



3ème année

Spécialité Informatique

Web sémantique et accès assisté à l'information

UNE INTRODUCTION AU WEB SÉMANTIQUE ET AUX ONTOLOGIES



Christine Porquet

Année 2010-2011

Une introduction au Web sémantique et aux ontologies

Mai 2001 – article de *Tim Berners-Lee, James Hendler, Ora Lassila -The Semantic Web, Scientific American*

« *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**.* »



« The Semantic Web is about two things. It is about common formats for integration and combination of data drawn from diverse sources, where on the original Web mainly concentrated on the interchange of documents. It is also about language for recording how the data relates to real world objects. That allows a person, or a machine, to start off in one database, and then move through an unending set of databases which are connected not by wires but by being about the same thing. »

<http://www.w3.org/2001/sw/>

<http://linkeddata.org>

13,1 milliards de triplets RDF – 142 millions de liens (novembre 2009)



- Christine PORQUET - ENSICAEN

- 3 -

Exemple 1:

Organiser un voyage à Budapest

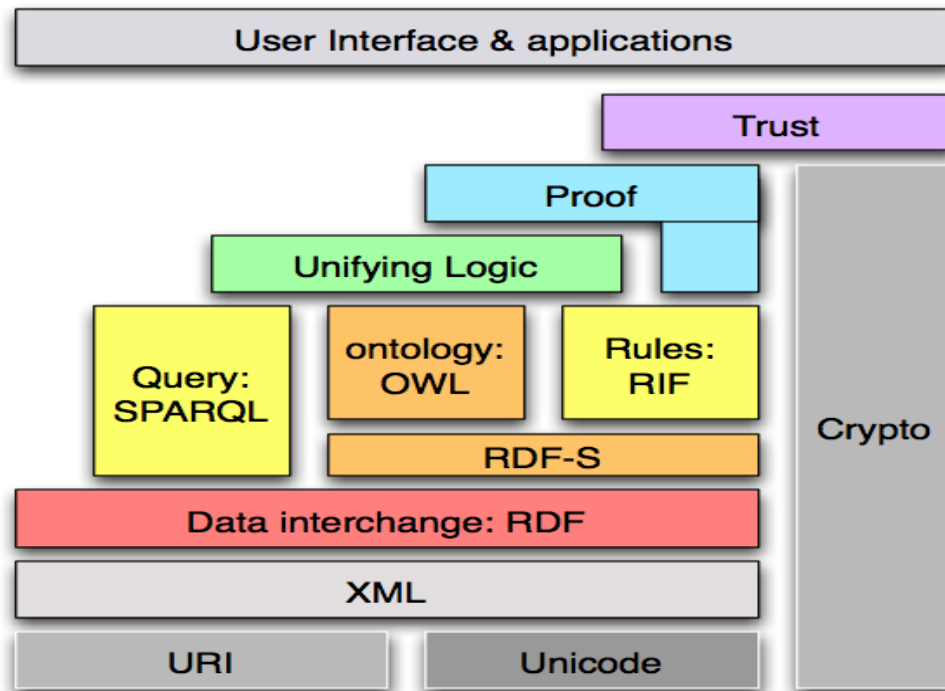


- Christine PORQUET - ENSICAEN

- 4 -

L'« arche » du Web Sémantique

(Tim Berners-Lee AAAI 07/2006)



- Christine PORQUET - ENSICAEN

- 5 -

Au sommaire



RDF : resource description framework
format d'échange de données



RDF Schema & OWL : **ontologies**



SPARQL : Simple Protocol and RDF Query Language



GRDDL : Gleaning Resource Descriptions from
Dialects of Languages



SKOS : Simple Knowledge Organization System

<http://www.w3.org/2001/sw/Specs.html>

- Christine PORQUET - ENSICAEN

- 6 -

Déroulement général de la normalisation d'une technologie Web par le W3C

1. Premiers frémissements : **Soumission de membre** sur un thème qui semble prometteur et/ou **veille stratégique** concernant des travaux à l'intérieur et à l'extérieur du W3C.
2. Après réussite d'un atelier et/ou débat sur une liste de diffusion du Comité consultatif, le Directeur annonce le développement d'une **proposition d'activité** nouvelle ou d'une **charte de groupe de travail**.
 - Une proposition d'activité décrit le cadre, la durée et les autres caractéristiques des travaux prévus et inclut les chartes d'un ou de plusieurs groupes de travail, groupes d'intérêts et éventuellement groupes de coordination pour mener à bien ces travaux.
 - Une charte de groupe de travail établit les prévisions pour chaque produit livrable du groupe (par exemple, les **rapports techniques**, les **ensembles de tests** et les **tutoriaux**).
3. Les groupes de travail produisent des **spécifications** et des **directives** qui passent par des cycles de vérification et de révision. Le processus du W3C pour réaliser ces rapports techniques comprend une vérification approfondie par les membres et le public, et l'obligation pour le groupe de travail de présenter une **expérience de mise en œuvre et d'interopérabilité**.
4. En fin de processus, le Comité consultatif examine le rapport technique à maturité et, s'il est entériné, le W3C le publie avec le statut de **recommandation**, c'est-à-dire l'équivalent W3C d'une norme du Web.

