de datos que describe los conceptos de un dominio específico, sus propiedades y cómo estos conceptos se relacionan entre si” Weigand(1997)

Ontologías

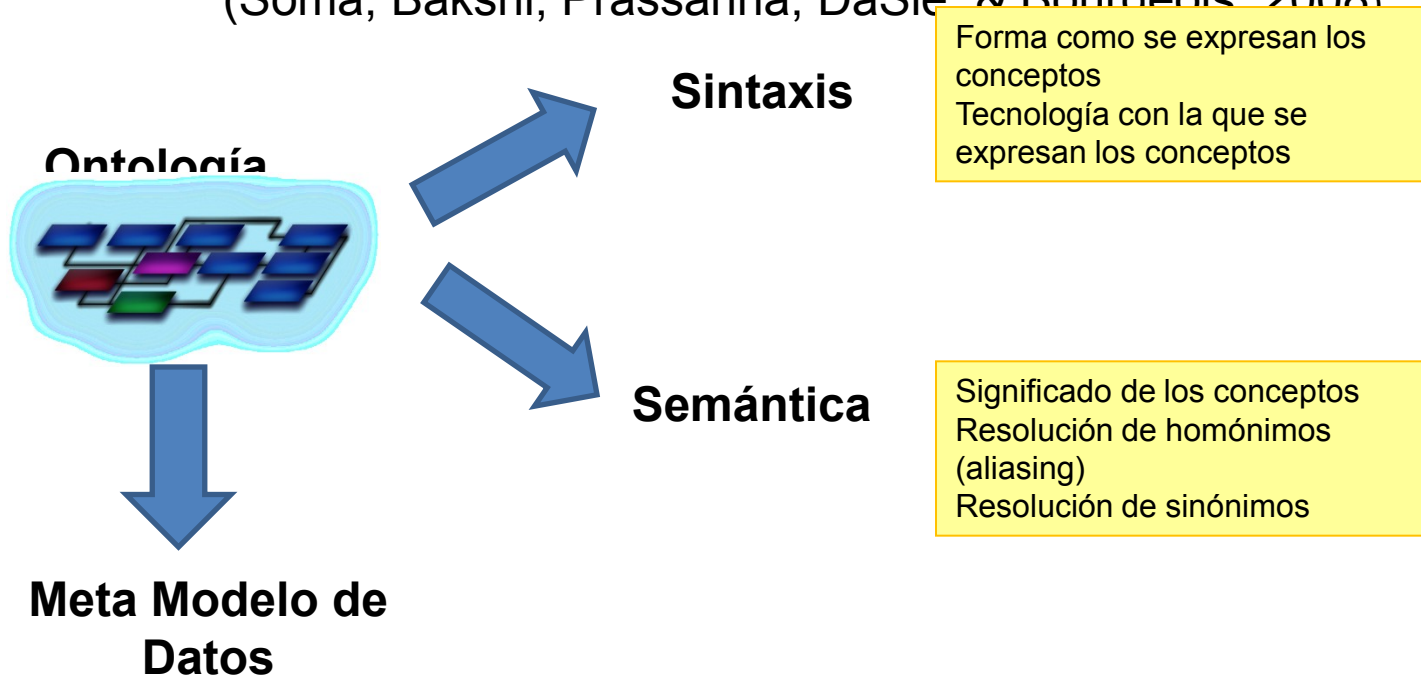
“An ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions to the vocabulary” [Neches 91]

Una *ontología* es un sistema de conceptos (o un vocabulario) usado como elemento básico (primitivo) para la construcción de sistemas basados en el conocimiento.

Ontologías

- Es una representación, a través de un modelo de datos, **de un conjunto de conceptos en un dominio y de las relaciones entre ellos.**
- Se utiliza comúnmente para resolver dos problemas fundamentales en grandes empresas: **integración de información y gestión del conocimiento**

(Soma, Bakshi, Prassanna, DaSieg & Bourgeois, 2008)



Ontologías

Compromisos Ontológicos: Acuerdo para usar un **vocabulario** que es consistente, aunque no completo

Ontologías

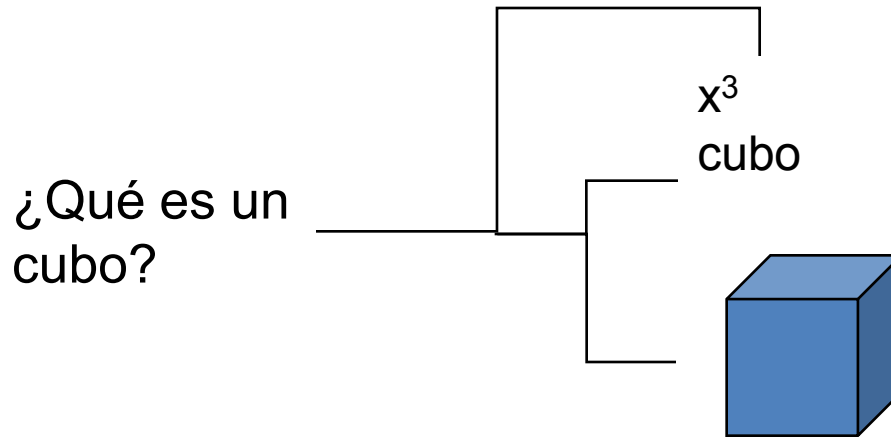
- ✓ Conceptos:
 - Ubicación (Ciudades, pueblos, etc.), alojamiento (hoteles, posadas, etc.), medios_de_transporte (aviones, trenes, carros, barcos, lanchas, motos)
- ✓ Relaciones:
 - Hotel_cuatro_estrellas es una subclase_de Hotel (**relación taxonómica entre conceptos**).
 - Nro_Vuelo es del tipo “Cadena de Caracteres” (**propiedad de atributos**),
 - Una ruta conecta dos diferentes ciudades (**relación no taxonómica entre conceptos**)
- ✓ Funciones:
 - Como pagar, la cual obtiene el precio de la habitación después de aplicar un descuento. Pagar(?habitacion, ?descuento) :-> ?Precio_final
- ✓ Instancias
 - AA7462, Hotel la Culata, Mérida.
- ✓ Axiomas
 - No es posible viajar desde Venezuela a USA por tren.

Componentes de una Ontología

El conocimiento se representa dentro de una ontología a través de:

- Conceptos
- Relaciones
- Funciones
- Instancias
- Axiomas

Ontologías



- **Conceptos.** Se emplean en un sentido amplio y pueden ser: tareas, funciones, acciones, estrategias, planes, etc.
- **Relaciones.** Representan un tipo de interacción entre conceptos del dominio.

Subclase-de: Concepto₁ x Concepto₂
Conectado con: Componente1 x Componente2

Ontologías

- **Funciones.** Son un tipo especial de relación.

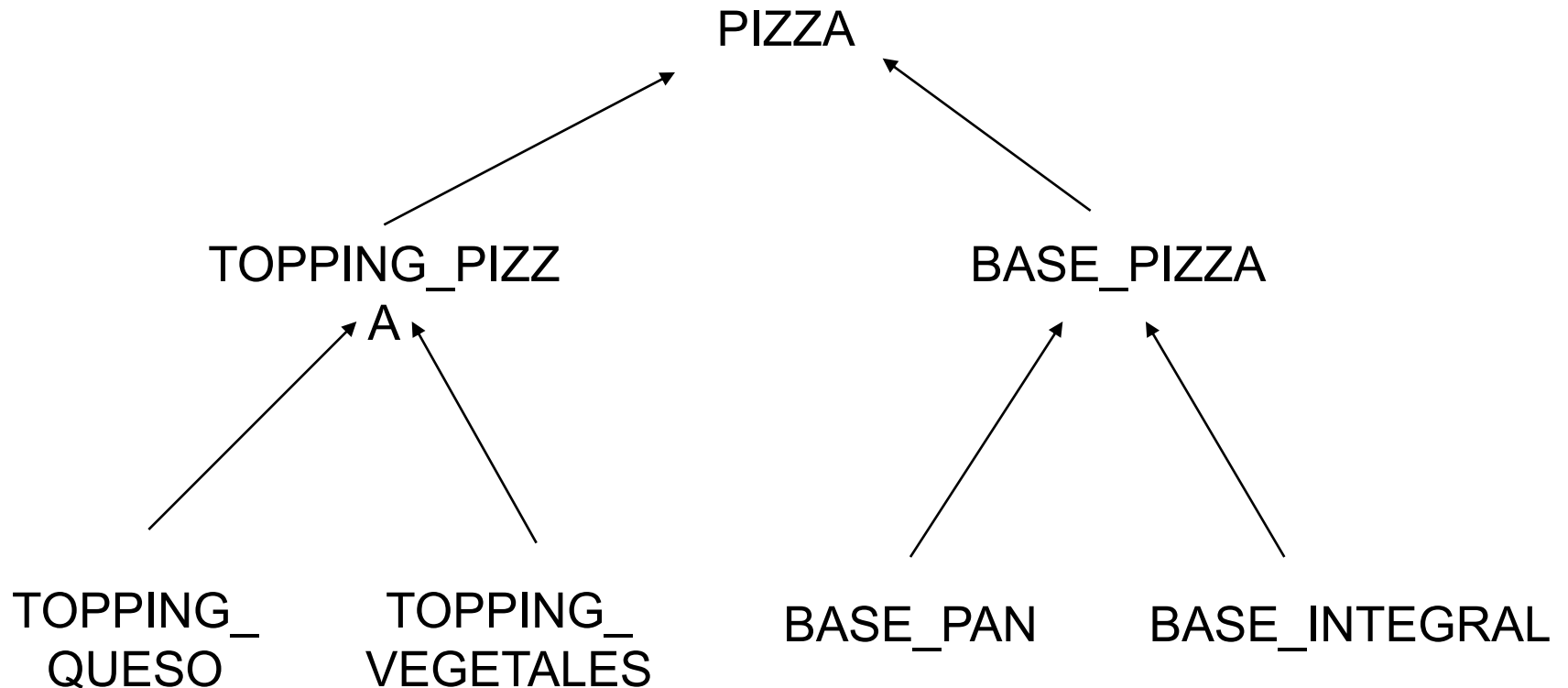
Madre-de--> Persona
Precio-objeto: Valor+Ganancia+IVA-->Precio

- **Instancias.** Concreciones en la ontología
- **Axiomas.** Propositiones que siempre son verdaderas

Conceptos

- Son las ideas básicas que se quieren formalizar para un determinado dominio de aplicación, y pueden estar organizadas en taxonomías.
- Pueden ser clases de objetos, métodos, planes, estrategias, procesos de razonamiento, etc.
- Por ejemplo:
 - Clase Pizza
 - Clase Topping:
 - Subclase ToppingQueso
 - Subclase ToppingCarne
 - Clase Base:
 - Subclase BaseIntegral
 - Subclase BaseNormal

Taxonomía de Conceptos



Relaciones

Representan las interacciones entre los conceptos del dominio.

- Forman la taxonomía del dominio
- Por ejemplo: Subclase_de, Parte_de
 - Base *es_parte_de* Pizza
 - Pizza Margarita *es_Subclase_de* Pizza

Propiedades de las Relaciones

describe a un individuo, pudiendo enlazarlo con otro

- Por ejemplo:
 - *tiene_ingrediente* tiene las subpropiedades:
 - *tiene_base*
 - *tiene_topping*
 - Pizza Margarita *tiene_ingrediente*
- Propiedad Inversa de *tiene_ingrediente*
 - *es_ingrediente_de*
 - *es_base_de*
 - *es_topping_de*

Propiedades de las Relaciones

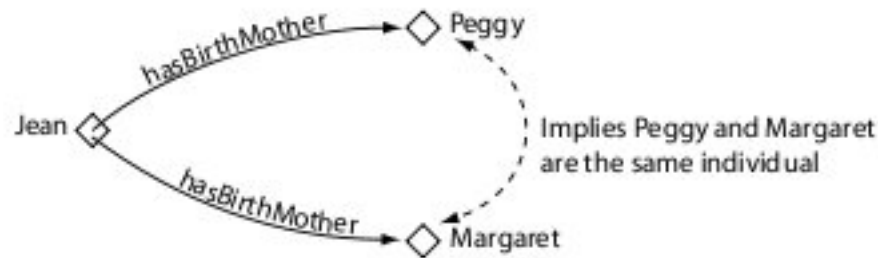
- **Propiedad Funcional**, cuando un individuo se relaciona con otro por medio de la propiedad.
 - tiene_base
- **Propiedad Inversa Funcional**, la relación inversa es funcional
 - es_base
- **Propiedad Transitiva**, la propiedad relaciona al individuo a con b y b con c , entonces se infiere que a se relaciona con c
 - tiene_ingrediente
- **Propiedad Simétrica**, cuando un individuo a se relaciona con b y b con a por medio de la misma propiedad

Propiedades funcionales

- Propiedades funcionales: Si una propiedad es funcional dado un objeto, puede estar a lo sumo un objeto relacionado con otro mediante esta propiedad.

