

Sistemas de Operación III
Tema IV: Sistemas
Distribuidos basados en
Web

Prof. Yudith Cardinale
Sept-dic 2011

Aplicaciones en Internet

- ◆ La aplicación distribuida más conocida es la World Wide Web o la Web
- ◆ Se trata de un sistema distribuido de servidores HTTP y clientes WEB para acceder a documentos vinculados.
- ◆ Con un navegador web, un usuario visualiza sitios web compuestos de páginas web que pueden contener texto, imágenes, videos u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces.

Aplicaciones en Internet

- ◆ La web nació gracias a Tim Berners-Lee a finales de 1990 en el CERN, el laboratorio Europeo de Física de partículas de Suiza.

Aplicaciones en Internet

- ◆ Idea original: permitir que un grupo numeroso de investigadores, dispersos geográficamente tuviera acceso a documentos compartidos.
- ◆ Vinculando los documentos entre sí, fue fácil integrarlos desde diferentes proyectos en un nuevo documento sin necesidad de realizar cambios centralizados.

Web 1.0 y sus características

- Etapa comprendida entre 1991-2003.
- Es un tipo de Web estática.
- Es sólo de lectura. El usuario no puede interactuar con el contenido de la página (nada de comentarios, respuestas, citas y etc.), estando totalmente limitado a lo que el Webmaster publica.
- Los Website eran tan estáticos que casi nunca se actualizaban.
- Uso de tecnologías asociadas a etiquetas y formulario en HTML con imágenes y botones en formato GIF.
- Los datos que se insertaban en los formularios HTML sólo eran enviados vía email a través de un cliente de correo electrónico que el usuario tenía instalado en su PC.

Web 2.0 y sus características

- Etapa comprendida en desde 2003- al presente.
- Es un tipo de Web dinámica
- Uso de tecnologías como HTML, PHP, ASP, Bases de Datos, Javascript, DHTML, XML, Soap, etc.
- Los usuarios de la Web pueden contribuir. Publican las informaciones y realizan cambios en los datos.
- Programas y herramientas en línea, sin necesidad de instalar nada en el PC ya que están en Internet.
- Orientada a compartir datos, donde todos aportan.
- Programas para PDAs y teléfonos celulares.

Web 3.0 y sus características

- La transformación de la red en una base de datos (Data Web).
- Uso de las tecnologías de inteligencia artificial.
- El desarrollo de programas que puedan razonar (Web semántica y SOA).
- Maximización de la interconexión (Omnipresencia).

Servicio Web

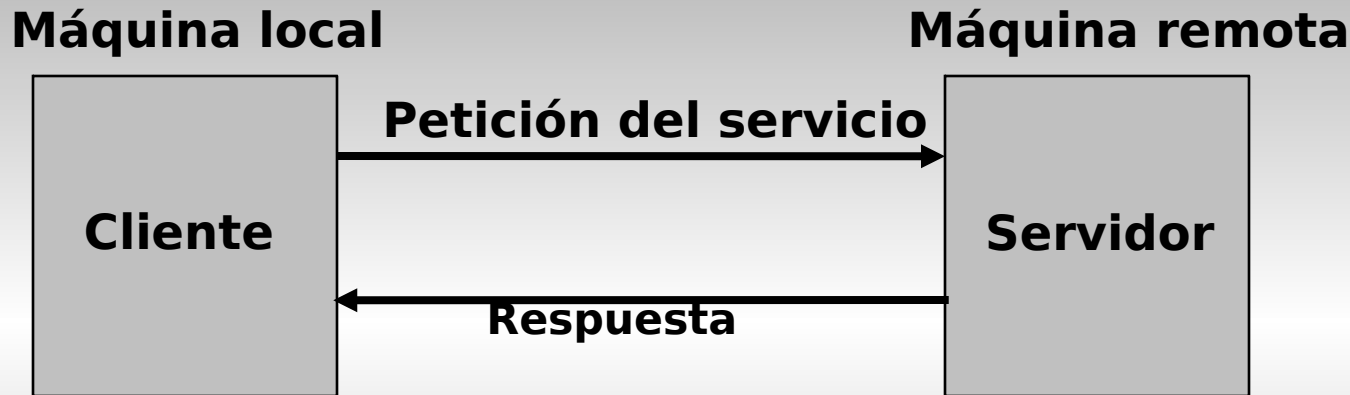
- ◆ Unidad interoperable y reusable que no tiene llamadas “empotradas” a otros servicios.
- ◆ Proveen interacción humano-máquina
- ◆ Software débilmente acoplado
- ◆ Funcionan internamente usando lenguajes tradicionales: java, .net, c#, fortran, ...
- ◆ Autodescriptivos (wsdl), tanto funcionalmente como de datos: descripción semántica
- ◆ No jerárquicos

Servicios Web

- ◆ Independientes de la implementación del propio servicio, a nivel de: lenguajes, sistemas de operación/plataforma, fuentes de bases de datos
- ◆ Proveen ejecución remota con el enfoque cliente-servidor
 - ◆ Otras experiencias: RPC, RMI, CORBA
- ◆ Orquestación o composición
- ◆ Descubrimiento y composición dictado por los requerimientos del negocio

Servicios WEB: modelo de comunicación

Modelo Cliente/Servidor:



- Todos los servicios sobre Internet funcionan bajo este esquema
- El medio de envío para la petición/respuesta es la red
- Un servidor debe poder manejar varios usuarios concurrentemente

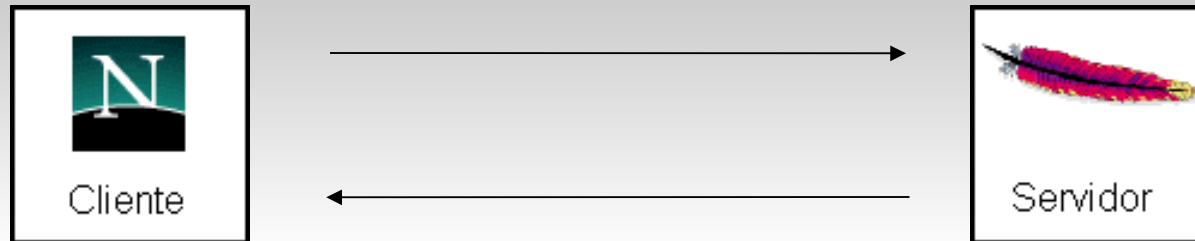
Servicios WEB: modelo de comunicación

Servicio HTTP: ejemplo del modelo Cliente/Servidor

- Los navegadores se comunican con los servidores Web por medio del protocolo HTTP (Hyper Text Transfer Protocol, RFC 2616).
- HTTP es un protocolo suficientemente ligero y rápido para ser utilizado en un sistema colaborativo de información de medios distribuidos. Es un protocolo genérico orientado a objetos, por lo cual se puede utilizar para diversos tipos de aplicaciones.
- HTTP fue creado en 1990 por Tim Berners-Lee, reemplazando en gran medida los sistemas Gopher.

Servicios WEB: modelo de comunicación

Servicio HTTP: ¿ Cómo funciona?