<http://www.w3.org/2005/10/Process-20051014/>

- Christine PORQUET - ENSICAEN

- 7 -

Sources principales utilisées dans ce cours

- Activités du W3C liées au développement du web sémantique. On trouve notamment à cette adresse toutes les recommandations du W3C concernant RDF, RDFS et OWL. <http://www.w3.org/2001/sw/>
- Diaporamas des chapitres 3 (sur RDF et RDFS) et 4 (sur OWL) du livre *A Semantic Web Primer* - Grigoris Antoniou & Frank van Harmelen
<http://www.ics.forth.gr/isl/swprimer/presentations/Chapter3.ppt>
<http://www.ics.forth.gr/isl/swprimer/presentations/Chapter4.ppt>
- Editeur d'ontologie Protégé – documentation, tutoriaux, etc.
<http://protege.stanford.edu/doc/users.html>
http://protegewiki.stanford.edu/index.php/Main_Page
- Communauté de développement autour de OWL (University of Manchester):
<http://owl.cs.manchester.ac.uk/>

- Christine PORQUET - ENSICAEN

- 8 -

Un peu de grec... un peu de philo...

- Paléontologie
paleo = ancien, ontos = être, logos = discours
- Ontologie
 - Étude des propriétés générales de ce qui existe
 - Branche de la philosophie qui s'intéresse à la qualité d'être, à la notion d'existence et ses catégories fondamentales
 - Exemples de questions ontologiques :
 - Qu'est qui fait qu'une entité existe ?
 - Qu'est qui en fait un objet physique ?

...et un peu d'informatique !

- En informatique, une ontologie est le résultat d'une formulation exhaustive et rigoureuse d'une conceptualisation:
 - organisation hiérarchique de concepts pertinents,
 - relations qui existent entre ces concepts,
 - règles et axiomes qui les contraignent.

Quels sont les besoins ?

- Une *identification* et un *accès* aux ressources du Web de façon simple, robuste, et efficace : **URI, Universal Resource Identifier**.
- Des langages pour décrire le *contenu* des documents/ressources et des moteurs d'inférences pour les manipuler...
- Des *ressources* de plusieurs types : ontologies (bases de connaissances), bases de données, éléments de code, etc.

Définir une sémantique

- 1) Préciser le domaine dans lequel est interprété le vocabulaire utilisé.
- 2) Donner les règles d'interprétation des mots de ce vocabulaire.
- 3) Définir la notion de modèles d'un ensemble d'expressions.

Présentation d'Ivan Herman

(semantic Web Activity Lead)



- Christine PORQUET - ENSICAEN

- 13 -

Exemple 2 : Dublin Core Metadata Initiative

<http://dublincore.org/>

- Le *Dublin Core* est un ensemble de 15 éléments de métadonnées ayant trait :
 - au **Contenu** : Title, Description, Subject, Source, Coverage, Type, Relation
 - à la **Propriété intellectuelle** : Creator, Contributor, Publisher, Rights
 - à la **Version** : Date, Format, Identifier, Language

- Christine PORQUET - ENSICAEN

- 14 -

Nom de l'élément	Identifiant	Définition
titre	Title	Le nom donné à la ressource
créateur	Creator	L'entité principalement responsable de la création du contenu de la ressource
sujet et mots-clefs	Subject	Le sujet du contenu de la ressource
description	Description	Une description du contenu de la ressource
éditeur	Publisher	L'entité responsable de la diffusion de la ressource, dans sa forme actuelle, tels un département universitaire, une entreprise.
contributeur	Contributor	Une entité qui a contribué à la création du contenu de la ressource
date	Date	Une date associée avec un événement dans le cycle de vie de la ressource
type	Type	La nature ou le genre du contenu de la ressource
format	Format	La matérialisation physique ou digitale de la ressource
identifiant	Identifier	Une référence non ambiguë à la ressource dans un contexte donné
source	Source	Une référence à une ressource à partir de laquelle la ressource actuelle a été dérivée
langue	Language	La langue du contenu intellectuel de la ressource
relation	Relation	Une référence à une autre ressource qui a un rapport avec cette ressource
couverture	Coverage	La portée ou la couverture spatio-temporelle de la ressource
droits	Rights	Information sur les droits sur et au sujet de la ressource

<rdf:RDF

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
xmlns:dc="http://purl.org/metadata/dublin_core#">
```