Por ejemplo, si se tienen tres objetos que son A, B y C y se tiene una propiedad funcional tienePadre, entonces se podrían asociar los objetos A y B por medio de la propiedad y daría como resultado A tienePadre B. Igualmente se podrían asociar los objetos A y C por medio de la propiedad y daría como resultado A tienePadre C. Como tienePadre es propiedad funcional, se concluye que B y C son el mismo objeto.

Propiedad Funcional



Si una propiedad es funcional, para un individuo determinado, no puede haber más de una persona que se relaciona con el individuo a través de la propiedad.

En la figura se muestra un ejemplo de una propiedad funcional `hasBirthMother` - algo que sólo puede tener una madre biológica.

Si decimos que el individuo Jean `hasBirthMother` Peggy y también se dice que el individuo Jean `hasBirthMother` Margaret, entonces porque `hasBirthMother` es una propiedad funcional, se puede inferir que Peggy y Margaret debe ser el mismo individuo.

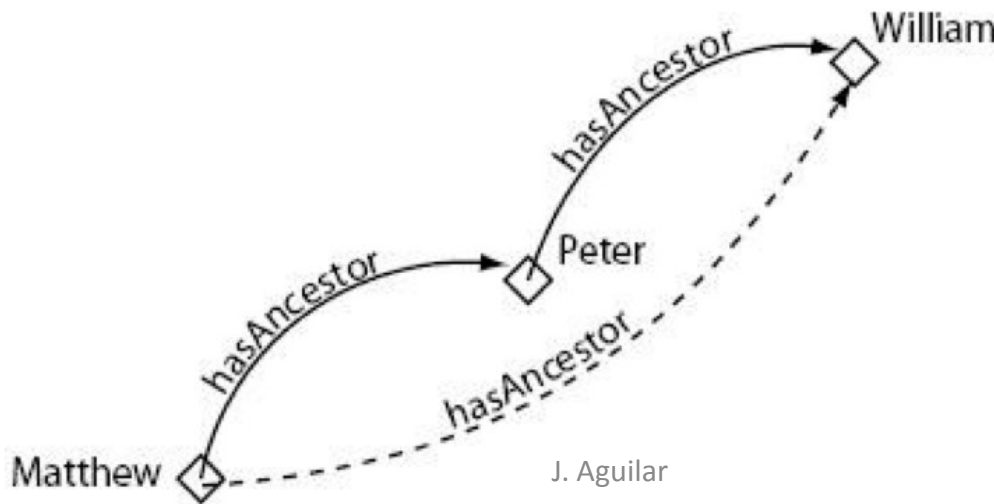
Propiedades funcionales inversa

- Propiedades funcionales inversas: Si una propiedad es funcional inversa, indica que puede estar a lo sumo un objeto relacionado con otro mediante esta propiedad de forma inversa a la propiedad funcional original.

Por ejemplo, si se tienen tres objetos que son A, B y C y se tiene una propiedad funcional esPadreDe, entonces se podría asociar el objeto B y A por medio de la propiedad y daría como resultado B esPadreDe A. Igualmente se podrían asociar los objetos C y A por medio de la propiedad y daría como resultado C esPadreDe A. Como esPadreDe es propiedad funcional inversa, se concluye que B y C son el mismo objeto.

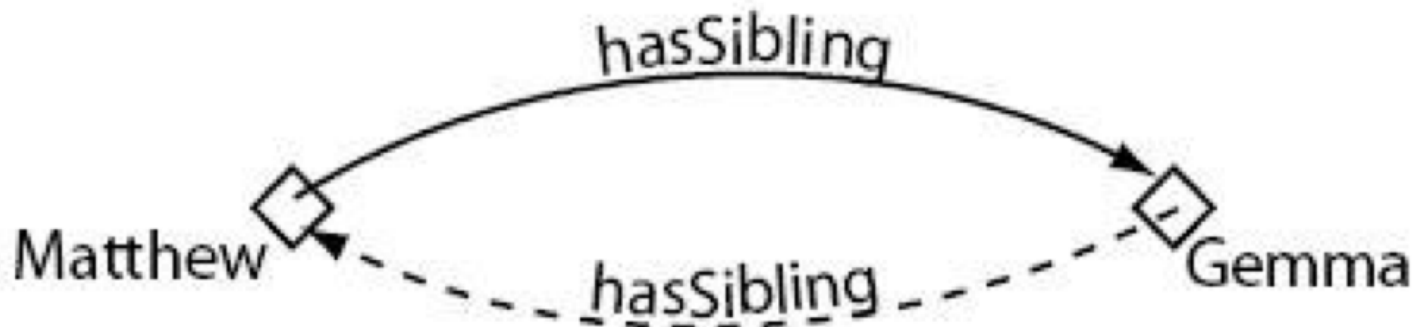
Propiedad Transitiva

- Propiedades transitivas: Si una propiedad es transitiva y relaciona dos objetos A y B, y además hay una propiedad que relaciona al objeto B con otro C, entonces se puede inferir que el objeto A está relacionado con el objeto C mediante la propiedad transitiva.

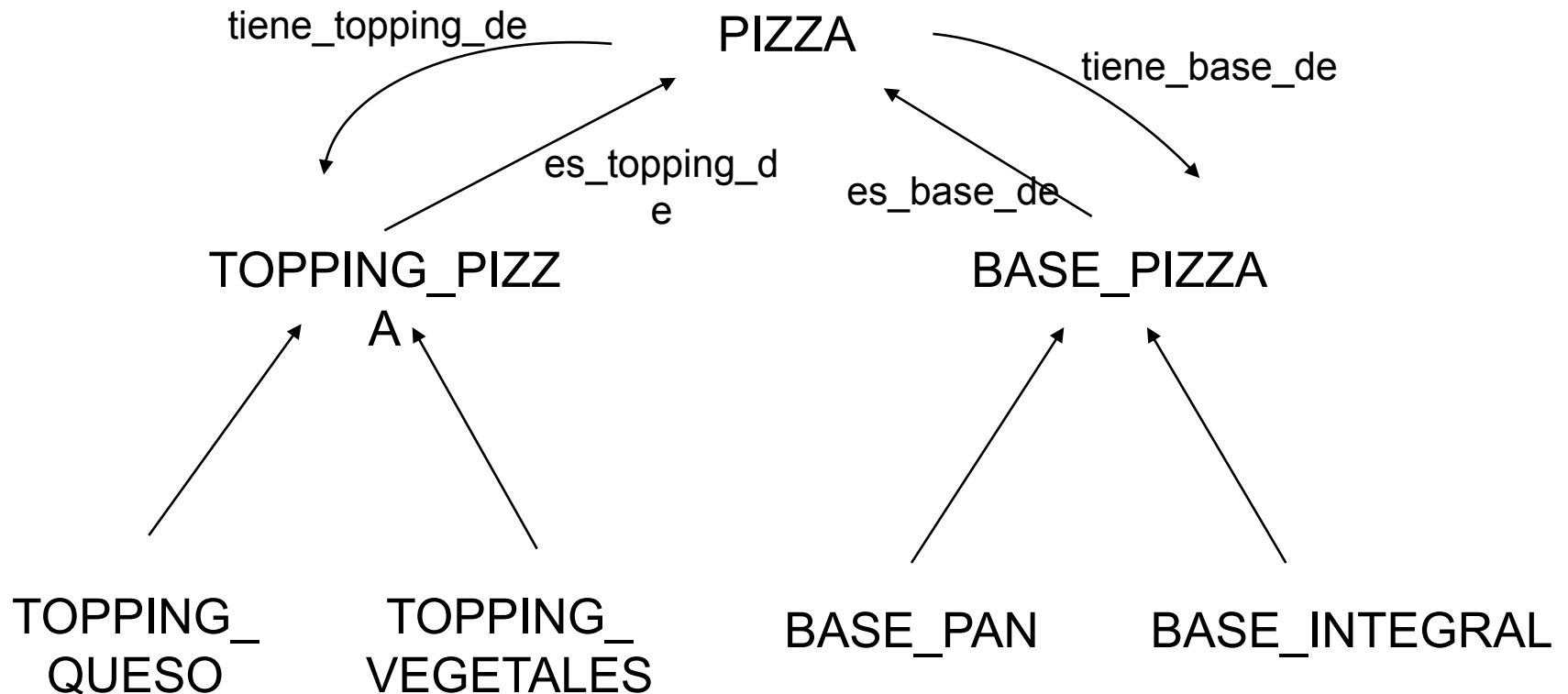


Propiedad simétrica

- Propiedades simétricas: Si una propiedad P es simétrica y la propiedad relaciona a los objetos A y B , entonces el objeto B es relacionado por medio de la propiedad P con el objeto A .



Taxonomía de Conceptos



Funciones

- Son un tipo concreto de relación donde se identifica un elemento mediante el cálculo de una función que considera varios elementos de la ontología.
- Por ejemplo, pueden aparecer funciones como: *asignar-fecha*

Axiomas

- Son teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología
- Por ejemplo:
 - Toda Pizza Margarita tiene Topping_Mozarella
 - Las Pizzas Vegetarianas no tienen Topping de Carne

Instancias

- Se usan para representar elementos o individuos en una ontología => hechos.
- Por ejemplo:
 - Pizza Vegetariana
 - Pizza Margarita

Características de la Ontología

- Categorías: taxonomía y herencia

$\text{tomate} \subset \text{Fruta}$

$\forall x x \in \text{tomate} \Rightarrow \text{Rojo}(x) \text{ y } \text{Redondo}(x)$

- Medidas

$\text{Precio}(\text{tomate}) = \$ (1.2)$

- Composición de Objetos

$\text{Partede}(\text{Caracas}, \text{Venezuela})$

Características de la Ontología

- Tiempo, Espacio y Evento

Subevento(BatallaCarabobo,IndependenciaVzla)

En(Caracas,Vzla)

T(En(jose, supermercado),ayer)

- Modelos Mentales

Conoceque(Agente,numerotelefono(Jose),cadena)

Cree(Luis,vuela(superman),mañana)

Cuestiones competenciales a las que debe de responder la ontología

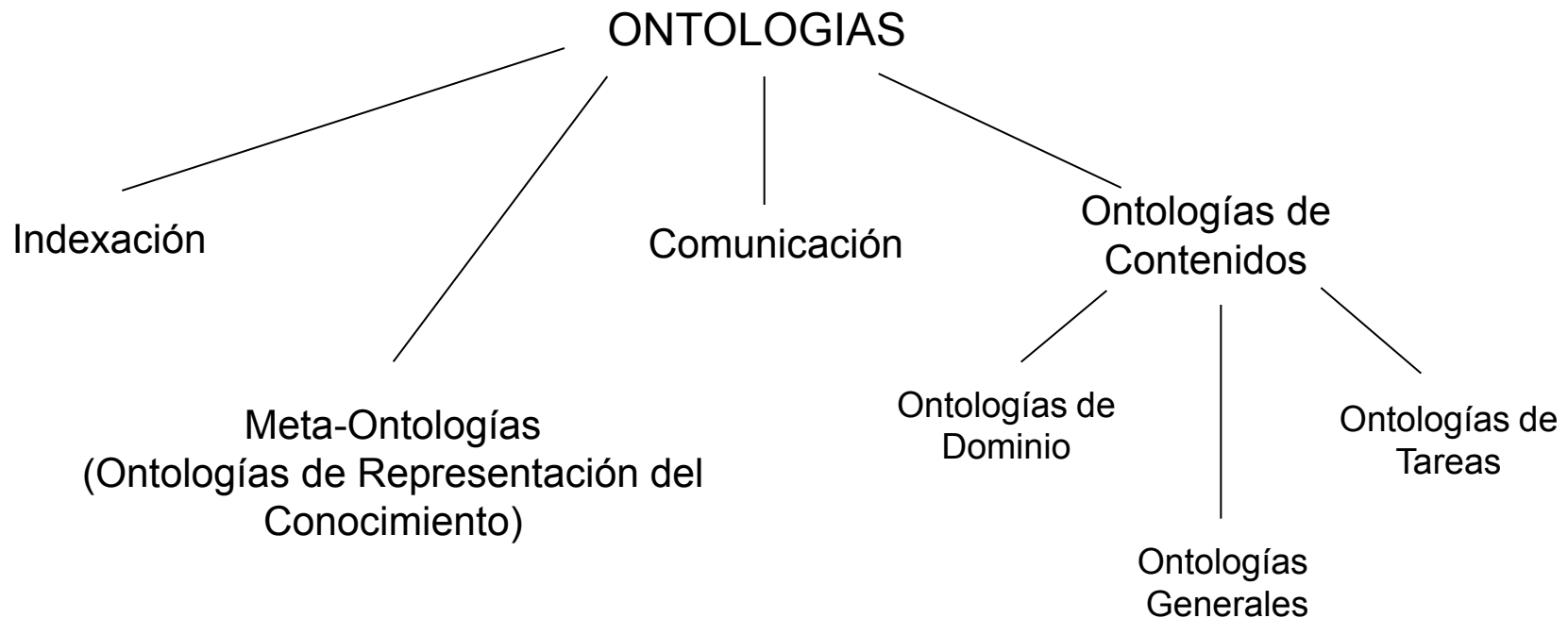
De acuerdo con Noy & Mc.Guinness

¿Para qué se va a utilizar?