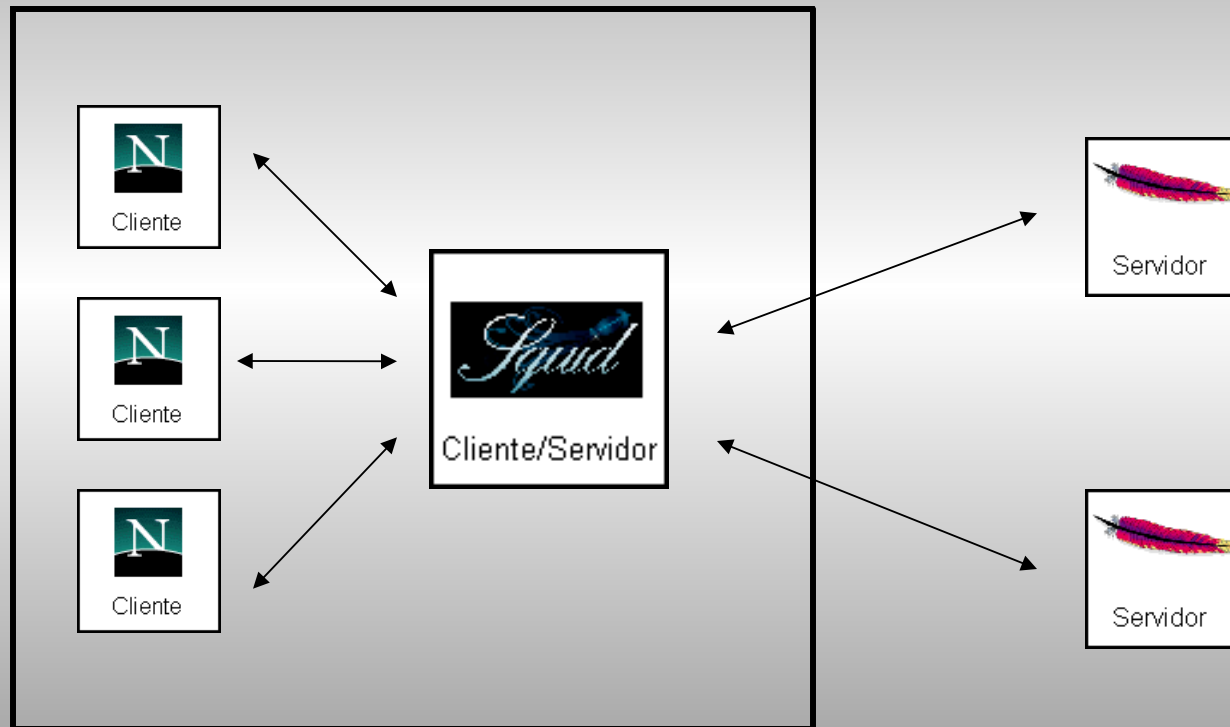


<http://www ldc.usb.ve/~yudith>

```
HTTP/1.0 200 OK
Server: Apache/2.0.16
Content-type: text/html
Last-Modified: Thu Jul 7 00:25:33 2001
Content-Length: 2003
<html><body>
...
</body></html>
```