```
<rdf:Description about="http://www.dlib.org"
  dc>Title="D-Lib Program - Research in Digital
           Libraries"
  dc:Description="The D-Lib program supports the
                 community of people with research interests in
                 digital libraries and electronic publishing."
  dc:Publisher="Corporation For National
               Research Initiatives"
  dc>Date="1995-01-07"/>
```

```
</rdf:RDF>
```

RDF :

Ressource Description Framework

- Description de métadonnées
- Esperanto du web sémantique
- Modèle de triplets

`Resource property value`

`Objet attribut valeur`

- Syntaxe XML

RDF : vocabulaire réservé

- `rdf:RDF`, `rdf:Description`,
`rdf:type`
- `rdf:Seq`, `rdf:Bag`, `rdf:Alt`,
`rdf:List`
- *`rdf:Statement`, `rdf:subject`,
`rdf:predicate`, `rdf:object`*

RDF Schéma (RDFS)

- Définition de classes et de propriétés
- Spécialisation de classes et de propriétés
- Restriction du domaine et co-domaine des propriétés.

+

- Méta-modèle **réflexif** où tout est ressource
- Remarque: RDF Schéma : nom mal choisi.
RDF Schéma n'est pas à RDF ce que XML Schéma est à XML.
XML Schéma contraint la structure des documents XML, tandis que