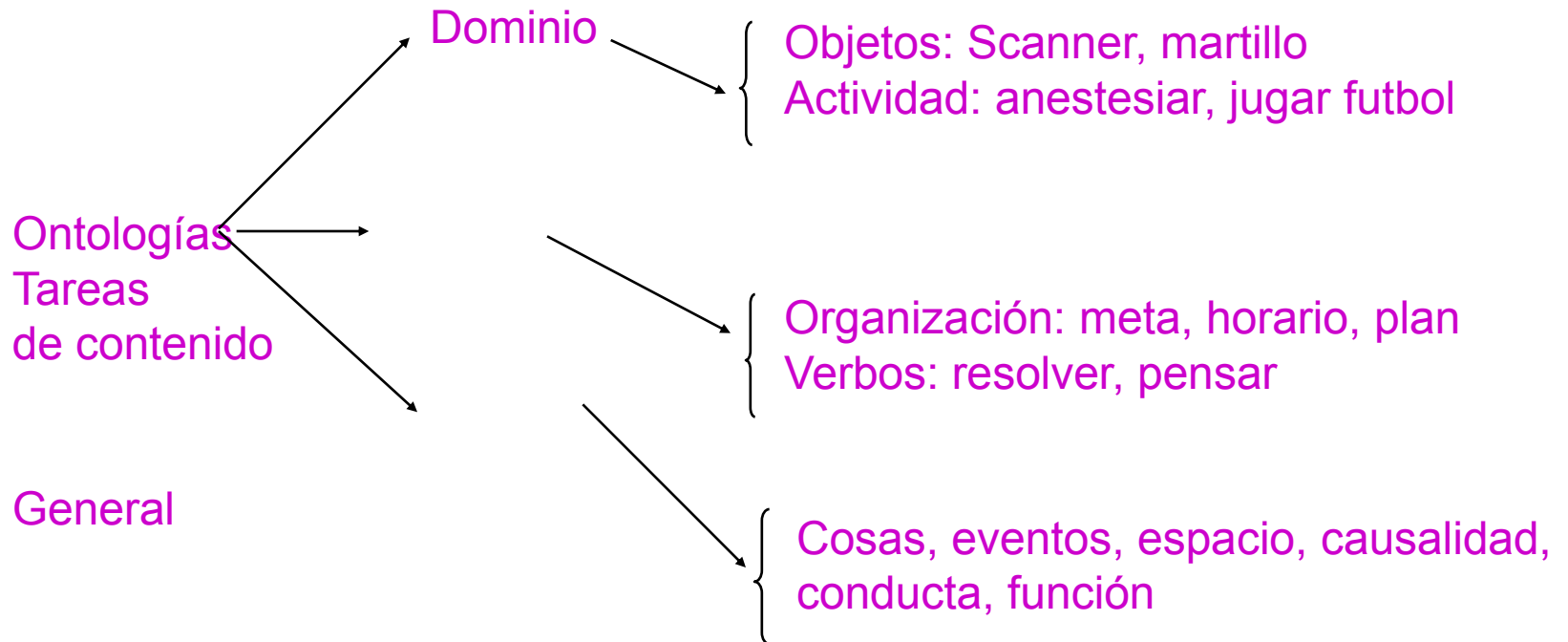
¿A qué tipo de preguntas va a dar respuesta ?

¿Quién la utilizará y mantendrá ?

Clasificación de las Ontologías



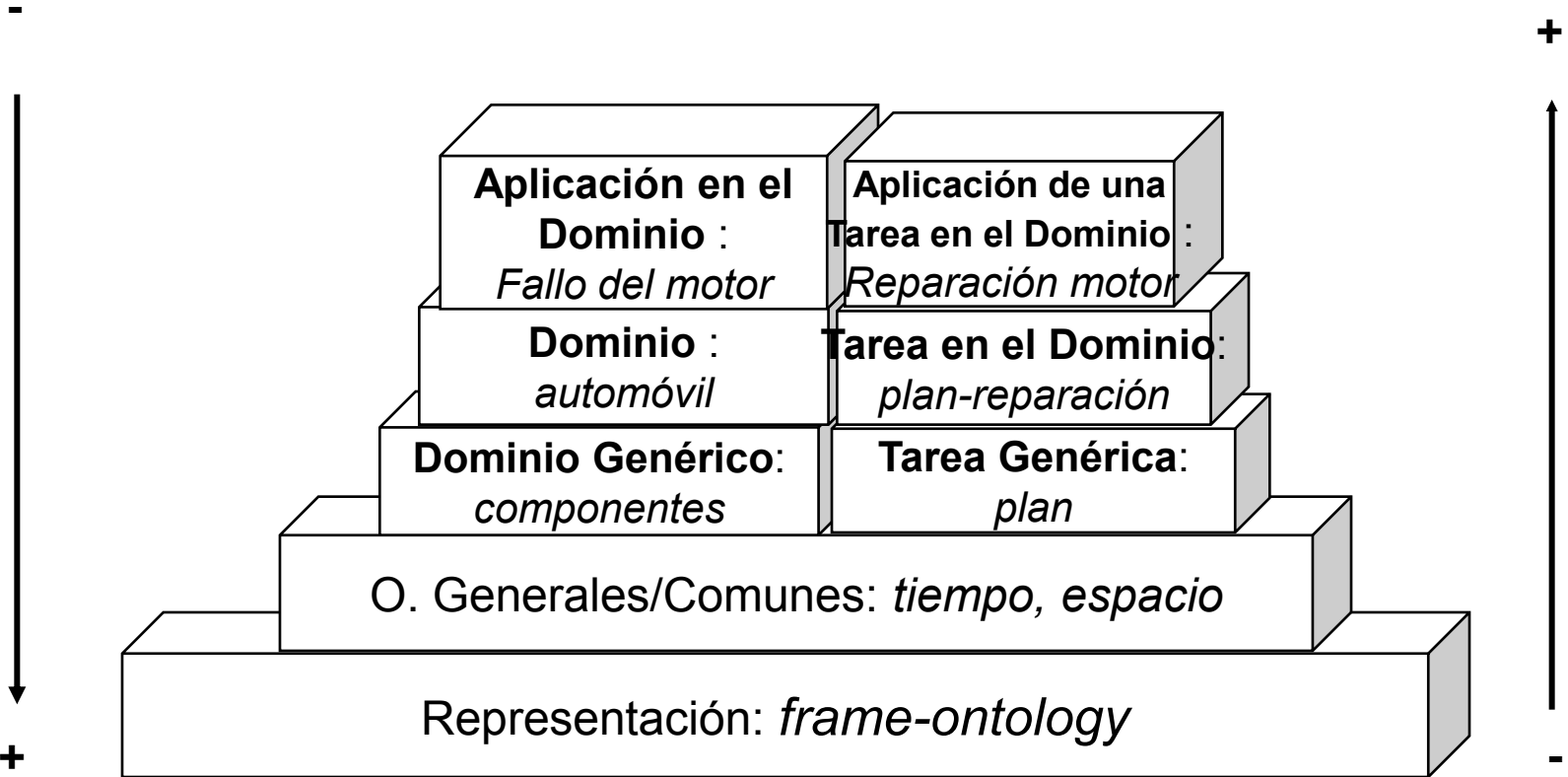
Ontologías de Contenido



Ontologías

Reusabilidad

Usabilidad



Tipos de Ontologías

- *Informales* si son expresadas en lenguaje natural
- *Semi-informales* si son expresadas de forma estructurada y restringida en lenguaje natural
- *Rigurosamente formales* si proporcionan términos definidos meticulosamente con semántica formal, teoremas y pruebas de propiedades tales como validez y completitud

Metodologías de Diseño de Ontologías

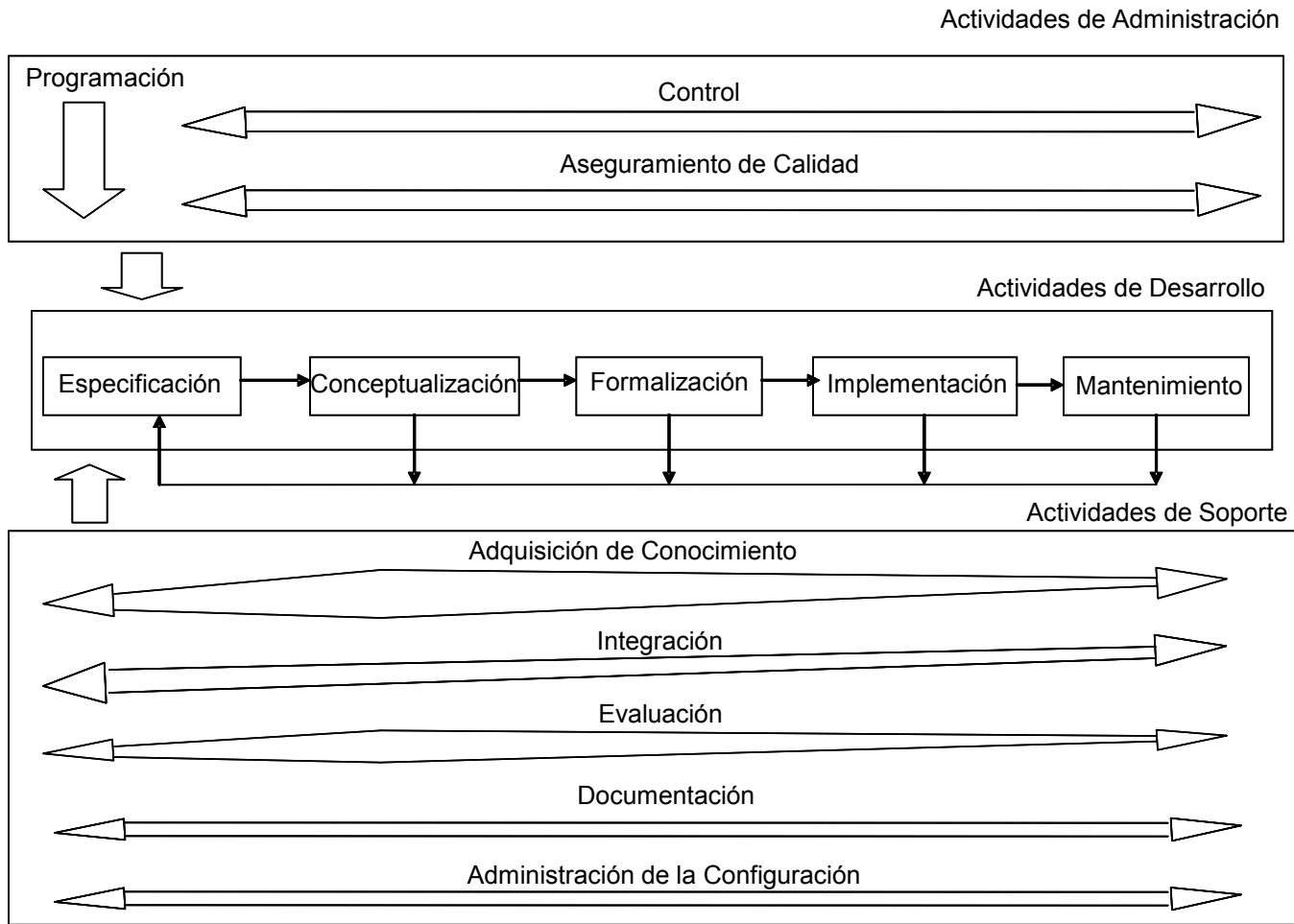
- Metodología de Uschold y King
- Metodología de Grüninger y Fox
- Metodología Kactus
- Metodología Methontology

Ontologías (metodología y fases)