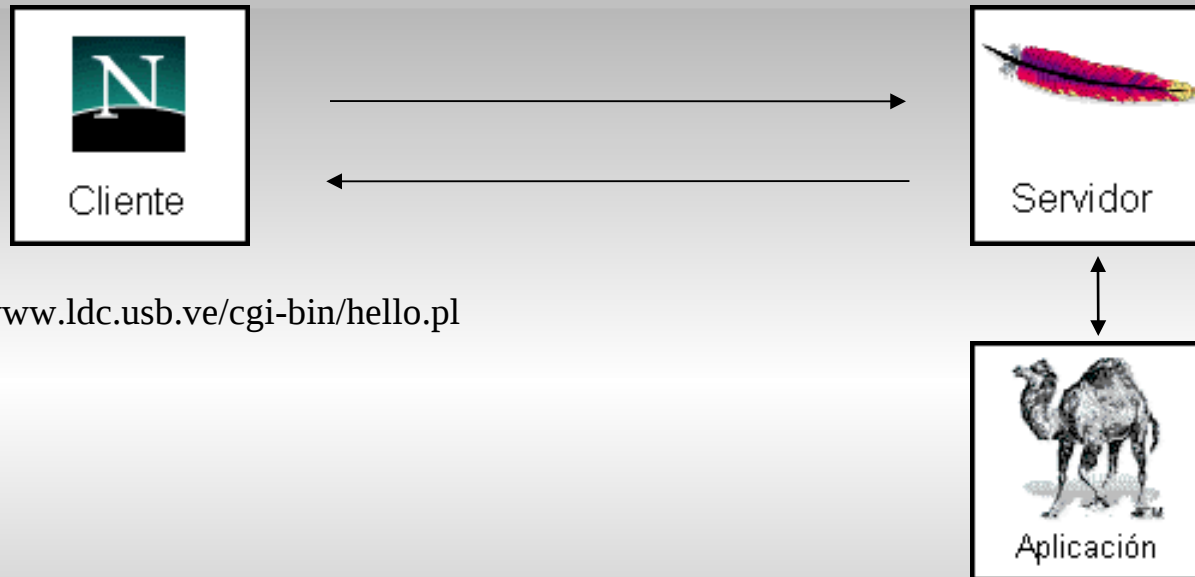
Servicios WEB: modelo de comunicación

Servicio HTTP: **PROXIES**



Servicios WEB: modelo de comunicación

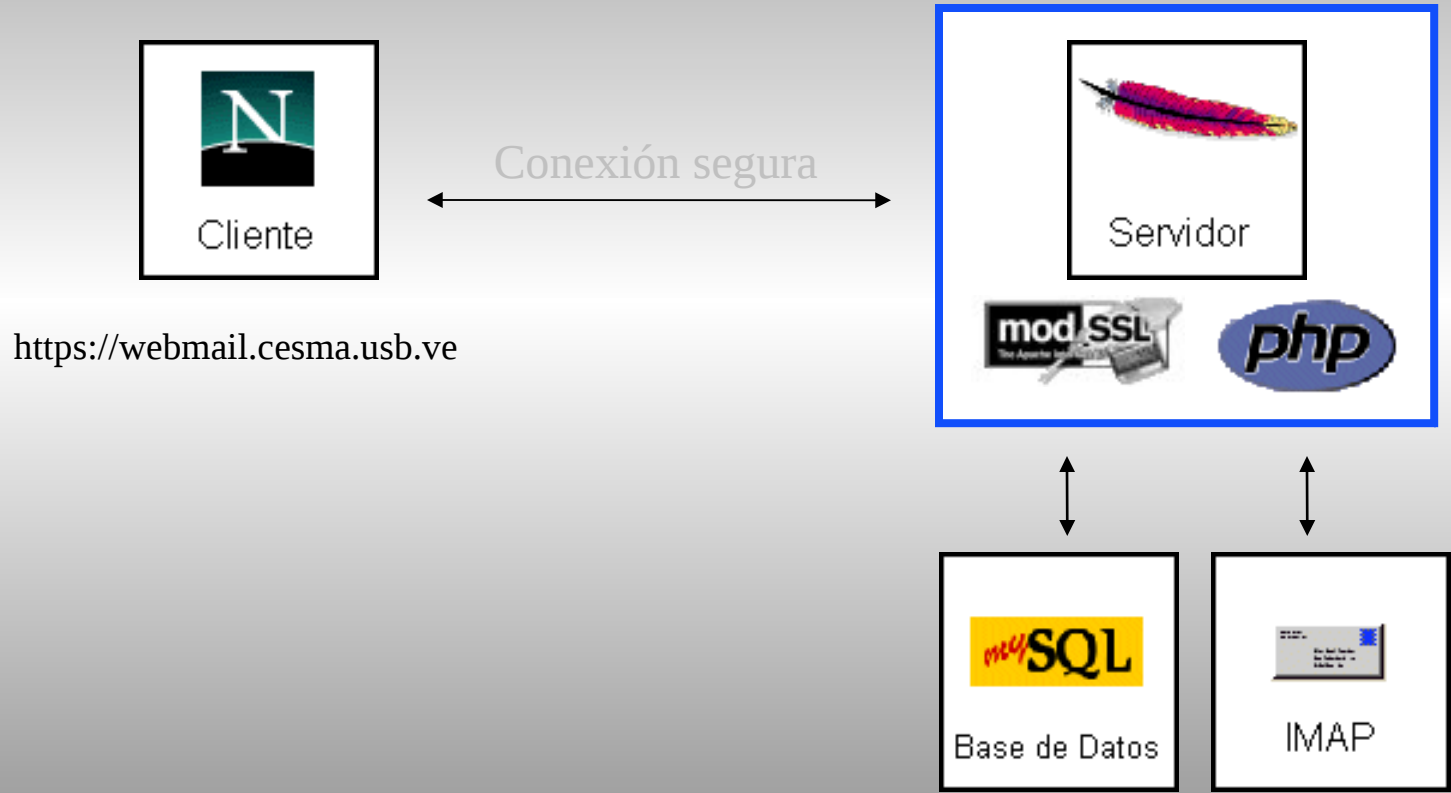
Servicio HTTP: **CGI o ASP o Servlets**



El servidor crea un nuevo proceso y coloca los valores apropiados en las variables de ambiente.
Se pasan los parámetros por stdin o por argumentos según el método del POST
Se ejecuta la aplicación CGI, los resultados se toman de stdout.

Servicios WEB: modelo de comunicación

Servicio HTTP: **Contenido dinámico usando módulos en el servidor**



Servicios WEB: modelo de comunicación

Servicio HTTP: **Problemas**

Problema	Solución
Causa gran congestión	Uso de conexiones persistentes
Es inseguro	Implementación de HTTPS
Es un protocolo sin estado	Uso de cookies, manejo de sesiones
Problemas de caching	Implementación de nuevos encabezados en HTTP/1.1

Motivación: Evolución de Servicios Web

The **Cloud of Linked Data**, using the Web to connect related data that was not previously linked!

Published Data are enhanced with semantics!

Standards to annotate and describe data:

XML, RDF, RDFS, OWL.

Standards to query data: SPARQL.

Ontologies representing almost any domain.

Hyperlink-based systems.

Protocols: http, uri, html

Documents and data were published

Arpanet: four servers connected

Files were transferred

Tools: ftp, telnet, e-mail

80's

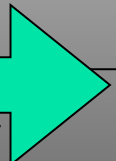
Universidad Simon Bolívar

90's

00's

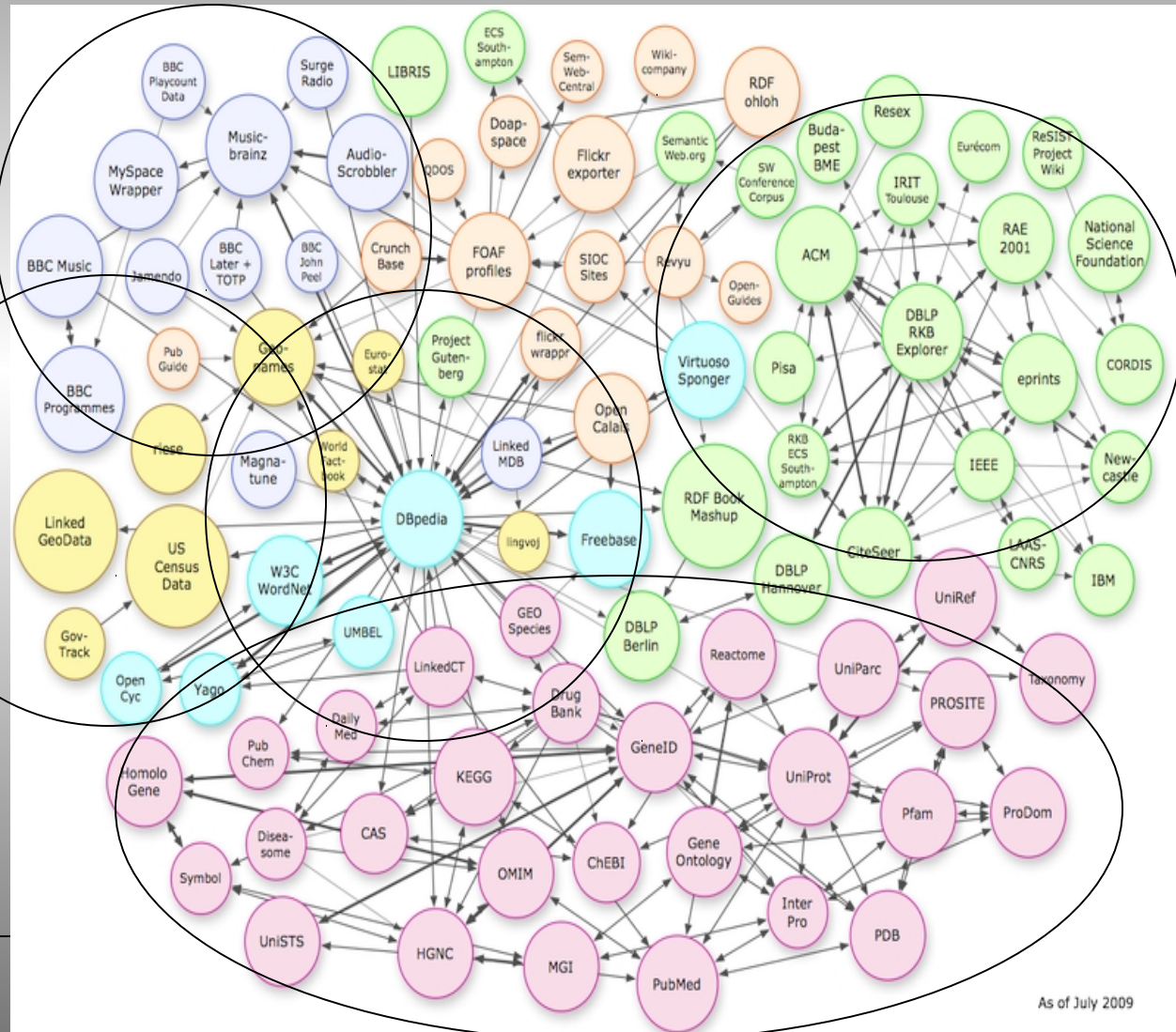
Prof. Yudith Cardinale

Now



Motivación: Evolución de Servicios Web

- **Explosion** in the number of:
 - Linking Open Data resources and databases
 - Different quality parameters.
 - **Controlled** vocabularies:
 - MeSH, GO, PO...
 - Highly interconnected data sources:
 - Different Sizes
 - Many links
 - Different in- and out-degrees, etc
- **Biological Web**: large datasets of linking data.
 - Genes, Diseases, Clinical Drugs, Proteins, and so on.