RDF schéma définit le vocabulaire utilisé dans les modèles de
données RDF.

RDFS : vocabulaire réservé

- **rdfs:Resource,**
rdfs:Class, rdfs:subClassOf,
rdfs:subPropertyOf
- **rdfs:domain, rdfs:range**
- **rdfs:comment, rdfs:seeAlso,**
rdfs:isDefinedBy, rdfs:label

RDF et RDF Schema

(« *A semantic Web primer* »
chap. 3)



Traduction française:

<http://www.yoyodesign.org/doc/w3c/rdf-primer/>

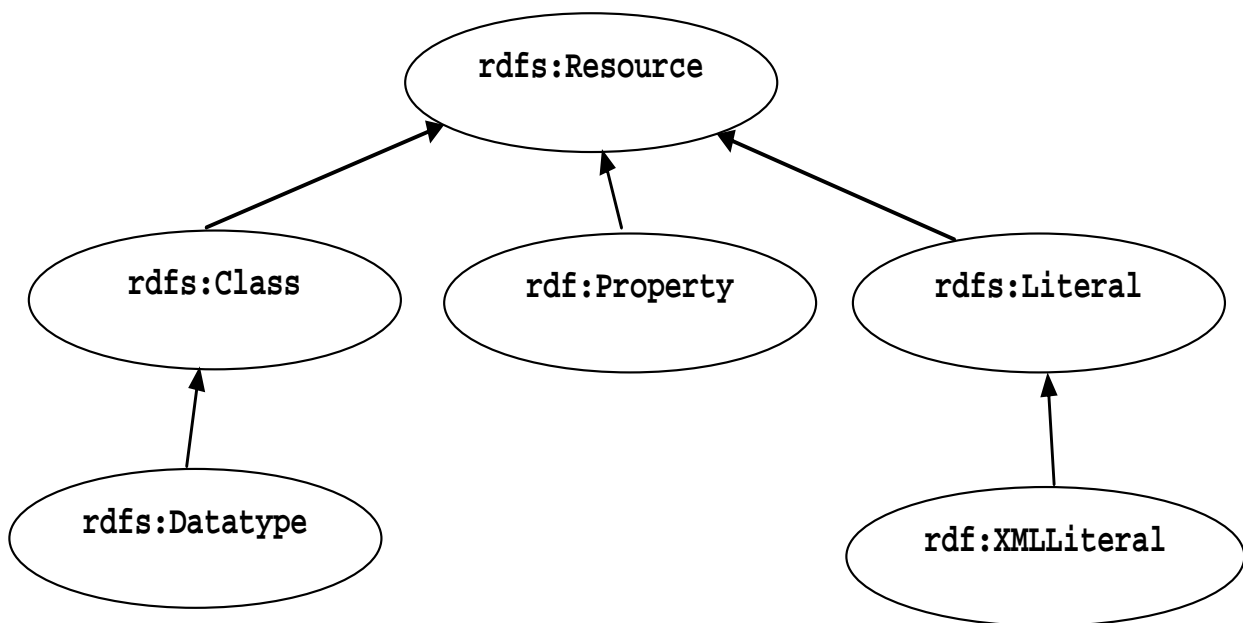
Exemple 3 :

Université – version RDFS



RDF Schéma : un méta-modèle réflexif (1/4)

Graphe d'héritage des classes (relation **rdfs:subClassOf**)

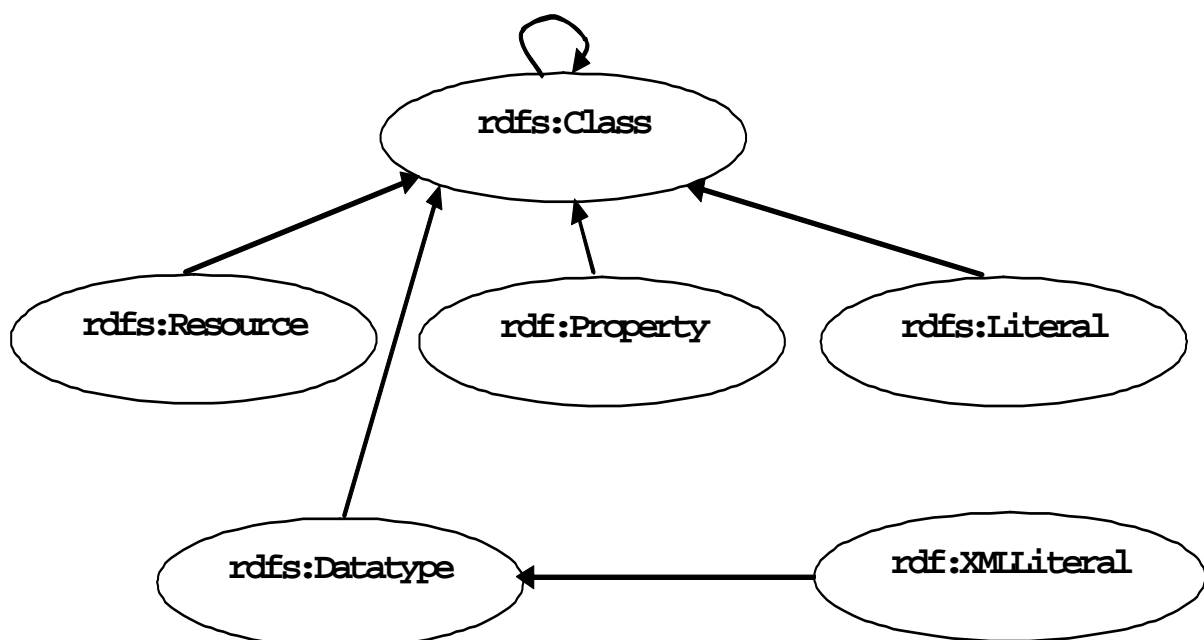


- Christine PORQUET - ENSICAEN

- 23 -

RDF Schéma : un méta-modèle réflexif (2/4)

Graphe d'instanciation des classes

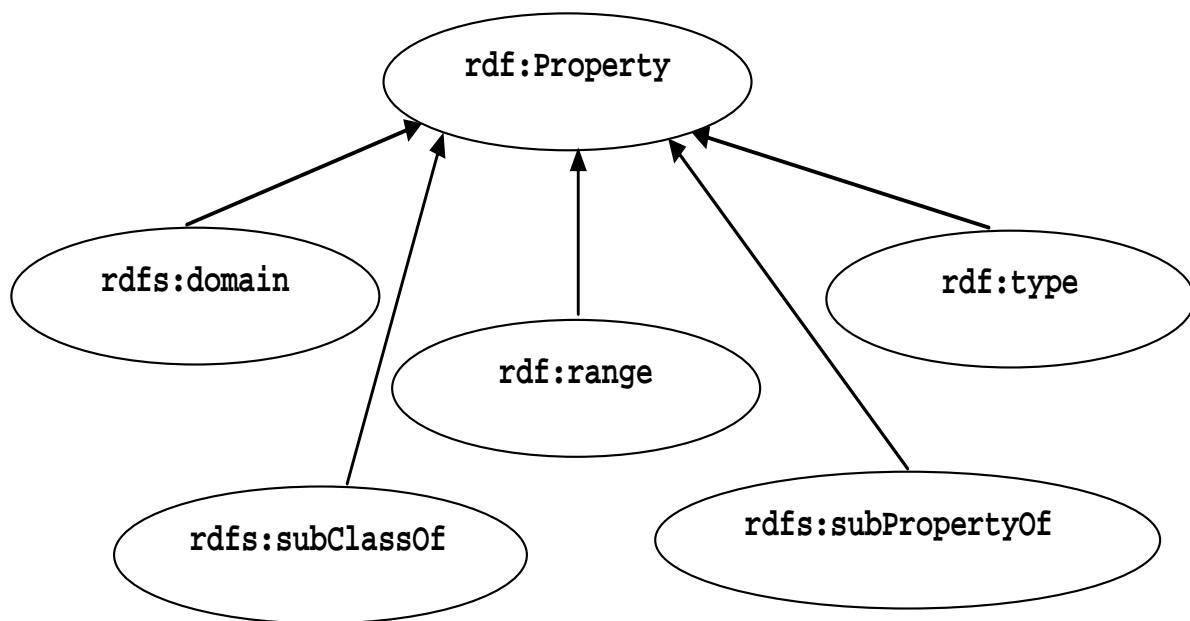


- Christine PORQUET - ENSICAEN

- 24 -

RDF Schéma : un méta-modèle réflexif (3/4)

Graphe d'instantiation des propriétés



- Christine PORQUET - ENSICAEN

- 25 -

RDF Schéma : un méta-modèle réflexif (4/4)



- Christine PORQUET - ENSICAEN

- 26 -

Exemples d'inférences en RDF/RDFS (1/3)

Héritage de type à l'aide de `rdfs:subClassOf`

A partir de :