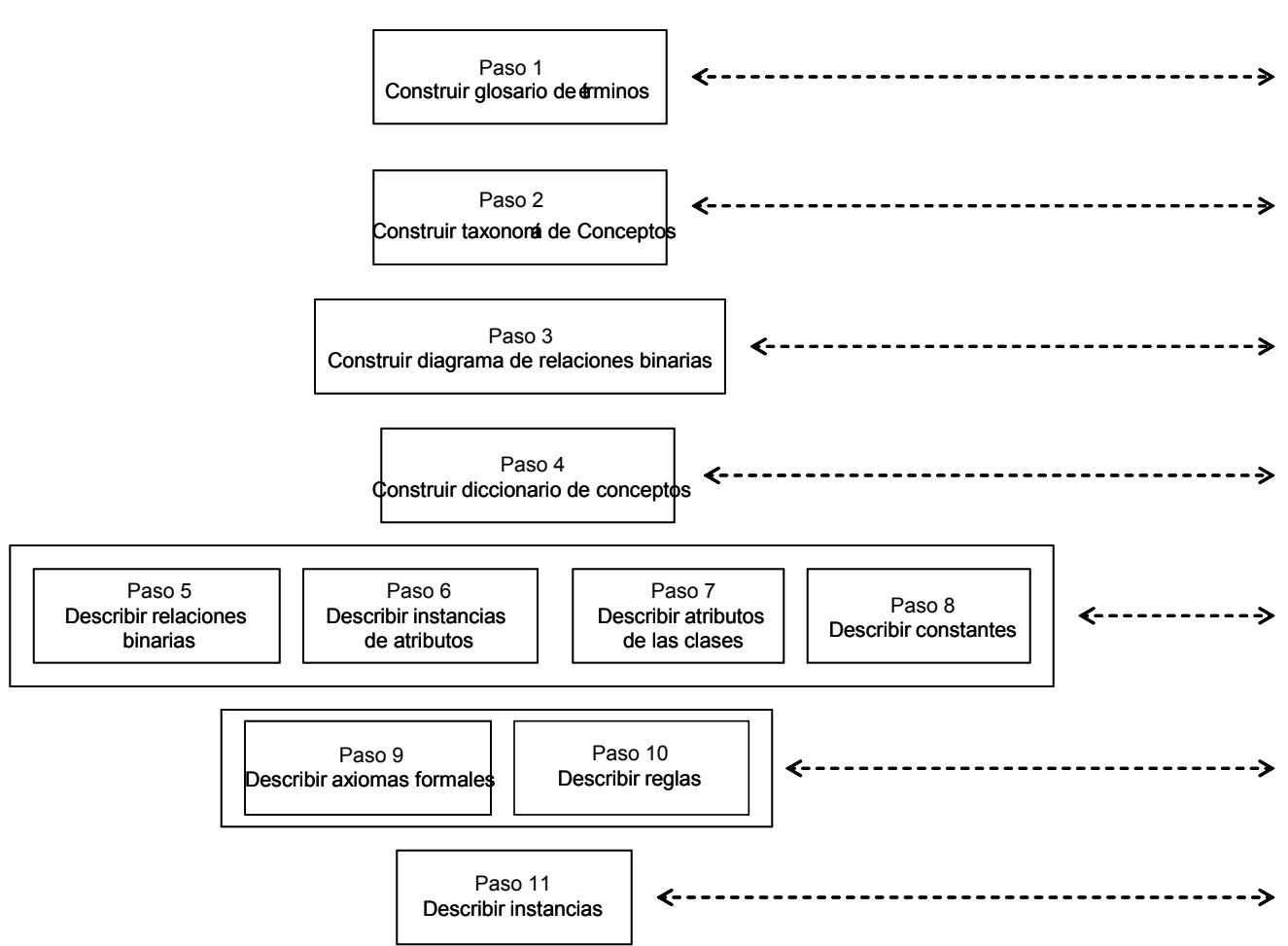
Methontology

1. Especificación
2. Conceptualización
3. Adquisición de conocimiento
4. Integración
5. Implantación
6. Evaluación
7. Documentación

Methontology



Pasos en Methontology



Representación de Ontologías

- Lenguajes basados en lógica de primer orden:
 - Ontolingua
 - DAML+ OIL (Darpa's Agent Markup Language + Ontologic Inference Language)
 - OWL (Web Ontology Language)
- Ambientes:
 - Protégé: <http://protege.stanford.edu/>
 - WebOde: <http://webode.dia.fi.upm.es/WebODEWeb/index.html>
 - OntoEdit: <http://ontoserver.aifb.uni-karlsruhe.de/ontoedit/>

Aplicaciones de una Ontología

- Como repositorios de conocimientos e información para la organización.
- Como herramienta para la adquisición de información, en situaciones en las que un equipo de trabajo la utiliza como soporte común para la organización del dominio.
- Para permitir la reutilización del conocimiento existente en nuevos sistemas.
- Como base para la construcción de lenguajes de representación del conocimiento, junto a la formalización del cálculo que tenga lugar entre los términos

Ventajas del uso de las ontologías

En tiempo de desarrollo...

**Consenso
entre
participantes** *

* **Comprensión
del dominio**

**Reutilización del
conocimiento** *

* **Validación**

En tiempo de ejecución...

Mejor *
**comunicación
entre agentes**

Integración *
fuentes de datos

* **Modelado contenido
semántico págs. web**

Interoperabilidad *

* **Aplicaciones
comercio electrónico**

* **Desambigüación
lenguaje natural**

Justificación de la ontología propuesta

Otras ventajas.....

1) Marco de referencia para la descripción formal de escenarios

=> garantizar el comportamiento consistente y no ambiguo

2) Definición del dominio de conocimiento

=> términos y conceptos relacionados con los escenarios educativos y su especificación

Ontologías: Estándares

- SHOE (*Simple HTML Ontology Extensions*)
<http://www.cs.umd.edu/projects/plus/SHOE/>
- OIL (*Ontology Inference Layer*)
- DAML (*DARPA Agent Markup Language*)
(<http://www.daml.org>), DAML+OIL
- RDF (<http://www.w3.org/RDF/>),
- OWL (*Web Ontology Language*)

Ontologías: Estándares

- OWL (*Web Ontology Language*):
<http://www.w3.org/2001/sw/WebOnt/>: 6 recomendaciones del W3C (10 febrero 2004)
 - [OWL Web Ontology Language Overview](#)
 - [OWL Web Ontology Language Guide](#)
 - [OWL Web Ontology Language Reference](#)
 - [OWL Web Ontology Language Semantics and Abstract Syntax](#)
 - [OWL Web Ontology Language Test Cases](#)
 - [OWL Web Ontology Language Use Cases and Requirements](#)
- + 2 Notas: Presentación de la sintaxis en XML y analizador de OWL en RDF/XML

Ontologías y Web

- En las web ontologías proporcionan entendimiento compartido de un dominio
- Es fundamental para hacer frente a las diferencias en la terminología
- Para comprender los datos en la web es fundamental que exista una ontología
- Para poder entender de forma automática los datos, y utilizar en un entorno distribuido, es crucial que la ontología sea:
 - explícitamente definido
 - Disponible en la web

Ontologías y Web

- Las ontologías son útiles para mejorar la precisión de las búsquedas en Internet
 - los motores de búsqueda pueden buscar páginas que hacen referencia a un concepto preciso en una ontología
 - Buen ejemplo: GoPubMed
- Búsquedas en la Web pueden explotar la información generalización / especialización
 - Si una consulta no puede encontrar un documento pertinente, el motor de búsqueda puede sugerir al usuario una consulta más general.
 - Si se recuperan demasiadas respuestas, el motor de búsqueda puede sugerir al usuario algunas especializaciones.

Web de información: Ontologías

- Entendimiento compartido de algún dominio de interés
- Vocabulario aceptado por una comunidad amplia
- Facilitan descripciones, búsqueda semántica y razonamiento

Lenguajes ontológicos: RDF

✓ **RDF permite**

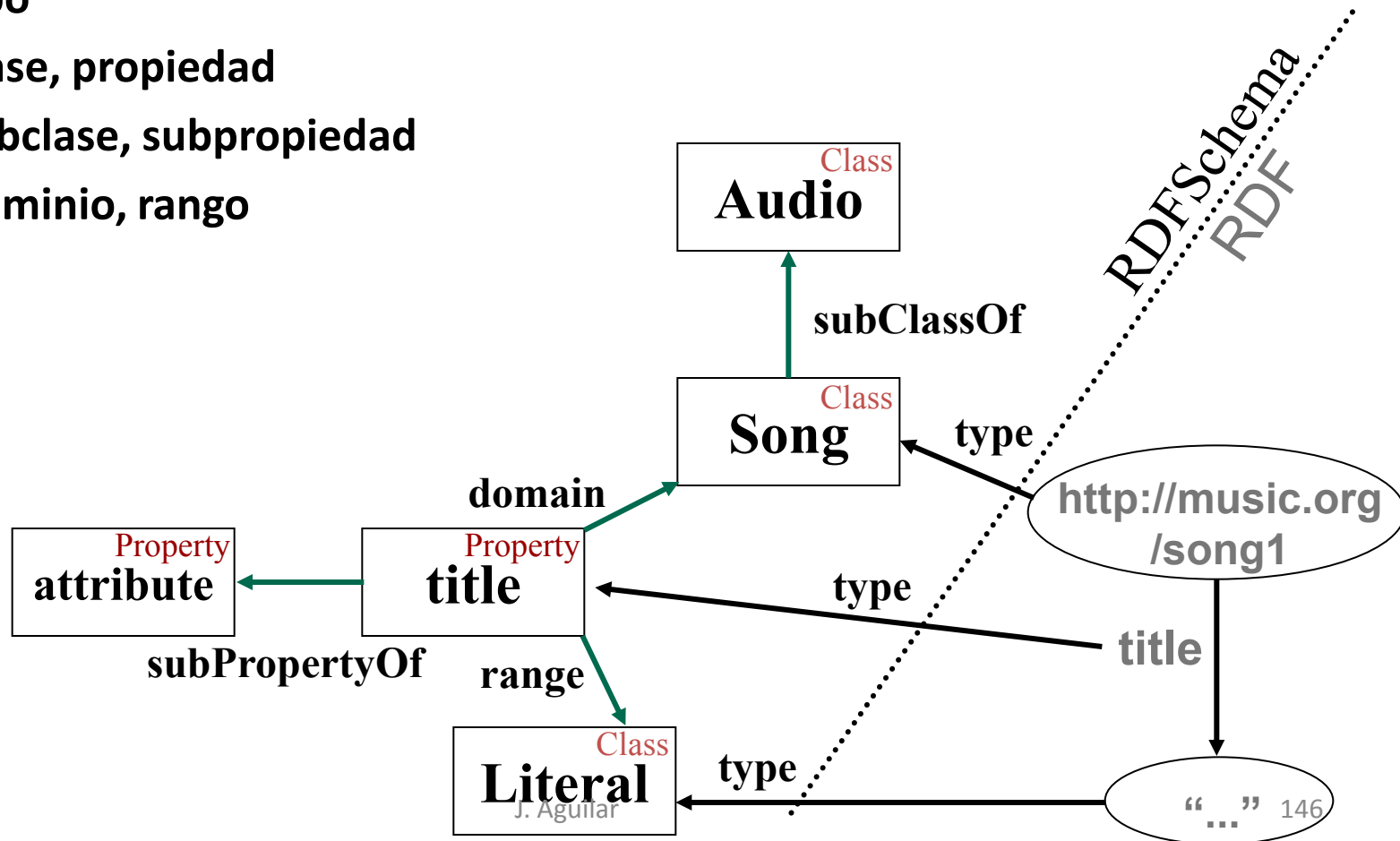
- ❖ Un mecanismo para anotar datos y recursos
- ❖ Modelo de datos simple
- ❖ Consistencia Sintáctica entre URIS

✓ **Limitantes RDF**

- ❖ No se puede representar la jerarquía de clases.
- ❖ No se pueden representar los dominios y rangos de los recursos, solo de las propiedades.
- ❖ No se pueden representar las jerarquías de propiedades.