Motivación: Evolución de Servicios Web

- Explosión en el número de:

– Linking de Data

resources &

– Different qu

parameters.

– Cent

– Highly in
data sources

Differen

M

• Differen

degrees

Molecular databases **1170, 95** more
than **2008** and **110** more than the year before !

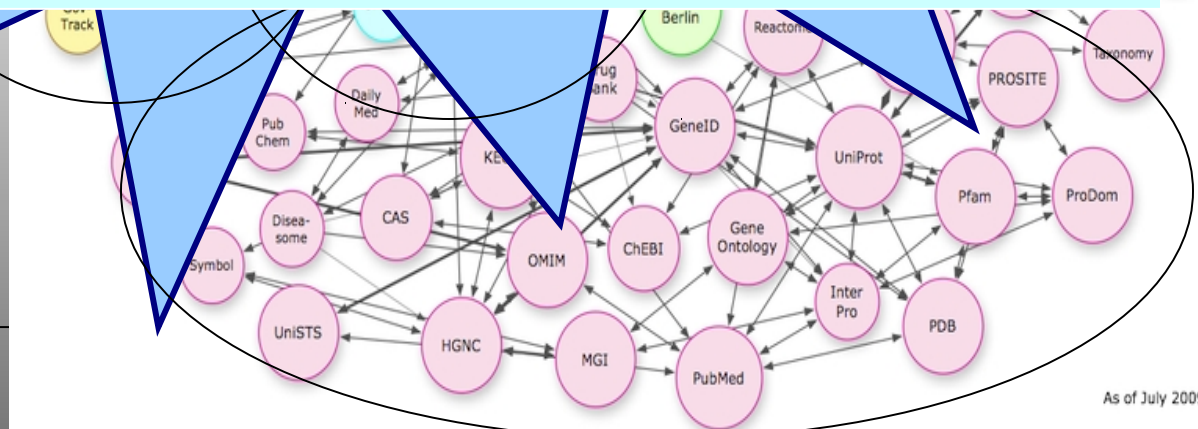
Services and tools published
by these databases follow a similar progression!

In October **2007**, Cloud of Linked Data
datasets consisted of over **two billion** RDF triples,
which were interlinked by over **two million** RDF links.

By May **2009** this had grown to **4.2 billion** RDF triples,
interlinked by around **142 billions** RDF links! Today
the Cloud of Linked Data has at least **13,112,409,691** triples.

• **Biological Web**: large
datasets of linking data.

- Genes, Diseases,
Clinical Drugs, Proteins,
and so on.



Motivación: Escenario actual

- ◆ **Internet**: repositorio de gran cantidad de información.
- ◆ **Billones** de fuentes de datos y de usuarios.
- ◆ Dado un dominio del saber, existen **cientos o miles** de fuentes de datos que mantienen datos relacionados con el mismo, algunos con **billones** de instancias.

Motivación: La Web

- ◆ La **Web** es **distribuida**:
 - ◆ Cada **nodo** es **autónomo** e **independiente**.
 - ◆ **No** existe un **control** central.
 - ◆ Cada **comunidad** usa su propio **vocabulario**.
 - ◆ **No** toda la información es **correcta** y **consistente o completa**.
- ◆ La **Web** es **dinámica**:
 - ◆ Contenido de las páginas cambia constantemente.
 - ◆ Nuevas páginas se añaden constantemente.

Motivación: La Web

- ◆ La **Web** es **masiva**:
 - ◆ Se estima:
 - ◆ 13 millardos de páginas descubiertas.
 - ◆ Esta cantidad se duplicará en un año.
- ◆ La **Web** es un **mundo abierto**:
 - ◆ Si algo no se encuentra, no se puede suponer que es falso.
- ◆ La **Web** está diseñada para ser entendida por personas y no por máquinas:
 - ◆ **Datos** son entendidos por la aplicaciones propietarias.
 - ◆ Es **difícil**
 - ◆ Intercambiar datos entre aplicaciones.
 - ◆ Agregar datos desde varias fuentes de datos.

La Web Semántica

- ◆ Próxima Generación de la Red.
- ◆ Extensiones a la Red actual.
- ◆ Hace el contenido de la Red accesible a sistemas de software, de forma que máquinas puedan entender los datos.
- ◆ *“The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities”*
(Tim Berners-Lee et al., Scientific American 2001)

La Web Semántica: Pre-condiciones

- ◆ Datos puedan intercambiarse.
 - ◆ XML, XML Schema
- ◆ Semántica de los datos sea explícita
 - ◆ RDF, Ontologías
- ◆ Propiedades de los datos y de los metadatos pueden ser inferidas.
 - ◆ Lógica-**debe ser suficientemente poderosa para describir propiedades complejas de los objetos pero no lo suficientemente compleja que se vuelva no decidible!**