```
(rdf:type gros_minet chat)
```

```
(rdfs:subClassOf chat mammifère)
```

On peut déduire que :

```
(rdf:type gros_minet mammifère)
```

Exemples d'inférences en RDF/RDFS (2/3)

Inférence de type à partir des contraintes `rdfs:range` et `rdfs:domain`

A partir de :

```
(rdfs:domain est_enseignant_de Professeur)
```

```
(rdfs:range est_enseignant_de Etudiant)
```

```
(est_enseignant_de Chris Alex)
```

On peut déduire que :

```
(rdf:type Chris Professeur)
```

```
(rdf:type Alex Etudiant)
```

Exemples d'inférences en RDF/RDFS (3/3)

Transitivité de `rdfs:subClassOf` et `rdfs:subPropertyOf`

A partir de :

```
(rdfs:subClassOf Chien Mammifère)  
(rdfs:subClassOf Mammifère Animal)
```

on peut déduire que :

```
(rdfs:subClassOf Chien Animal)
```

A partir de :

```
(rdfs:subPropertyOf parent ancêtre)  
(rdfs:subPropertyOf ancêtre membre_de_la_famille)
```

on peut déduire que :

```
(rdfs:subPropertyOf parent membre_de_la_famille)
```

The Web Ontology Language OWL

Basé sur :

- DAML-ONT: un langage de « frames » sur RDF;
- OIL: une logique de description sur XML.

OIL + DAML-ONT = DAML+OIL
W3C + DAML+ OIL = OWL

Une couche au-dessus de
RDF + RDFS + XML-Schema

Propose des constructeurs de logique de
description dans une syntaxe d'objets.

OWL : vocabulaire réservé

- owl:Class, rdfs:subClassOf, rdfs:subPropertyOf
- owl:DatatypeProperty, owl:ObjectProperty
- owl:TransitiveProperty, owl:FunctionalProperty
- owl:complementOf, owl:unionOf, owl:intersectionOf
- owl:oneOf
- owl:Restriction, owl:allValuesFrom, owl:someValueFrom, owl:hasValue, owl:value, owl:maxCardinality, owl:minCardinality
- owl:disjointWith, owl:equivalentProperty, owl:sameAs, owl:differentFrom

- Christine PORQUET - ENSICAEN

- 31 -

OWL: syntaxe

```
<Class rdf:ID="Human">
  <rdfs:subClassOf="owl:Thing"/>
  <rdfs:subClassOf>
    <Restriction>
      <onProperty rdf:resource="#hasParents"/>
      <allValuesFrom rdf:resource="#Human"/>
    </Restriction>
  </rdfs:subClassOf>
</Class>

<Class rdf:ID="Man">
  <intersectionOf>
    <Class about="#Human"/>
    <Class about="#Male"/>
  </intersectionOf>
</Class>
```

- Christine PORQUET - ENSICAEN

- 32 -

OWL

- Pour compléter les possibilités d'inférences de RDF/RDFS
- Pour raffiner les descriptions
- S'appuie sur une logique de description basée sur RDF/RDFS avec une syntaxe RDF/XML
- 3 versions :
 - OWL Lite
 - OWL DL DL = Description Logics
 - OWL Full Indécidable

Logique de description (1/4)

- Modèle objet pour la classification de concepts
- Concept = ensemble d'individus
- Rôle = relation binaire entre individus
- Concepts et rôles : niveau terminologique (Tbox)
- Individus : niveau des assertions (Abox)

Logique de description (2/4)

- Relation de **subsumption** : organiser les concepts par niveau de généralité
- Un concept A subsume B si l'ensemble des individus de B est inclus dans l'ensemble des individus de A :
A est plus général que B
- Classification: déterminer la position d'un concept dans une hiérarchie de subsumption

Logique de description (3/4)

- Les concepts sont définis par des expressions mettant en jeu des concepts et des rôles.
- Une femme est une personne du sexe féminin
woman : person \cap female
- Une mère est une femme qui a des enfants
mother : woman \cap \exists hasChild.person

Logique de description (4/4)



- Pour en savoir plus:

<http://dl.kr.org/>

- Sémantique formelle et complexité:

<http://owl.cs.manchester.ac.uk/navigator/>

Classes

- La superclasse de tout : `owl:Thing`
- La classe vide (sans instances) : `owl:Nothing`

$\text{Thing} = \text{Nothing} \cup \neg \text{Nothing}$

$\text{Nothing} = \neg \text{Thing}$

- La métaclasse `owl:Class`

exemple: `human rdf:type owl:Class`

- remarque:

`owl:Class rdfs:subClassOf rdfs:Class`

allValuesFrom

- Un humain a des parents humains
 $\text{human}(x) \Rightarrow (\text{parent}(x,y) \Rightarrow \text{human}(y))$
`human : all parent human`
- `owl:Restriction` : définit une classe anonyme