Principio: Diseño Minimalista

- RDFSchema: primitivas tipo OO
 - tipo
 - clase, propiedad
 - subclase, subpropiedad
 - dominio, rango

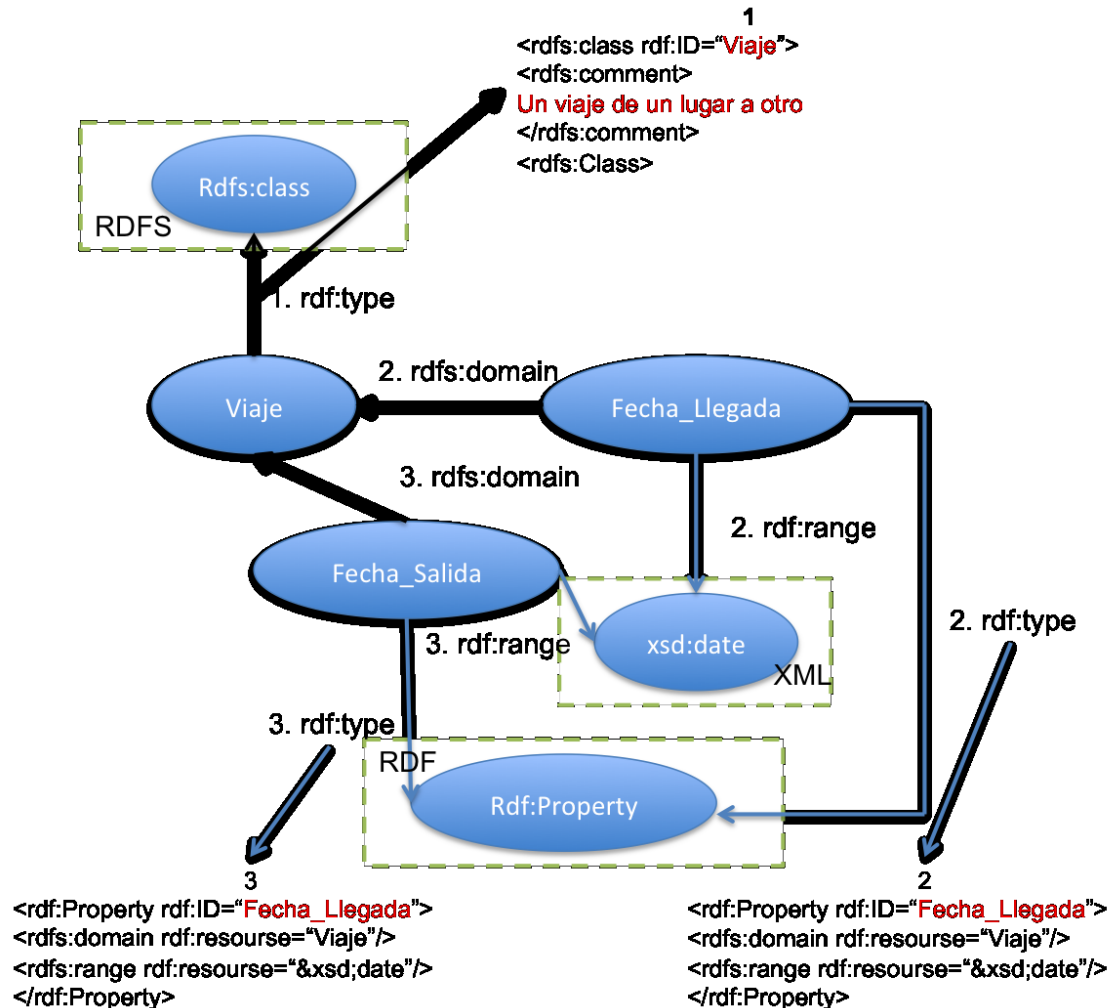


Lenguajes ontológicos: RDF Schema

- ✓ RDF Schema extiende RDF con un vocabulario que permite definir conceptos básicos y relación entre esos conceptos

- ❖ Class, type, subClassOf
- ❖ Property, subProperty, range y domain

- ✓ RDFS permite
 - ❖ Uso de un vocabulario simple para describir los recursos
 - ❖ Vocabulario consistente y compartido
 - ❖ Simple inferencia



Lenguajes ontológicos: RDF SCHema

- RDF (S) es muy débil para describir los recursos con detalles
 - Restricciones de rango y dominio
 - Restricciones de cardinalidad
 - Propiedades transitivas, simétrica o inversa
 - Dificultad de soporte de razonamiento

lo anterior es utopía si no resolvemos:

1. Escalabilidad
2. Falta de integridad referencial
3. Autoridad distribuída
4. Múltiples fuentes
5. Diversidad de contenidos
6. Uso impredecible de la información

...

...

y sobretodo, operar con grafos con *muchos* nodos

Se requiere

- Grafos que modelen procesos y datos reales (redes complejas)
- Bases de datos de grafos
- Interfaces para visualizar grafos
- Aplicaciones de marcado a gran escala

Contenido

Web de conocimiento / razonamiento
OWL

Web de Conocimiento

- Web Ontology Language (OWL)
 - Extiende RDFS para definir/relacionar ontologías
 - Posibilita un nivel avanzado de inferencia para la Web Semántica
 - Permite:
 - definir y restringir clases (conceptos).
 - definir y restringir propiedades.
 - definir relaciones entre clases.

Lenguajes ontológicos: OWL (Ontology Web Language)

OWL permite

- ✓ Definición de clases o conceptos mediante restricciones sobre propiedades, valores o cardinalidad.
-

- ✓ Definición de clases mediante operaciones booleanas sobre otras clases: intersección, unión y complemento.

- ✓ Relaciones entre clases (p.ej. Inclusión, disyunción, equivalencia).

3. owl:ObjectProperty

- ✓ Propiedades de las relaciones (p. ej. inversa, simétrica, transitiva).

```
<owl:ObjectProperty rdf:about="#Fecha_de_Salida">
  <rdfs:domain rdf:resource="#Vehiculos"/>
  <rdfs:range rdf:resource="#Fecha"/>
  <rdfs:isSubPropertyOf rdf:resource="#Fecha_de_Salida_unicamente_una"/>

```

```
<rdfs:range>  
<owl:Restriction>  
<owl:oneOfProperty rdfs:resource="http://www.w3.org/2002/07/owl#oneOfProperty"/>
```

```
<owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/
XMLSchema#nonNegativeInteger">1
</owl:qualifiedCardinality>
<owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#dateTime"/>
</owl:Restriction>
</rdfs:range>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#Fec"
<rdfs:domain rdf:resource="#viajes"/>
```

```
<owl:ObjectProperty rdf:about="#Fecha_de_Llegada">
<rdfs:domain rdf:resource="#viajes"/>
<rdfs:range>
Definición de
<owl:Restriction>
<owl:onProperty rdf:resource="#Fecha"/>
<owl:qualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#integer" value="1"/>
</owl:qualifiedCardinality><owl:onDataRange rdf:resource="#xsd:date"/>
>
</owl:Restriction>
```

Definición de la clase Viaje en owl. [Rodriguez/Aguilar]

```
<!--@type="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1  
onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#dateTime"/
```

Principio: Diseño Minimalista

- **Lenguaje Ontología Web (OWL)**
- **Aumentan el detalle del modelo conceptual**
- **Primitivas OWL:**
 - **clases y propiedades equivalentes**
equivalentClass, equivalentProperty
 - **clases disjuntas**
disjointWith
 - **propiedades opuestas, transitivas, simétricas**
inverseOf, TransitiveProperty, SymmetricProperty
 - **Cardinalidades**
maxCardinality, minCardinality
 - **Propiedad valor único**
FunctionalProperty
 - **Propiedad valor identificador**
InverseFunctionalProperty

OWL

- W3C's Grupo de Trabajo Ontología define 3 sublenguajes OWL diferentes:
 - OWL Full
 - Lenguaje más expresivo
 - OWL DL
 - Correspondiente a DL SHOIN
 - OWL Lite
 - Correspondiente a DL CAPS

Construcción de ontología owl

- Las ontologías son usadas para capturar el conocimiento sobre algún dominio de interés.
- Una ontología describe los conceptos dentro del dominio y la relación que tiene entre esos conceptos.
- Lenguaje estándar es OWL desarrollado por W3C.
- Igual que Protégé, OWL permite describir conceptos y además nuevas facilidades. Cuenta con un conjunto de operadores (intercesión, unión, y negación).
- OWL esta basado en lógica descriptiva que permite el uso del razonador.

Componentes de una ontología owl

Ontologías	OWL	PROTÉGÉ
Individuos	Individuos	Casos (instance)
Relaciones	Propiedades	Slots
Conceptos	Clases	Clases

Individuos en owl

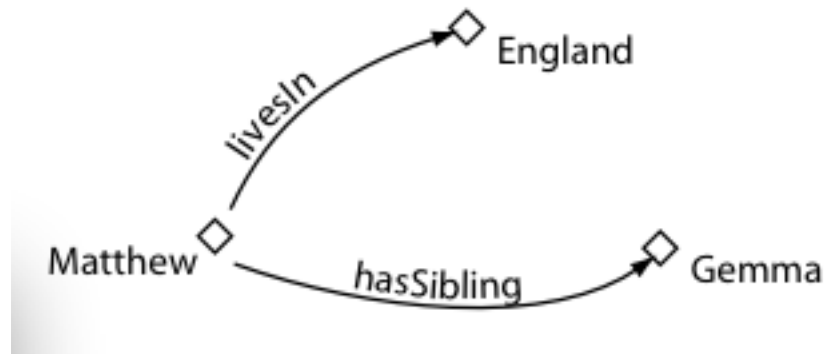
- Los individuos representan objetos del dominio de interés y son también conocidos como instancias.



Representación de Individuos

PROPIEDADES EN owl

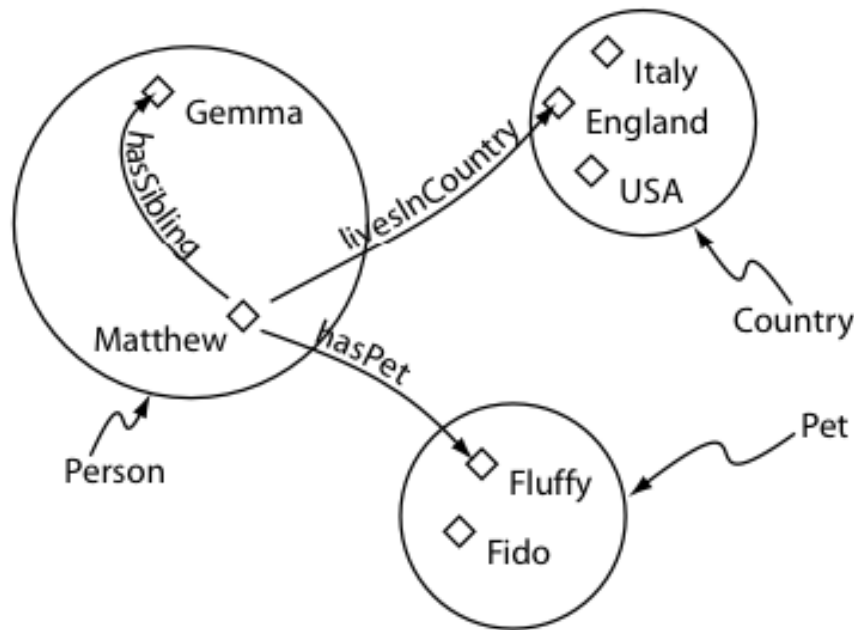
- Las propiedades son relaciones binarias sobre los individuos y pueden ser inversas, transitivas o simétricas.



Representación de Propiedades

CLASES EN owl

- Las clases OWL se entienden como conjuntos que contienen individuos y pueden ser organizadas dentro de una jerarquía de clases y subclases conocida como taxonomía. Las clases también son conocidas como conceptos, pues son una representación concreta de éstos.



Representación de clases

Lógica Descriptiva y owl

Lógica de primer orden

- El fundamento que garantiza la pureza lógica de la ontologías es la lógica de primer orden. Sobre ella se asienta las lógicas descriptivas (DL), así como el OWL.
- Porque usamos lógicas descriptivas:
 - Lógica de primer orden es indecidible (es fácil afirmar cosas de objetos, pero computacionalmente complejo)
 - Se requiere de un lenguaje formal para construir y combinar definiciones de categorías (p.ej. Relaciones de subconjunto y superconjunto)
 - Razonadores semánticos se basan en ella: FaCT++, Rancer, Pellet, ...

Lógica descriptiva

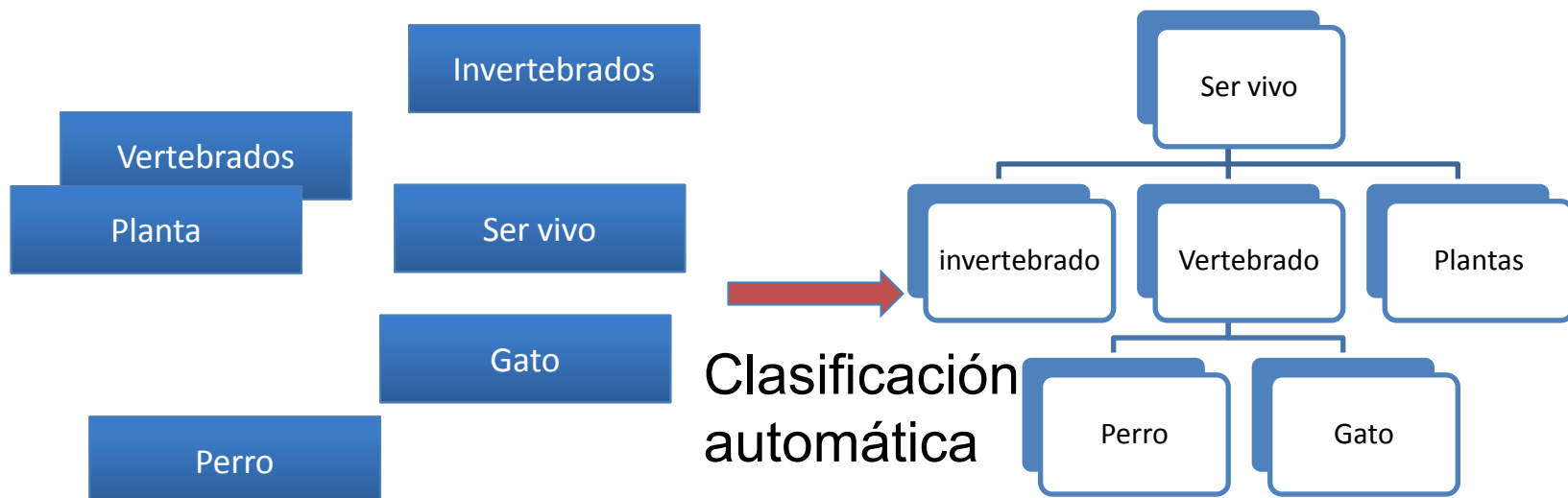
- Lenguajes de representación del conocimiento
- DL se diseñó como una extensión de marcos y redes semánticas, equipada con semántica basada en la lógica.
- Características :
 - Un formalismo descriptivo: conceptos, roles (relaciones), individuos.
 - Un formalismo terminológico: axiomas que describen propiedades genéricas.
 - Un formalismo asertivo: introduce propiedades de individuos.