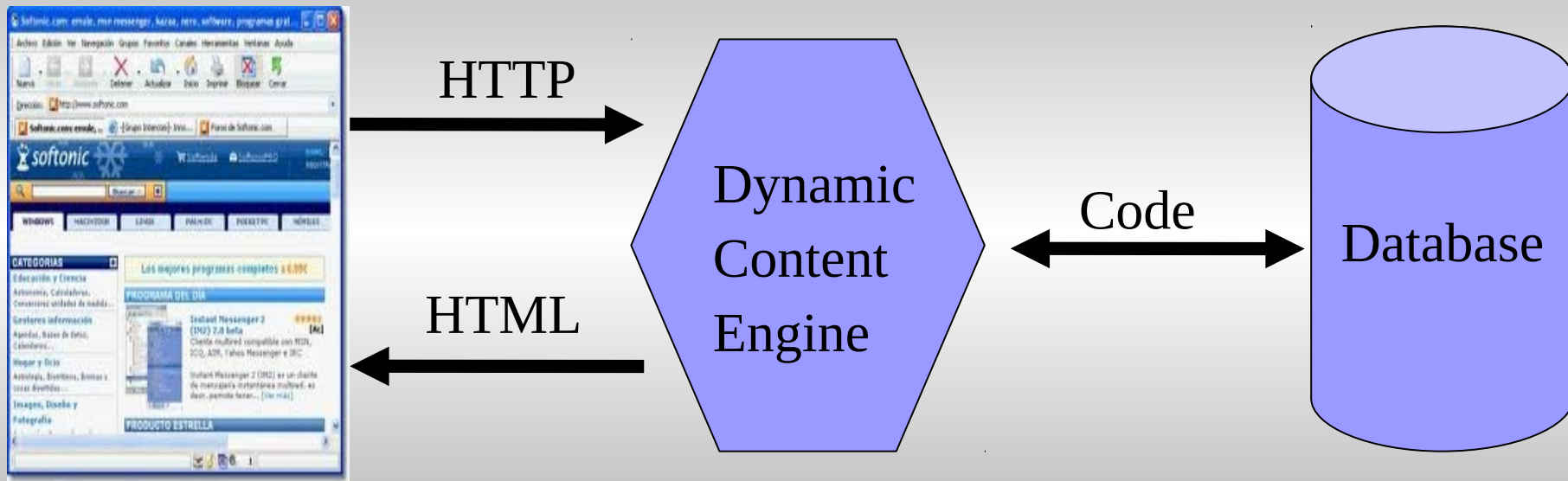
La Web Semántica: Requisitos

- ◆ Desarrollo de lenguajes que sean entendibles por máquinas.
- ◆ Desarrollo de herramientas que permitan acceder eficientemente a los datos y al conocimiento.
- ◆ Desarrollo de herramientas que hagan uso de estos lenguajes y permitan descubrir, acceder, presentar y mantener fuentes de datos.
- ◆ Desarrollar aplicaciones que ofrezcan un nuevo nivel de servicio a los humanos

La Web Semántica: Tecnologías

- ◆ XML (eXtensible Markup Language):
 - ◆ Lenguaje para el intercambio de datos en el Web y entre aplicaciones
 - ◆ Permite crear tags que pueden ser usadas por scripts o programas y poder generar funciones sofisticadas
 - ◆ El programador debe conocer los tags
- ◆ RDF (Resource Description Framework)
 - ◆ Expresa significado
 - ◆ Definido en tripletas (sujeto, verbo(predicado), objeto)
 - ◆ Las tripletas pueden escribirse en XML
 - ◆ Los sujetos y los objetos son identificados por URI (URLs)
 - ◆ Los verbos también pueden identificarse con URI
- ◆ Ontologías: OWL (Ontology Web Language)
 - ◆ Colección de información
 - ◆ Da el mismo significado a diferentes términos

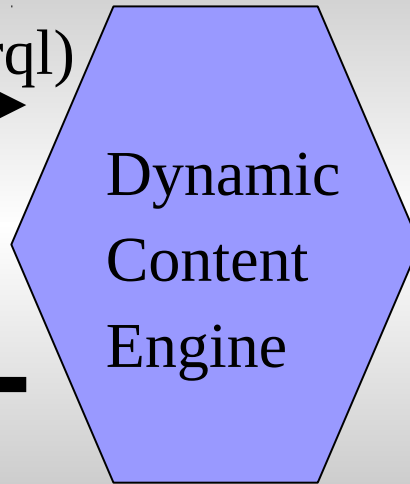
Aplicaciones Web Tradicionales



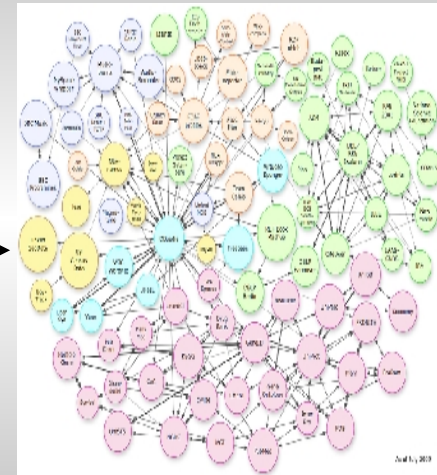
Aplicaciones Web Semántica



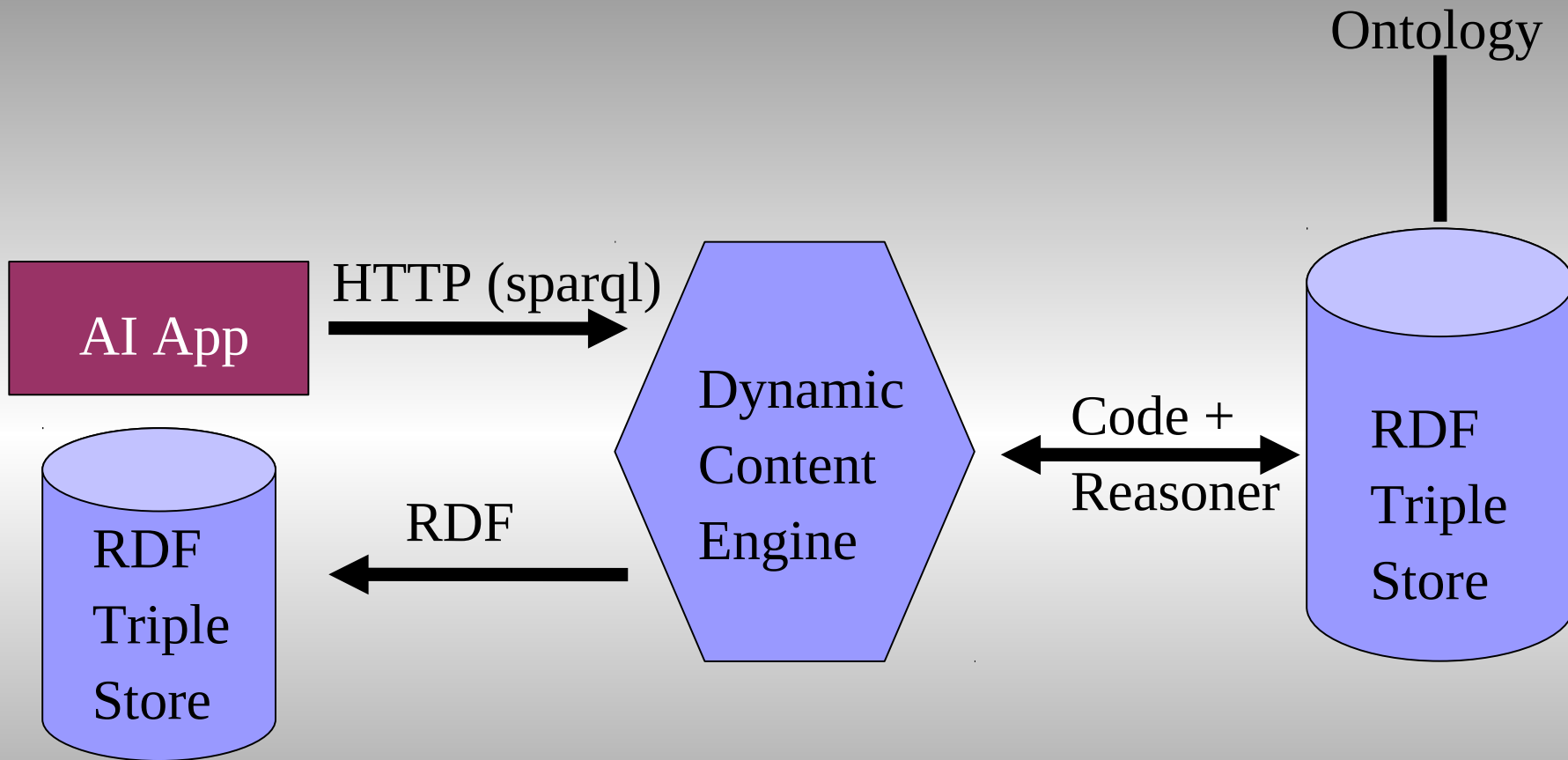
HTTP(Sparql)