```
owl:Class human
  rdfs:subClassOf
    owl:Restriction
      owl:onProperty parent
      owl:allValuesFrom human
```

equivalentClass

`human \Leftrightarrow all parent human`

```
owl:Class human
  owl:equivalentClass
    owl:Restriction
      owl:onProperty parent
      owl:allValuesFrom human
```

someValuesFrom

human \Leftrightarrow some parent woman

```
owl:Class human
  rdfs:subClass of
    owl:Restriction
      owl:onProperty parent
      owl:someValuesFrom woman
```

cardinality

- nombre de valeurs sémantiquement distinctes d'une propriété
- Les humains ont exactement deux parents

```
owl:Class human
  rdfs:subClass of
    owl:Restriction
      owl:onProperty parent
      owl:cardinality 2
```

Cardinalité

- `owl:cardinality 2`
- `owl:maxcardinality 2`
- `owl:mincardinality 2`
- Remarque: en OWL Lite :
uniquement 0 ou 1

Sous-classes & héritage multiple

`woman \Rightarrow human \wedge female`

`owl:Class woman`

`rdfs:subClassOf human`

`rdfs:subClassOf female`

Intersection

`human` \wedge `female` \Leftrightarrow `woman`

Définit une équivalence entre
woman et l'intersection de human et female

`owl:Class woman`

`owl:intersectionOf`

`owl:Class human`

`owl:Class female`

Propriétés

- **`owl:ObjectProperty`**
propriété dont la valeur est un individu (une instance)
- **`owl:DatatypeProperty`**
propriété dont la valeur est une valeur littérale (integer, string, float, boolean, date)
- **`owl:AnnotationProperty`**
`rdfs:comment` `rdfs:label`
ne participent pas aux inférences, purement documentaire

equivalentProperty

- `P1 owl:equivalentProperty P2`
- `X P1 Y ⇔ X P2 Y`
- Utile quand on importe une ontologie
- `hasPart owl:equivalentProperty ns:sous-partie`

inverseOf

- `P1 owl:inverseOf P2`
- `X P1 Y ⇔ Y P2 X`
- `hasParent owl:inverseOf hasChild`
- `John hasParent Jim ⇔ Jim hasChild John`

SymmetricProperty

- `P rdf:type owl:SymmetricProperty`
- $X P Y \Leftrightarrow Y P X$
- Exemple : `sibling` (frère ou sœur)

`John sibling Jack \Leftrightarrow Jack sibling John`

TransitiveProperty

- $X P Y \wedge Y P Z \Rightarrow X P Z$
- Exemple :
`partOf rdf:type`
`owl:TransitiveProperty`

`axis partOf engine`
 `\wedge engine partOf car`
 `\Rightarrow axis partOf car`

FunctionalProperty

- husband rdf:type owl:FunctionalProperty
- Une valeur **unique** pour une ressource donnée
- $X \text{ husband } Y \wedge X \text{ husband } Z \Rightarrow Y = Z$

InverseFunctionalProperty

- mother rdf:type owl:InverseFunctionalProperty
- Une ressource **unique** pour une valeur donnée
- $X \text{ mother } Z \wedge Y \text{ mother } Z \Rightarrow X = Y$

Individus identiques ou différents

- OWL ne fait **aucune présomption de nom unique**. Le simple fait que deux noms soient différents ne signifie pas qu'ils se rapportent à des individus différents.

- Individus identiques: **sameAs**

Emile_Ajar owl:sameAs Romain_Gary

- Individus différents: **differentFrom**

Jarosław_Kaczyński owl:differentFrom Lech_Kaczyński

Individus mutuellement distincts: **AllDifferent**

```
<owl:AllDifferent>
  <owl:distinctMembers
    rdf:parseType="Collection">
    <vin:WineColor rdf:about="#Red"/>
    <vin:WineColor rdf:about="#White"/>
    <vin:WineColor rdf:about="#Rose"/>
  </owl:distinctMembers>
</owl:AllDifferent>
```

Enumération: oneOf

Uniquement OWL DL & OWL Full