Lógica descriptiva

- Principales tareas de inferencia con lógica descriptiva:
 - Subsunción (comprobar si una categoría es subconjunto de otra)
 - Clasificación (comprobar si un objeto pertenece a una categoría)
 - Ejemplo:
 - Soltero = $Y(\text{NoCasado}, \text{Adulto}, \text{Masculino})$
 - $\text{Soltero}(x) \Rightarrow \text{NoCasado}(x) \wedge \text{adulto}(x) \wedge \text{masculino}(x)$
- (lógica de primer orden)

Lógicas descriptivas

- Ejemplo: realiza clasificación automática (realizada por el motor de inferencias del lenguaje-razonador) en tiempo de ejecución



Lenguajes ontológicos: Owl dl

	Lógica Descriptiva	OWL	Ejemplo
Intercepción	$C_1 \cap \dots \cap C_2$	<u>intersectionOf</u>	<i>Persona</i> $\cap$ <i>Masculino</i>
Unión	$C_1 \cup \dots \cup C_2$	<u>unionOf</u>	<i>Doctor</i> $\cup$ <i>Abogado</i>
Negación	$\neg C$	<u>complementOf</u>	$\neg$ <i>Masculino</i>
Nominales	$\{x_1\} \cup \dots \cup \{x_2\}$	<u>oneOf</u>	$\{María\} \cup \{Juan\}$
Restricción universal	$\forall P. C$	<u>allValuesFrom</u>	$\forall$ <i>tieneHijos</i> . <i>Doctor</i>
Restricción existencial	$\exists P. C$	<u>someValuesFrom</u>	$\exists$ <i>tieneHijos</i> . <i>Abogado</i>
<u>Cardinalidad máxima</u>	$\leq nP$	<u>maxCardinality</u>	≤ 3 <i>tieneHijos</i>
<u>Cardinalidad mínima</u>	$\geq nP$	<u>minCardinality</u>	≥ 1 <i>tieneHijos</i>
Valores específicos	$\exists P. \{x_1\}$	<u>hasValue</u>	$\exists$ <i>tieneColea</i> . $\{Jose\}$
Sub clases	$C_1 \subseteq C_2$	<u>subClassOf</u>	<i>Profesor</i> $\sqsubseteq$ <i>Persona</i>
Equivalencia	$C_1 \equiv C_2$	<u>equivalentClass</u>	<i>Hombre</i> $\equiv$ <i>Persona</i> $\cap$ <i>Masculino</i>
Disjuntos	$C_1 \cap C_2 \subseteq \perp$	<u>disjointWith</u>	<i>Masculino</i> $\cap$ <i>Femenino</i> $\subseteq \perp$
<u>Subpropiedad</u>	$P_1 \subseteq P_2$	<u>subPropertyOf</u>	<i>tieneHija</i> $\subseteq$ <i>tieneHijos</i>
Equivalente	$P_1 \equiv P_2$	<u>equivalentProperty</u>	<i>Costo</i> $\equiv$ <i>Precio</i>
Inversa	$P_1 \equiv P_2 -$	<u>inverseOf</u>	<i>TieneHijo</i> $\equiv$ <i>TienePadres</i> $-$

Lógica Descriptiva y OWL. [Fuente: Oscar Corcho García. Representación de Conocimiento: Lógica Descriptiva]

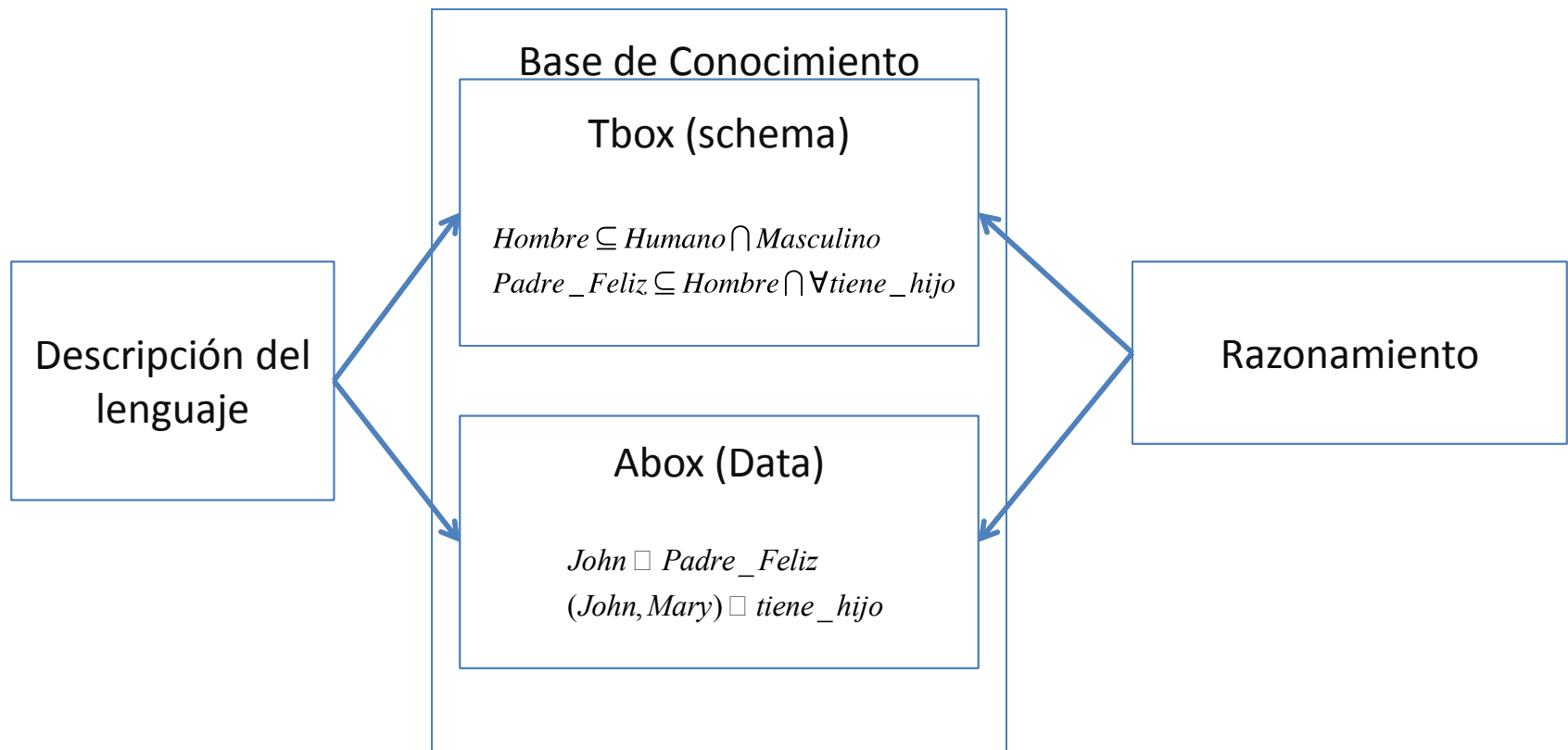
Correspondencia entre OWL y DL

Constructor OWL	Representación DL	Ejemplo
owl:equivalentTo (C,D)	$C \equiv D (C \sqsubseteq D \text{ y } D \sqsubseteq C)$	<i>Persona</i> $\equiv$ <i>Humano</i>
rdfs:subClassOf (C,D)	$C \sqsubseteq D$	<i>Padres</i> $\sqsubseteq$ <i>Persona</i>
owl:complementOf (C,D)	$C \equiv \neg D (\text{negacion})$	<i>Varon</i> $\equiv \neg$ <i>Mujer</i>
owl:disjointWith (C,D)	$C \sqsubseteq \neg D$	<i>Padre</i> $\sqsubseteq \neg$ <i>Madre</i>
owl:intersectionOf (C,D)	$C \sqcap D (\text{conjuncion})$	<i>Padres</i> $\sqcap$ <i>Varon</i>
owl:unionOf (C,D)	$C \sqcup D (\text{disjuncion})$	<i>Padre</i> $\sqcup$ <i>Madre</i>
owl:oneOf (I1, I2)	$\{I_1\} \sqcup \{I_2\}$	$\{\text{Juan}\} \sqcup \{\text{Maria}\}$
owl:someValuesFrom(P,C)	$\exists P.C (\text{existencial})$	$\exists \text{tieneHijo.Hija}$
owl:allValuesFrom(P,C)	$\forall P.C (\text{universal})$	$\forall \text{tieneHijo.Hijo}$
owl:hasValue (P,I1)	$\exists P.\{I_1\}$	$\exists \text{tieneHijo.}\{\text{Juan}\}$
owl:cardinality(P,n)	$= n.P$	$= 2.\text{tienePadres}$
owl:minCardinality(P,n)	$\geq n.P$	$\geq 1.\text{tieneHija}$
owl:maxCardinality(P,n)	$\leq n.P$	$\leq 2.\text{tieneHijos}$

Un *concepto* en DL se refiere a una *clase* en OWL.

Un *rol* en DL es una *propiedad* en OWL.

Arquitectura DL



Lógicas descriptivas: TBox

- Tbox: contiene declaraciones terminológicas generales. Vocabulario de un dominio de aplicación en función de: Conceptos, Roles, etc. Son de dos tipos.
 - Definición de concepto

$$A \equiv C$$

Ejemplo

$$\text{Mujer} \equiv \text{Persona} \sqcap \text{Femenino}$$

$$\text{Madre} \equiv \text{Mujer} \sqcap \exists \text{tiene_hijo. Persona}$$

- Axiomas descriptivos de roles, etc. $C_1 \subseteq C_2$

Ejemplo

$$\exists \text{tiene_hijo. Persona} \subseteq \text{Persona}$$

Lógicas descriptivas: ABox

- Abox: contiene aserciones sobre elementos y relaciones concretas del dominio. *Es decir, son aserciones acerca de individuos usando vocabulario.* Dos tipos:
 - Instancias de conceptos

$$o \equiv C$$

Ejemplo

Moby – Dick $\equiv$ Ballena

Juan : Hombre $\cap \exists$ tiene – hijo

- Instancias de axiomas

$$(o_1, o_2) : R$$

Ejemplo

(Ana, Juan) : tiene_hijo

Formalizar en DI y luego en owl dl

- Definición de conceptos.
 - El pasto y los arboles son plantas. Las hojas son parte del árbol, pero existen otras partes de un árbol que no son hojas. Un perro debe comer al menos huesos. Una oveja es un animal solo debe comer pasto. Una jirafa es un animal que solo debe comer hojas. Las vacas locas solo se alimenta de cerebros que pertenecen a las ovejas.
- Restricciones:
 - Animales son disjuntos con plantas.
- Propiedades:
 - Comer es aplicado a los animales y su inverso es comido_por.
- Individuos
 - Tom
 - Flossie es una vaca
 - Rex es un perro y es una mascota de Mick
 - Fido es un perro
 - Tibbs es un gato

Formalizar en DI y luego en owl dl

1. El pasto y los arboles son plantas.
2. Las hojas son parte del árbol, pero existen otras partes de un árbol que no son hojas.
3. Un perro debe comer al menos huesos.
4. Una oveja es un animal y solo debe comer pasto.
5. Una jirafa es un animal que solo debe comer hojas.
6. Una vaca loca es una vaca que se alimenta de cerebros que son parte de las ovejas.
7. Animales o parte de animales son disjuntos con plantas o parte de plantas.

- Propiedades:

- Comer es aplicado a los animales y su inverso es comido_por.

1. $Pasto \subseteq Planta$

1. $Arboles \subseteq Planta$

2. $Hojas \subseteq Arboles$

3. $Perro \subseteq Animal \cap \exists come.Huesos$

4. $Oveja \subseteq Animal \cap \forall come.Pasto$

5. $Jirafa \subseteq Animal \cap \forall come.Hojas$

6. $VacaLocas \subseteq vacas \cap \forall come(cerebro \subseteq Oveja)$

7. $Animal \subseteq \neg Planta$

Formalizar en DI y luego en owl

1.1.(LD). $Pasto \subseteq Planta$.