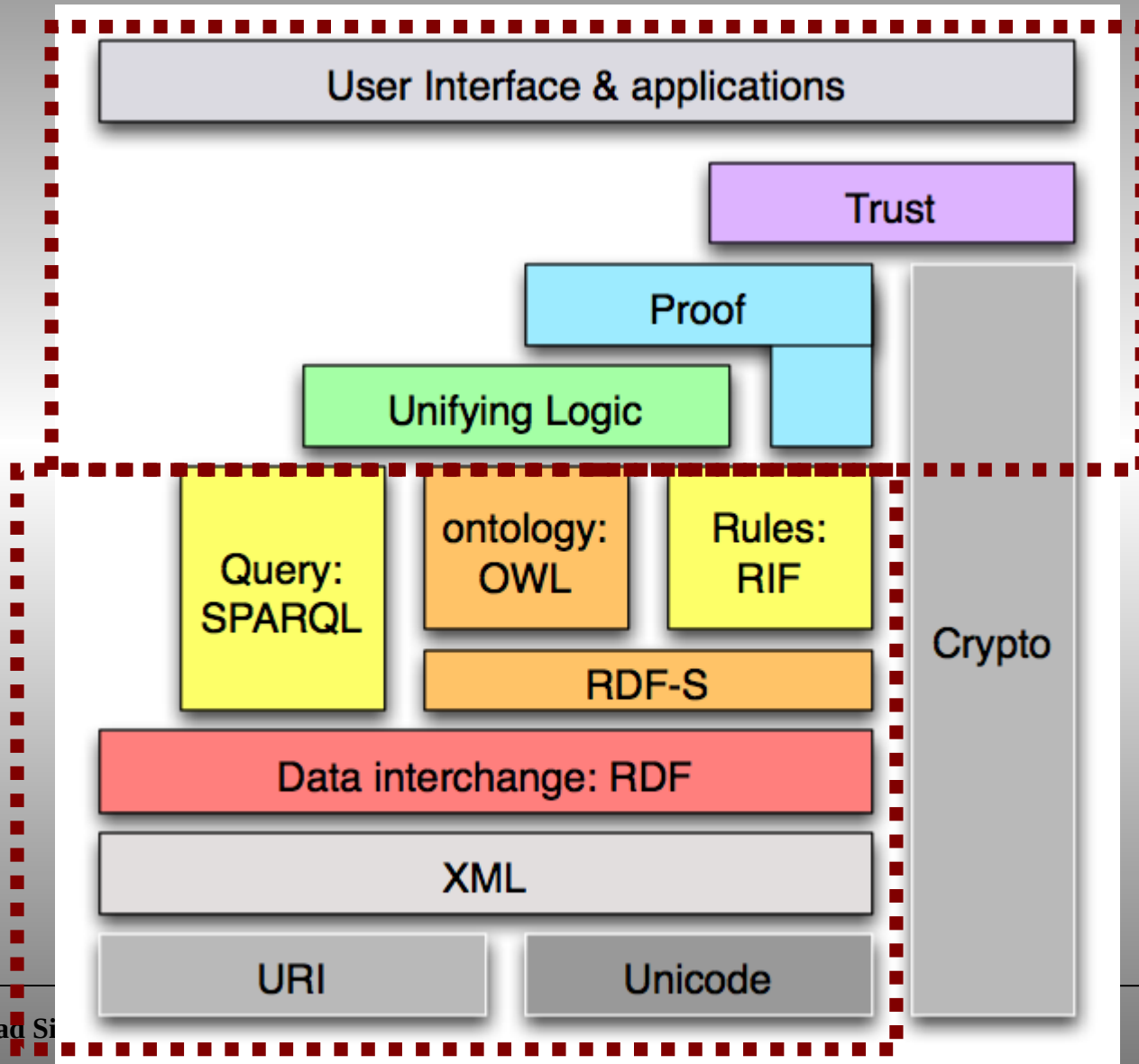
Code



Aplicaciones Web Semántica



Stack de la Web Semántica



Research activities

Standards activities

URI: Universal Resource Identifiers

- ◆ Identifican los recursos en la Web: Importante para la Web Semántica
- ◆ Son interpretados consistentemente en todos los contextos
- ◆ Asociar un URI con un recurso significa que “cualquiera” puede enlazarlo, referirlo, accederlo u obtener una representación particular del recurso.
- ◆ URLs es un ejemplo de URI: Identifica el recurso e incluye información sobre cómo y dónde accederlo (dependencia de la localización)

URI: Universal Resource Identifiers

- ◆ La sintaxis de un URL está determinada por:
 - ◆ Su esquema asociado o protocolo. El nombre del esquema es parte del URI (http, mailto, ftp, telnet, etc).
 - ◆ El nombre DNS del servidor que contiene el documento, aunque también es posible utilizar una dirección IP.
 - ◆ También se incluye el número de puerto; cuando se deja afuera se usa un puerto pre-establecido.
 - ◆ Finalmente, el nombre del documento.

Scheme	Host name	Port	Pathname
--------	-----------	------	----------

http :// 130.37.24.11 : 80 /home/steen/mbox

XML: Datos Semi-Estructurados

eXtensible Markup Language:

```
<db>
  <book>
    <title>Complete Guide to DB2</title>
    <author>Chamberlin</author>
  </book>
  <book>
    <title>Transaction Processing</title>
    <author>Bernstein</author>
    <author>Newcomer</author>
  </book>
  <publisher>
    <name>Morgan Kaufman</name>
    <state>CA</state>
  </publisher>
</db>
```

XML: Propiedades

- ◆ Lenguaje basado en
 - ◆ etiquetas.
 - ◆ modelos de datos semi-estructurados
- ◆ XML es flexible
 - ◆ Metalenguaje
 - ◆ Etiquetas pueden definirse
 - ◆ Adaptable a dominios/aplicaciones
- ◆ Separación de: Contenido, Estructura y Presentación
- ◆ Documentos son estructurados
 - ◆ Estructura de árbol
 - La estructura puede reflejar significado
 - ◆ Lenguajes para definir esquemas pueden ser usados para forzar cierta estructura
- ◆ Existen Múltiples Vocabularios

Contribuciones de XML a la Web Semántica

- Mejora a HTML en
 - Lenguaje dependiente del dominio.
 - Correspondencias entre XML y HTML (XSL)
 - Separa la estructura de los datos de la forma en que serán presentados.
- Permite auto-describir los datos.
- Estandariza el intercambio de datos y conocimiento.
- Permite la “integración” de datos con diferentes esquemas y propiedades.
- XML DTD's o XMLSchema para definir la estructura del documento.
 - Permite una validación mínima

Limitaciones de XML

- ◆ Muchas posibilidades diferentes de codificar un universo de discurso.
- ◆ No se dice nada sobre el significado de las estructuras.
- ◆ Difícil de entender otros documentos.
 - ◆ Se debe entender el significado de cada vocabulario.
 - ◆ No se logró una estandarización.