```
<owl:Class rdf:ID="WineColor">
  <rdfs:subClassOf
    rdf:resource="#WineDescriptor"/>
  <owl:oneOf
    rdf:parseType="Collection">
    <owl:Thing rdf:about="#White"/>
    <owl:Thing rdf:about="#Rose"/>
    <owl:Thing rdf:about="#Red"/>
  </owl:oneOf>
</owl:Class>
```

Restriction de valeur: hasValue

Uniquement OWL DL & OWL Full

```
owl:Class human
  rdfs:subClassOf
    owl:Restriction
      owl:onProperty numOfLegs
      owl:hasValue 2
```

unionOf

Uniquement OWL DL & OWL Full

```
owl:Class humanoid
```

```
  owl:unionOf
```

```
    chimpanzee
```

```
    gorilla
```

```
    man
```

complementOf

Uniquement OWL DL & OWL Full

```
owl:Class invertebrate
```

```
  owl:complementOf
```

```
    vertebrate
```

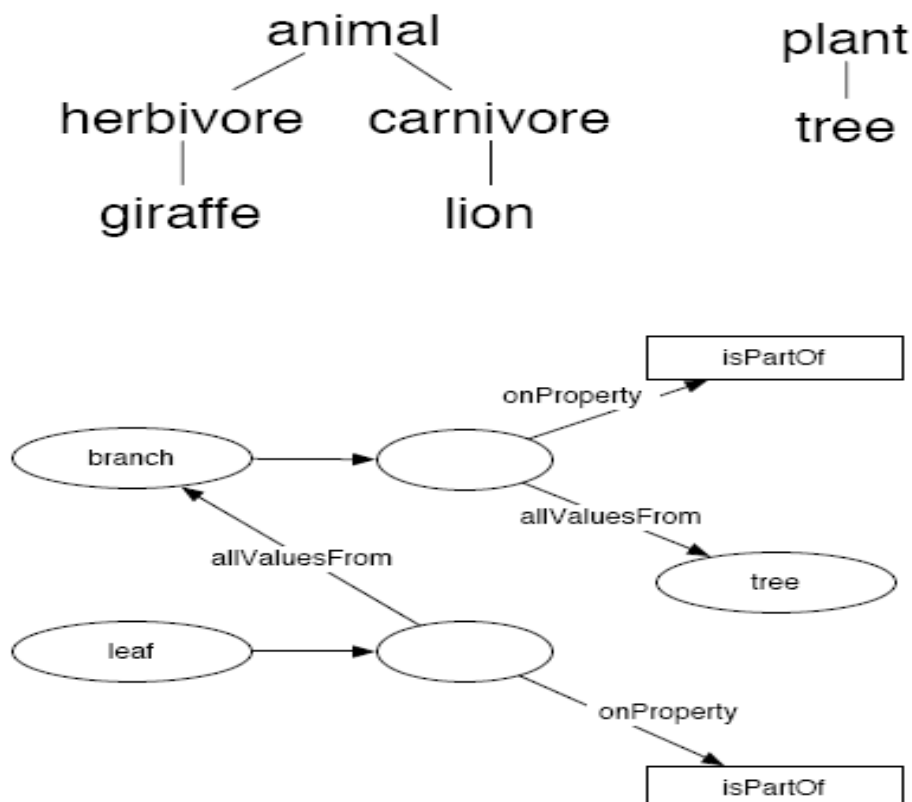
disjointWith

Uniquement OWL DL & OWL Full

- Garantit que l'individu membre d'une classe ne peut pas être simultanément l'instance d'une autre classe.

```
<owl:Class rdf:ID="Pasta">
  <rdfs:subClassOf
    rdf:resource="#EdibleThing"/>
  <owl:disjointWith rdf:resource="#Meat"/>
  <owl:disjointWith rdf:resource="#Fowl"/>
  <owl:disjointWith rdf:resource="#Seafood"/>
  <owl:disjointWith rdf:resource="#Dessert"/>
  <owl:disjointWith rdf:resource="#Fruit"/>
</owl:Class>
```

African Wildlife Ontology



African wildlife ontology



Publication par l'**INSEE** de données géographiques en RDF (<http://rdf.insee.fr/geo/>)

- Modélisation du code officiel géographique (millésime 2003) sous forme d'une **ontologie** définie en OWL-Lite.
- Fichiers contenant les instances des classes définies dans cette ontologie (au format RDF):
 - un fichier "**Régions**" contenant la liste des régions ainsi que leur chef-lieu et les départements inclus dans ces régions;
 - un fichier "**Départements**" contenant la liste des départements, leurs chefs-lieux ainsi que les arrondissements inclus dans ces départements;
 - pour chaque département (ex: le Calvados):
 - un fichier "**Arrondissements**" incluant la liste des communes et cantons par arrondissement du département;
 - un fichier "**Cantons**" qui modélise les liens entre communes et cantons ainsi que les relations de voisinage entre communes (y compris les communes limitrophes du département).

L'ontologie