(OWL)rdf : subClassOf (Pasto,Planta)

1.2.(LD). $Arboles \subseteq Planta$.

(OWL)rdf : subClassOf (Arboles,Planta)

2.(DL) $Hojas \subseteq Arboles$

(OWL)owl : subClassOf (Hojas,Arboles)

3.(DL) $Perro \subseteq Animal \cap \exists come.Huesos$

(OWL)owl : subClassOf (Perro,(intersection(Animal,someValuesFrom(come,Huesos))))

4.(DL) $Oveja \subseteq Animal \cap \forall come.Pasto$

(OWL)owl : subClassOf (Animal,intersectionOf (Animal,owl:someValuesFrom(come,Pasto)))

5.(DL) $Jirafa \subseteq Animal \cap \forall come.Hojas$

(OWL)owl : subClassOf (Animal,intersection(Animal,owl:allValuesFrom(come,Hojas))

6.(DL) $VacaLocas \subseteq Vacas \cap \forall come(cerebro \subseteq Oveja)$

(OWL)owl : subClassOf (Vacas,intersection(Vacas,owl:allValuesFrom(come,Cerebro))

owl : subClassOf (Cerebro,Oveja)

7.(DL) $Animal \subseteq \neg Planta$

(OWL)owl : disjointWith(C,D)

Resultados

1. $Pasto \subseteq Planta$

1. $Arboles \subseteq Planta$

2. $Hojas \subseteq Arboles$

3. $Perro \subseteq Animal \cap \exists come.Huesos$

4. $Oveja \subseteq Animal \cap \forall come.Pasto$

5. $Jirafa \subseteq Animal \cap \forall come.Hojas$

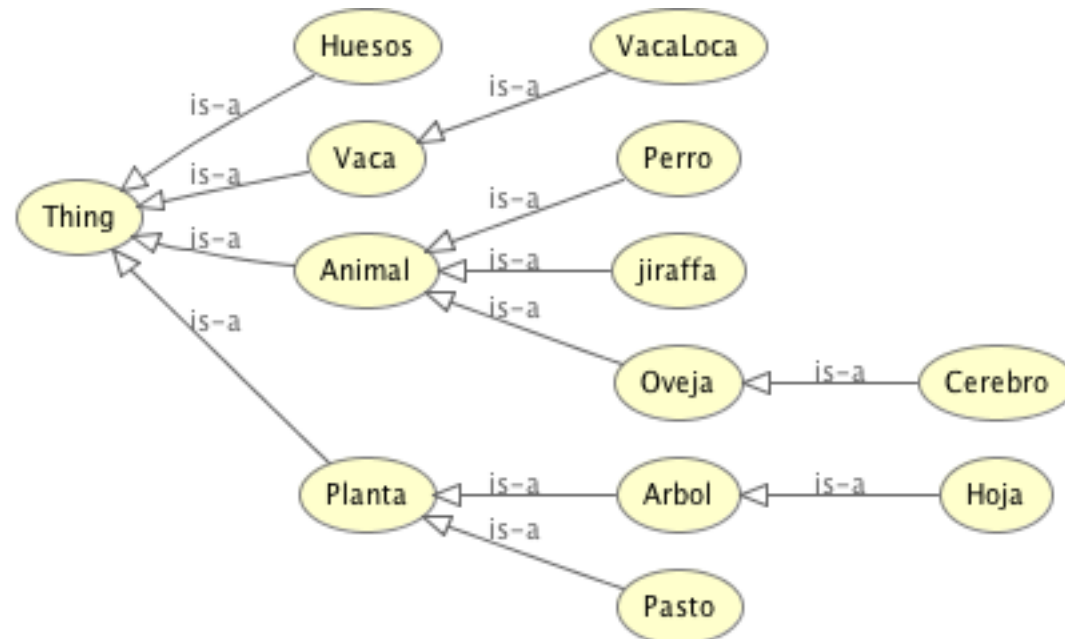
6. $VacaLocas \subseteq vacas \cap \forall come(cerebro \subseteq Oveja)$

7. $Animal \subseteq \neg Planta$

Asserted class hierarchy:



- ▼ ● Thing
 - Animal
 - Arbol
 - Hoja
 - Huesos
 - Oveja
 - Pasto
 - Plantas
 - Vacas
 - VacasLocas
 - jirafa



Lenguajes ontológicos: Owl y Reglas

El lenguaje basado en reglas para la Web Semántica

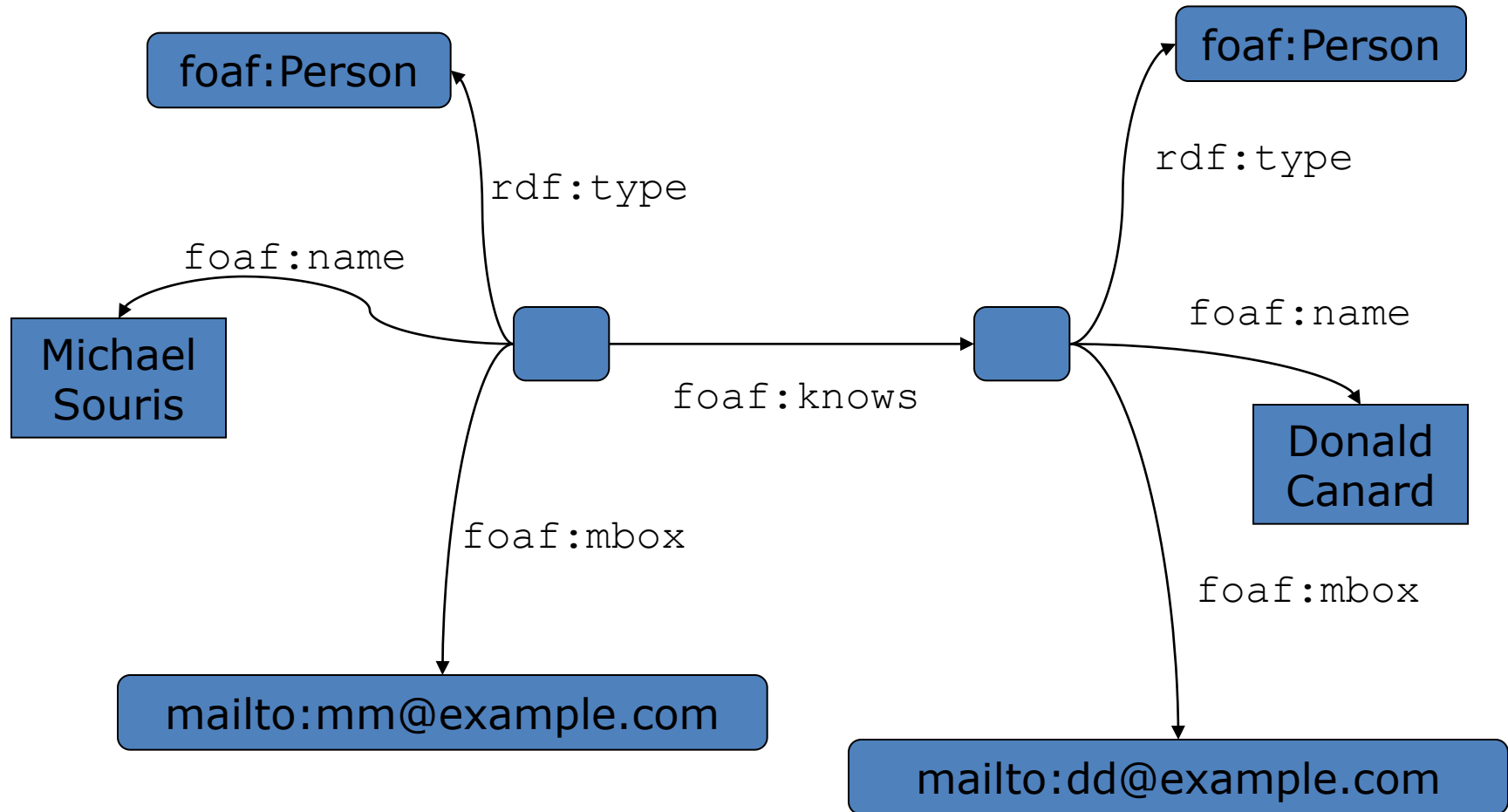
- ✓ Combinando OWL DL y OWL Lite con el lenguaje RuleML
- ✓ Las reglas están compuestas por una implicación entre un antecedente (cuerpo) y un consecuente (cabecera), que se interpreta de la siguiente forma:
 - ❖ Si las condiciones especificadas en el antecedente se cumplen, entonces las condiciones indicadas en el consecuente deben también verificarse

Aplicaciones: Redes Sociales

- ¿Cómo encontrar personas:
 - con intereses similares a los de uno?
 - relacionadas a alguien?
 - autoridades en un tópico?
 - etc.
- Redes sociales:
 - Información personal en la Web semántica.
 - Enlaces a información personal de personas conocidas.

Aplicaciones:

Amigo De Un Amigo, Brickley & Miller



Aplicaciones:

FOAF

```
<foaf:Person>
```

```
  <foaf:name>Peter Parker</foaf:name>
```

```
  <foaf:gender>Male</foaf:gender>
```

```
  <foaf:title>Mr</foaf:title>
```

```
  <foaf:givenname>Peter</foaf:givenname>
```

```
  <foaf:family_name>Parker</foaf:family_name>
```

```
  <foaf:homepage rdf:resource="http://www.peterparker.com"/>
```

```
  <foaf:weblog rdf:resource="http://www.peterparker.com/blog"/>
```

```
</foaf:Person>
```

Aplicaciones:

Publicación de datos FOAF

- ¿Cómo mis datos FOAF pueden ser conocidos por otros usuarios?
 - Soy “conocido” por alguien “conocido”:
 - Vía “FOAF:knows”
 - Publicando mi info en **FOAF Bulleting Board**:
 - Sitio con enlaces a decenas de archivos FOAF
 - Incluyendo en mi sitio Web un enlace a mi archivo FOAF:
 - `<link rel="meta" type="application/rdf+xml" title="FOAF" href="foaf.rdf" />`

Aplicaciones: herramientas para FOAF

- FOAF Explorer
 - Vista HTML de FOAF
- FOAFNaut
 - Visualización de relaciones entre personas
- Plink
 - Sitio de redes sociales
- FOAFBot, Whwhwhwh
 - Interfaces IRC sobre FOAF

Aplicaciones:

Búsqueda Semántica

- Pediatras que vivan a menos de 10 cuadras de mi casa
- Hoteles 3 estrellas en Viña del Mar que estén cerca del casino
- Carreras del área biología en Universidad de Cauca
- Nombre y dirección postal de profesores de escuelas primarias de Bogotá

Aplicaciones: Búsqueda Semántica

(Guha, McCool, Miller)



Aplicaciones:

– Tabulator: navegador Web Semántica

- interacción genérica árbol
- modelos

The Tabulator Project

testDataset

- ▶ Data on location of libraries mostly in the UK
- ▶ Martin Hepp
- ▶ Semantic Web Coordination Group plan includes calendar info

TAGmobile road trip BOS->Amerst: photo locations

mentions

- ▶ Record

ColorPicture

is mentioned in

- ▶ TAGmobile road trip BOS->Amerst: photo

is type of

- ▶ <http://www.w3.org/2004/lambda/Pictures/i>
- ▶ <http://www.w3.org/2004/lambda/Pictures/i>
- ▶ <http://www.w3.org/2004/lambda/Pictures/i>

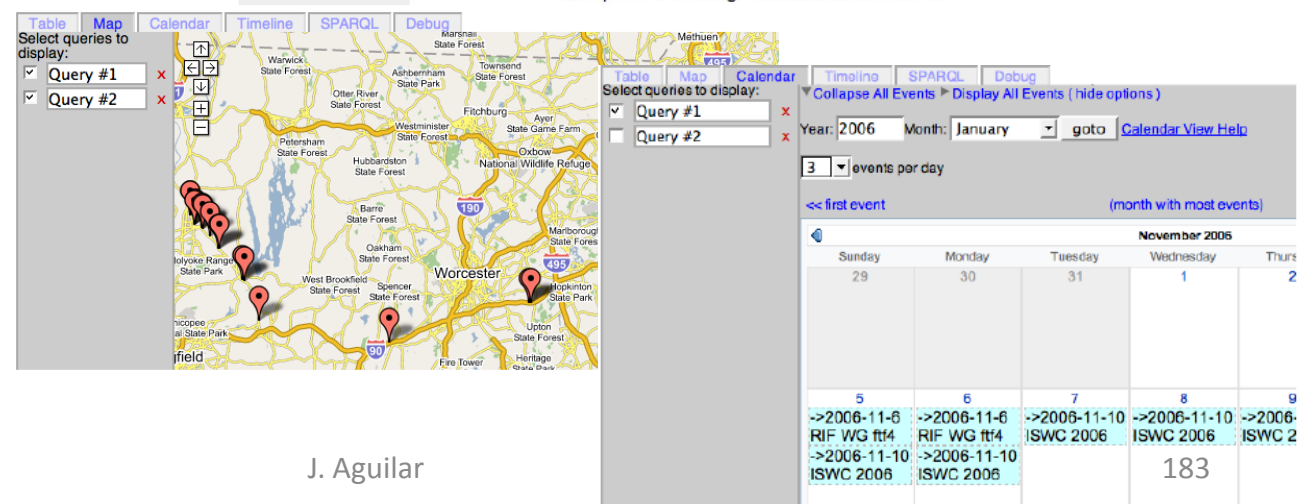


Table **Map** **Calendar** **Timeline** **SPARQL** **Debug**

Select queries to display:

- ☒ Query #1
- ☒ Query #2

Table **Map** **Calendar** **Timeline** **SPARQL** **Debug**

Select queries to display:

- ☒ Query #1
- ☐ Query #2

Calendar

Collapse All Events Display All Events (hide options)

Year: 2006 Month: January goto [Calendar View Help](#)

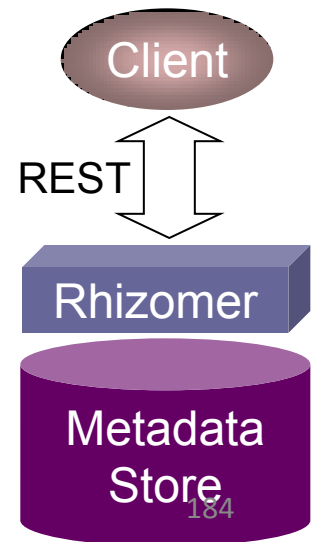
3 events per day

<< first event (month with most events)

November 2006				
Sunday	Monday	Tuesday	Wednesday	Thursday
29	30	31	1	2
5 ->2006-11-6 RIF WG tt4 ->2006-11-10 ISWC 2006	6 ->2006-11-6 RIF WG tt4 ->2006-11-10 ISWC 2006	7 ->2006-11-10 ISWC 2006	8 ->2006-11-10 ISWC 2006	9 ->2006-11-10 ISWC 2
			183	

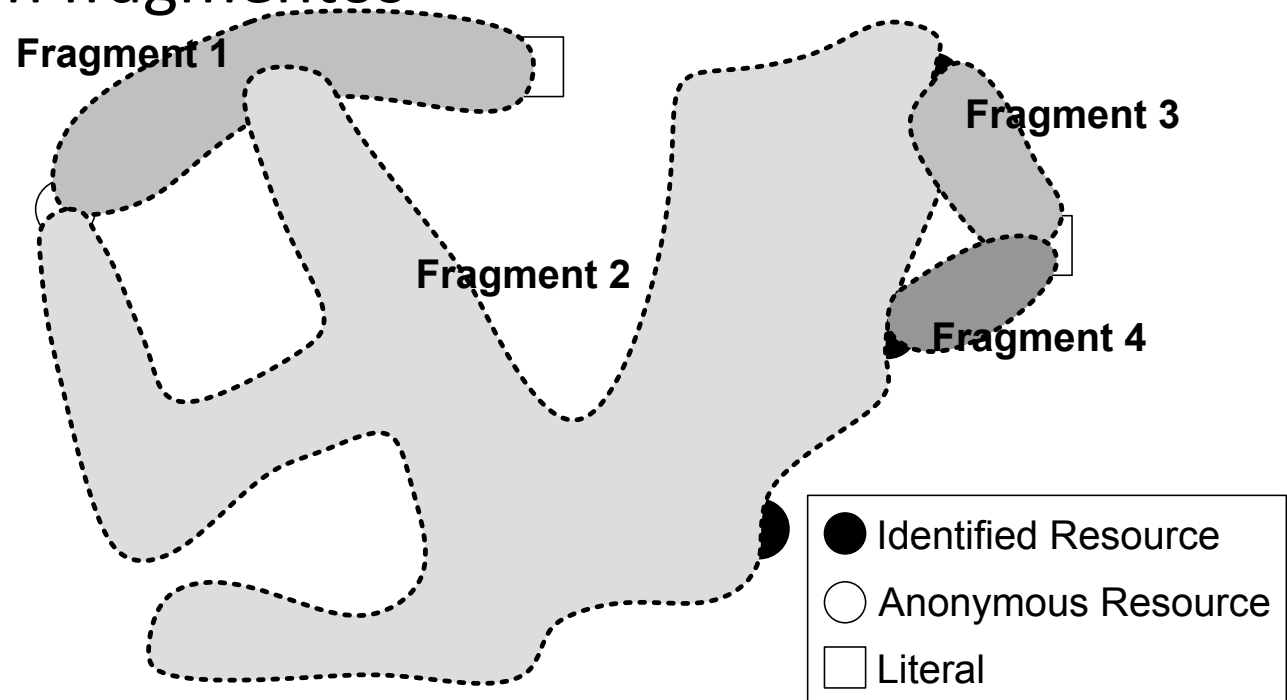
Aplicaciones:

- **Rhizomer**: modelo genérico Web “clásica”
 - Hábito interacción con HTML
 - enlaces, tablas, formularios,...
 - Web Semántica → HTML + Web 2.0
 - Abstracción almacén RDF (Sesame, Jena,...)
 - Servicio web, REST (HTTP):
 - GET: consultar
 - PUT: modificar
 - POST: crear nuevo
 - DELETE: borrar



Aplicaciones:

- **Rhizomer**
 - Almacén grafo
 - Navegación fragmentos



Buscadores en la WS

- Falcons: <http://iws.seu.edu.cn/services/falcons/objectsearch/index.jsp>
- Sindice: <http://sindice.com>
- Watson: <http://watson.kmi.open.ac.uk/WatsonWUI>
- Lexxe: <http://www.lexxe.com/>
- MicroSearch (Yahoo): <http://www.yr-bcn.es/demos/microsearch/search.do?p=Peter+Mika&y=Search&fr=lo>
- SWSE: <http://www.swse.org>
- Semantic WebSearch: <http://www.semanticwebsearch.com/query>
- URI Search: <http://dev.uriqr.com>
- Hakia: <http://www.hakia.com>
- Worio: <http://www.worio.com>
- Wolframalpha: <http://www.wolframalpha.com>
- PowerSet: <http://www.powerset.com>

D I en la WS

OJOmicroformatos OJO mi

OJO **YAHOO! SEARCH** **BLOG** A look i

[Gratis Alta en buscadores](#)
Alta en 5.000 buscadores grat
500 más cada mes. Popularid

« [A New Research Lab in Haifa](#) | [Main](#) | [A Search-Sized Bite Out of the Big Apple](#) »

Yahoo! + Web Se
artículo de Javier Casares pub

Yahoo! se está poniendo la
mejorando la relevancia de
grandes que de soporte a l

Para ello ha aplicado dos pi
en el código fuente y que
y su forma de expresión, a
ayudará a la accesibilidad,
programas para invidentes
búsqueda.

La segunda [hace referenci](#)

MARCH 13, 2008

The Yahoo! Search Open Ecosystem

A few weeks ago, we began [talking](#) about the new Yahoo! Search open platform. Today, we're releasing more details about two important components of the initiative – the developer platform as well as our support of a number of [semantic web](#) standards.

The Data Web in Action

While there has been remarkable progress made toward understanding the semantics of web content, the benefits of a data web have not reached the mainstream consumer. Without a killer semantic web app for consumers, site owners have been reluctant to support standards like [RDF](#), or even [microformats](#). We believe that app can be web search.

By supporting semantic web standards, Yahoo! Search and site owners can bring a far richer and more useful search experience to consumers. For example, by marking up its profile pages with microformats, [LinkedIn](#) can allow Yahoo! Search and others to understand the semantic content and the relationships of the many components of its site. With a richer understanding of LinkedIn's structured data included in our index, we will be able to present users with more compelling and

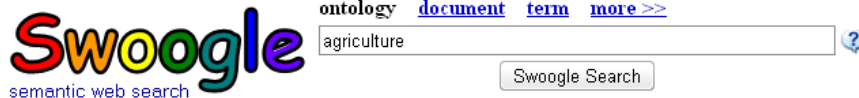
Yahoo indexa microformatos (Marzo de 2008): [hCard](#), [hCalendar](#), [hReview](#), [hAtom](#), [XFN](#), [Dublin Core](#), [Creative Commons](#), [FOAF](#), [GeoRSS](#), [MediaRSS](#), [RDFa](#) y [eRDF](#)

Buscadores de ontologías / vocabularios

Want more results? [Login](#)



Want more results? [Login](#)



[RDF version](#)

list ontologies matching ontology search

1 - 10 of total 177 results for **agriculture** in 1.14 seconds

sort by | [date](#) | [triple](#) |

<http://morpheus.cs.umbc.edu/aks1/ontosem.owl>

[DEF] , agricultural-product, agricultural-product-of, **agriculture**, agrobiolgy, agyrophobia, ailuromania
SemanticWebDocument, RDFXML, 2005-08-25, 3M, ontoRatio(1.00), [metadata](#), [cached](#)

[manual](#) o

<http://tap.stanford.edu/data/>

SemanticWebDocument, RDFXML, 2006-03-24, 15M, ontoRatio(0.14), [metadata](#), [cached](#)

This v

<http://semweb.mcdonaldbradley.com/OWL/Cyc/FreeToGov/060704/FreeToGovCyc.owl>

[DEF] , **Agriculture**, AgricultureDepartment, Agrobacterium, Aid, Aiding, AidingAllies, Ailment
SemanticWebDocument, RDFXML, 2004-08-06, 17M, ontoRatio(0.70), [metadata](#), [cached](#)

<http://www.daml.org/2001/12/factbook/factbook-ont>

[DEF] , **Agriculture**, AgricultureProduct, Aid, Airport, AirportBreakdown, And, Annually, Area, Assessment, Band
SemanticWebDocument, RDFXML, 2003-06-02, 48K, ontoRatio(1.00), [metadata](#), [cached](#)

<http://www.daml.org/2003/09/factbook/factbook-ont>

[DEF] , **Agriculture**, AgricultureProduct, Aid, Airport, AirportBreakdown, And, Annually, Area, Assessment, Band
SemanticWebDocument, RDFXML, 2005-09-21, 52K, ontoRatio(1.00), [metadata](#), [cached](#)

J. Aguilar

Una aplicación en español-ar:

TemaTres

TemaTres

Buscar

Menú

- Inicio
- Lista sistemática
- Lista alfabética
- Acerca de...

Lista sistemática

[=] Bienestar social

- . Bienestar social
- [TE estado de bienestar](#)
- [TE política social](#)

[XML / RDF](#)
[Zthes](#)
[SKOS-Core](#)

Mi cuenta

Correo electrónico:

Clave:

Ver ejemplos: <http://www.r020.com.ar/tematres/#ej>

Conclusiones

- La Web Semántica es aún una visión
 - que ha comenzado a caminar pero aún tiene mucho camino por delante
- Comienzan a haber herramientas apropiadas y estándares emergentes

Actualmente:

- **No está:**
 - En los motores de búsqueda
 - En bases de datos
 - En la mayor parte de la web “real”
- **Empieza a estar:**
 - En los sitios web que usan estándares de manera estricta
 - En algunos repositorios
 - En un reducido (pero selecto) número de sitios web relacionados con la Administración y/o con iniciativas de carácter científico o cultural

Conclusiones

- **Actualmente:**
 - Un nuevo formato universal de datos: XML
 - Una fuerte impulso al uso de estándares Web
 - Un renovado debate sobre el uso, definición y alcance de los metadatos
 - Un nuevo formato universal para expresar metadatos: RDF con aplicación a tesauros y lenguajes documentales
- **En el futuro:**
 - ¿Servidores de ontologías?
 - ¿Nuevos sistemas de búsqueda y acceso a la información?
 - ¿Una nueva generación de repositorios, bibliotecas digitales y sistemas de información?

Posibilidades que da

- Recuperación de información mejorando las posibilidades de los motores de busca
- Catalogación para describir el contenido y las relaciones entre los contenidos de una página o sede web, una Intranet, o una biblioteca entera
- Los agentes de software inteligentes se pueden beneficiar de RDF para encontrar y "entender" más fácilmente el significado de la información que procesan y detectar el nivel de relevancia.
- Calificación de contenidos. Para facilitar la valoración del contenido de artículos, páginas web, etc.
- Para describir los derechos de propiedad de las páginas web, para expresar las *preferencias de privacidad* de un usuario, las *políticas de privacidad* de una sede Web
- Para facilitar la seguridad al comercio electrónico unido con las firmas digitales.

Web Semántica en la Actualidad