XML y The Semantic Web

- ◆ Datos puedan intercambiarse.
 - ◆ SI
- ◆ Semántica de los datos sea explícita
 - ◆ NO
- ◆ Propiedades de los datos y de los metadatos pueden ser inferidas.
 - ◆ NO

RDF(Resource Description Framework)

- ◆ Ayuda a crear relaciones entre datos al estilo web (web-like)
 - ◆ No es fácil en típicas bases de datos relacionales
- ◆ Modelo de datos de RDF
 - ◆ Tripletas (recurso, propiedad, valor)
 - ◆ Recurso (Sujeto)
 - ◆ Propiedad (Predicado)
 - ◆ Valor (Objeto)
- ◆ Sintaxis de RDF
 - ◆ Diferentes vocabularios de XML
 - ◆ [<http://www.w3.org/RDF/>]

RDF-Modelo de Datos

- ◆ Recurso
 - ▾ Ente del cual se habla
- ◆ Propiedad
 - ◆ Define relaciones del ente con otros entes o valores
- ◆ Afirmación
 - ◆ **Recurso** tiene **Propiedad** con **Valor(Objeto)**

Recursos, Propiedades y Objetos están identificados con un único Universal Resource Identifier (URI) => palabras en los documentos estén atados a una única definición que puede ser encontrada en el Web.

RDF-Ejemplo

◆ **Afirmación:**

- ◆ *“Ora Lassila es el creador del documento
<http://www.w3.org/Home/Lassila>”*

◆ **Estructura:**

- ◆ Recurso (sujeto)
<http://www.w3.org/Home/Lassila>
- ◆ Propiedad (predicado)
<http://www.schema.org/#Creator>
- ◆ Valor (objeto) *“Ora Lassila”*

RDF-Propiedades

- ◆ Existen clases primitivas
- ◆ Relaciones
 - ◆ SubClassOf
 - ◆ SubPropertyOf
- ◆ Instanciaciones de Clases
- ◆ Existe una definición formal de la semántica de RDF
 - ◆ Teoría de Modelos
 - ◆ Conjunto de Reglas de inferencia que permiten inferir nuevas tripletas

RDF-Desventajas

- ◆ Dos documentos pueden usar definiciones diferentes del mismo concepto y no se puede decidir que son equivalentes.
- ◆ Tiene problemas de ambigüedad en la definición de sus elementos.
- ◆ No se permite el manejo/definición de restricciones de integridad.
- ◆ Problemas de razonamiento y evaluación de consultas puede ser intratable y no decidible.

RDF y The Semantic Web

- ◆ Datos puedan intercambiarse.
 - ◆ SI
- ◆ Semántica de los datos sea explícita
 - ◆ Parcialmente. Extensiones de RDF Schema permite definir la semántica de built-in predicates.
- ◆ Propiedades de los datos y de los metadatos pueden ser inferidas.
 - ◆ Parcialmente

RDF Schema

- ◆ Lenguaje de descripción de vocabularios RDF
- ◆ No proporciona vocabularios específicos, sin facilidades para describir las clases y propiedades de un dominio específico.
- ◆ Las facilidades que proporciona RDFS se expresan como un vocabulario RDF, es decir, un conjunto especializado de recursos RDF con sus propios significados.
- RDFS puede verse como un sistema de tipos para RDF, similar al de los lenguajes orientados a objetos: recursos como ejemplares de una o más clases, jerarquía de clases, propiedades asociadas a clases...

Ontologías

- ◆ Conceptualización formal del mundo
- ◆ Lenguaje basado en Description Logics.
- ◆ Definen parte del mundo y permiten interacciones entre datos con diferentes formatos
- ◆ Es un documento que define formalmente las relaciones entre términos a través de:
 - ◆ Una taxonomía: clases de objetos y relaciones entre ellos
 - ◆ Reglas de inferencia: expresan más significado

Language de Ontologías

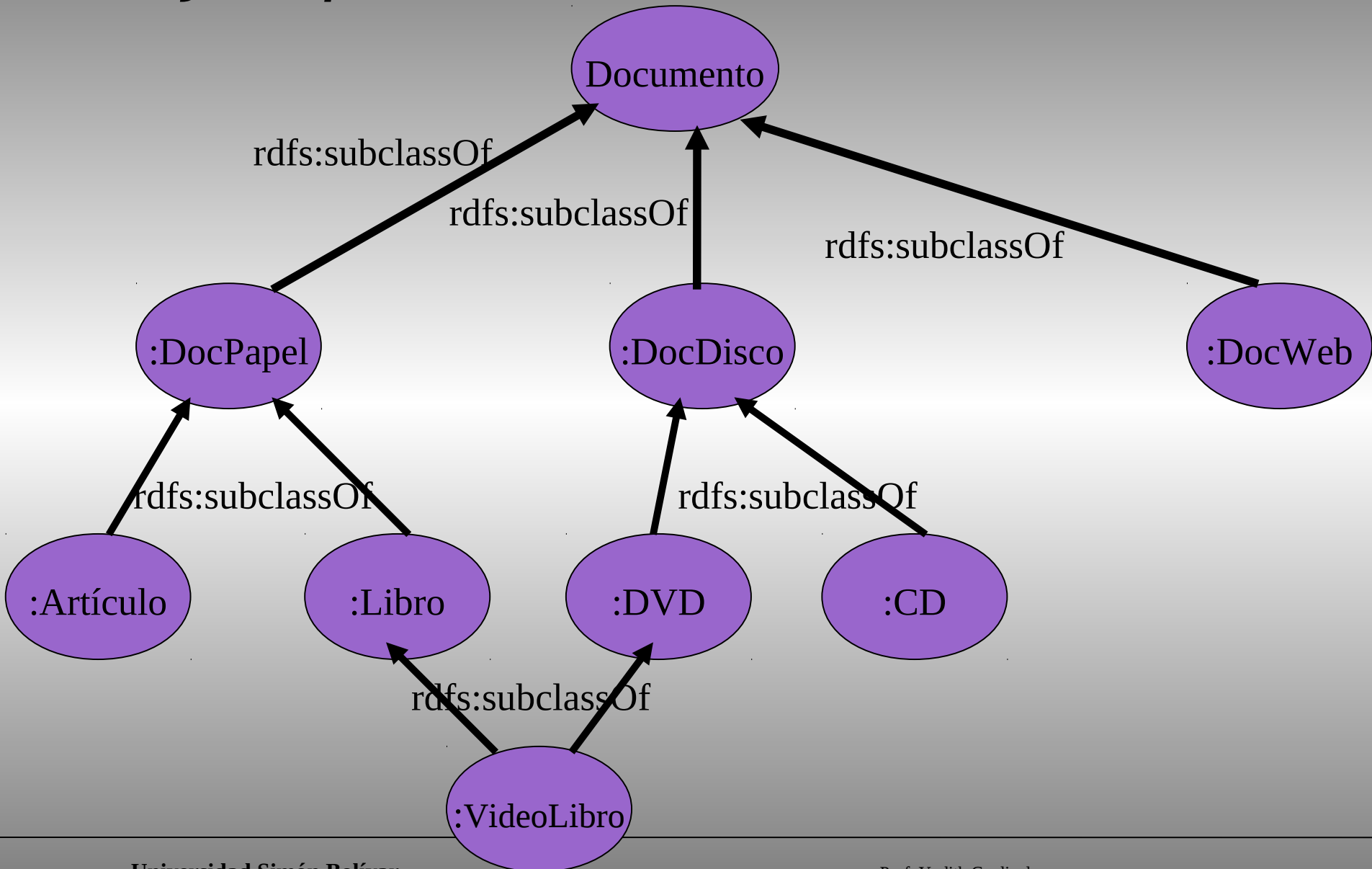
- ◆ Usualmente introduce
 - ◆ Conceptos: clases, entidades
 - ◆ Propiedades de los conceptos: atributos, slots, roles.
 - ◆ Relaciones entre conceptos.
 - ◆ Restricciones de integridad
 - ◆ Axiomas
 - ◆ Instancias □
- ◆ pueden ser
 - ◆ simples: sólo conceptos.
 - ◆ frame-based: sólo conceptos y propiedades.
 - ◆ logic-based: conceptos, propiedades y restricciones
 - ◆ deep: conjunto complejo de propiedades (ciencia e ingeniería)
 - ◆ Shallow: términos que no cambian y organizan grandes cantidades de datos (banco, finanzas) □
- ◆ expresados a través de diagramas.

Taxonomías

◆ Vocabulario para construir taxonomías

- ◆ Taxonomía: Clasificación de recursos en forma de jerarquía de clases.
- ◆ Ontología (en RDFS) = Taxonomía +
 - ◆ propiedades de las clases +
 - ◆ a qué clases puede aplicarse cada propiedad (dominio)
+
 - ◆ qué valores puede tener cada propiedad (rango)
- ◆ Con `rdfs:subClassOf` se pueden definir jerarquías de clases
`:LibroTecnico rdfs:subClassOf :Libro .`
- ◆ Una clase puede ser subclase de varias clases
`:LibroTecnico rdfs:subClassOf :DocsTecnicos .`

Ejemplo de Taxonomía



OWL (*Web Ontology Language*)

- ◆ Lenguaje de Ontología:
 - ◆ Vocabulario
 - Relaciones entre los entes en el vocabulario
 - ◆ Taxonomía
 - ◆ Conjunto de reglas que explican el significado del vocabulario y que permiten inferir nuevos hechos.
- ◆ Grupo W3 y DARPA promueven su definición.
- ◆ La idea central de OWL es permitir representaciones eficientes de ontologías que puedan someterse a procedimientos de decisiones
- ◆ Chequea si la ontología es lógicamente consistente o determina si un concepto particular cae dentro de la ontología

OWL (*Web Ontology Language*)

- ◆ OWL usa el “enlazado” (“linking”) provisto por RDF para permitir que las ontologías estén distribuidas
- ◆ Permite que las ontologías hagan referencias a términos que están en otras ontologías
- ◆ Soporta diferentes tipos de inferencias (clasificación y inclusión)
- ◆ Existe un gran número de razonadores disponibles.

OWL-Elementos

- ◆ Clases e instancias de clases
 - ◆ Clases (T-Box Ontología)
 - ◆ Instancias (A-Box Knowledge Base)
- ◆ Jerarquía de clases/instancias.
- ◆ Propiedades de Igualdad/Desigualdad
 - ◆ sameClassAs
 - ◆ samePropertyAs
 - ◆ sameIndividualAs
 - ◆ differentIndividualAs
- ◆ Propiedades
 - ◆ inverseOf
 - ◆ Transitive
 - ◆ Symmetric

OWL-Elementos

- ◆ Definición de una ontología:

```
<owl:Ontology rdf:about="">  
</owl: Ontology >
```
- ◆ El valor del atributo rdf:about es usualmente nulo para indicar que el sujeto de la afirmación es this.
- ◆ Definición de una clase

```
<owl:Class rdf:ID="Animal">  
  <rdfs:label> Animal </rdfs:label>  
</owl:Class>  
<owl:Class rdf:ID="Male">  
  <rdfs:subClass rdf:resource="#Animal">  
</owl:Class>
```

OWL-Elementos

```
<owl:Class rdf:ID="Man">  
  <rdfs:subClass rdf:resource="#Person">  
  <rdfs:subClass rdf:resource="#Male">  
</owl:Class>
```

```
<owl:Class rdf:ID="Wine">  
  <rdfs:subClass rdf:resource="&food;PotableLiquid"/>  
  <rdfs:label xml:lang="en">wine</rdfs:label>  
  <rdfs:label xml:lang="fr">vin</rdfs:label>  
</owl:Class>
```

```
<owl:Class rdf:ID="Pasta">  
  <rdfs:subClassOf rdf:resource="#EdibleThing"/>  
</owl:Class>
```

OWL-Ventajas

- ◆ **Poder expresivo equivalente a Description Logic.**
 - ◆ Semántica bien definida
 - ◆ Propiedades son conocidas
 - ◆ Existen algoritmos de inferencia
 - ◆ Evidencias de tractabilidad empírica. (Intratable)
- ◆ **Restricciones de cardinalidad**
- ◆ **Definición de Clases**
 - ◆ Extensión
 - ◆ Comprensión
- ◆ **Poder expresivo determinado por**
 - ◆ Clase de operadores que se seleccionen
 - ◆ Clase de axiomas que se permitan
- ◆ **Existen cientos de ontologías definidas en diversos dominios del saber.**

OWL-Desventajas

- ◆ Debe existir un número limitado de constructores/axiomas para que el proceso de inferencia sea decidible.
- ◆ No es muy natural.
- ◆ No permite definir/chequear restricciones de integridad.
- ◆ Open World Assumption
 - ◆ Razonamiento es más complejo
- ◆ Complejidad puede ser Intractable

OWL y The Semantic Web

- ◆ Datos puedan intercambiarse.
 - ◆ Si
- ◆ Semántica de los datos sea explícita
 - ◆ Si
- ◆ Propiedades de los datos y de los metadatos pueden ser inferidas.
 - ◆ Parcialmente

Composición automática de servicios web

- ◆ Considere un científico que necesita combinar un conjunto de servicios web para satisfacer un requerimiento.
- ◆ Existen muchos servicios web disponibles.
- ◆ Un servicio puede resolver una o más tareas simultáneamente.
- ◆ Servicios web pueden tener un costo asociado.
- ◆ El científico debe resolver dos tareas complejas:
 - Identificar un conjunto de servicios que satisfaga sus Requerimientos.
 - Seleccionar los servicios de manera que el costo sea mínimo.