```
<!-- Ontology Information -->
<owl:Ontology rdf:about="http://rdf.insee.fr/geo/ontologie-geo-2006.rdf">
  <rdfs:label xml:lang="fr">Ontologie du Code Officiel Géographique
  INSEE</rdfs:label>
  <owl:versionInfo xml:lang="fr">Version 1.0 - 2006-07-18</owl:versionInfo>
</owl:Ontology>

<!-- Classes -->
<owl:Class rdf:about="Region"
  rdfs:label="Région">
  <rdfs:subClassOf rdf:resource="NUTS_2"/>
  <rdfs:subClassOf rdf:resource="Territoire_FR"/>
</owl:Class>

<owl:Class rdf:about="Departement"
  rdfs:label="Département">
  <rdfs:subClassOf rdf:resource="NUTS_3"/>
  <rdfs:subClassOf rdf:resource="Territoire_FR"/>
</owl:Class>

<owl:Class rdf:about="Arrondissement">
  <rdfs:label xml:lang="fr">Arrondissement</rdfs:label>
  <rdfs:subClassOf rdf:resource="Territoire_FR"/>
</owl:Class>

<owl:Class rdf:about="Canton">
  <rdfs:subClassOf rdf:resource="Territoire_FR"/>
</owl:Class>
```

- Christine PORQUET - ENSICAEN

- 63 -

Le fichier des régions

```
<geo:Pays rdf:about="PAYS_FR">
  <geo:code_ISO>FR</geo:code_ISO>
  <geo:nom xml:lang="fr">France</geo:nom>
  <geo:subdivision>
    <geo:Region rdf:about="REG_25">
      <geo:code_region>25</geo:code_region>
      <geo:nom xml:lang="fr">Basse-Normandie</geo:nom>
      <geo:chef-lieu>
        <geo:Commune rdf:about="COM_14118">
          <geo:code_commune>14118</geo:code_commune>
          <geo:nom xml:lang="fr">Caen</geo:nom>
        </geo:Commune>
      </geo:chef-lieu>
    </geo:Region>
    <geo:subdivision>
      <geo:Departement rdf:about="DEP_14">
        <geo:code_departement>14</geo:code_departement>
        <geo:nom xml:lang="fr">Calvados</geo:nom>
      </geo:Departement>
    </geo:subdivision>
    ...
  </geo:subdivision>
  ...
</geo:Pays>
```

- Christine PORQUET - ENSICAEN

- 64 -

Le fichier des départements

```
<geo:Departement rdf:about="DEP_14">
  <geo:code_departement>14</geo:code_departement>
  <geo:nom xml:lang="fr">Calvados</geo:nom>
  <geo:chef-lieu>
    <geo:Commune rdf:about="COM_14118">
      <geo:code_commune>14118</geo:code_commune>
      <geo:nom xml:lang="fr">Caen</geo:nom>
    </geo:Commune>
  </geo:chef-lieu>
  <geo:subdivision>
    <geo:Arrondissement rdf:about="ARR_141">
      <geo:code_arrondissement>141</geo:code_arrondissement>
      <geo:nom xml:lang="fr">Bayeux</geo:nom>
    </geo:Arrondissement>
  </geo:subdivision>
  <geo:subdivision>
    <geo:Arrondissement rdf:about="ARR_142">
      <geo:code_arrondissement>142</geo:code_arrondissement>
      <geo:nom xml:lang="fr">Caen</geo:nom>
    </geo:Arrondissement>
  </geo:subdivision>
  ...
</geo:Departement>
```

- Christine PORQUET - ENSICAEN

- 65 -

Le fichier des arrondissements du Calvados

```
<geo:Arrondissement rdf:about="ARR_141">
  <geo:code_arrondissement>141</geo:code_arrondissement>
  <geo:nom xml:lang="fr">Bayeux</geo:nom>
  <geo:chef-lieu rdf:resource="COM_14047"/>

  <geo:subdivision>
    <geo:Canton rdf:about="CAN_1402">
      <geo:code_canton>1402</geo:code_canton>
      <geo:nom xml:lang="fr">Balleroy</geo:nom>
    </geo:Canton>
  </geo:subdivision>
  ...
  <geo:subdivision>
    <geo:Commune rdf:about="COM_14035">
      <geo:code_commune>14035</geo:code_commune>
      <geo:nom xml:lang="fr">Balleroy</geo:nom>
    </geo:Commune>
  </geo:subdivision>
  ...
</geo:Arrondissement>
```

- Christine PORQUET - ENSICAEN

- 66 -

Le fichier des cantons du Calvados

```
<geo:Canton rdf:about="CAN_1402">
  <geo:code_canton>1402</geo:code_canton>
  <geo:nom xml:lang="fr">Balleroy</geo:nom>
  <geo:chef-lieu rdf:resource="COM_14035"/>
  <geo:subdivision>
    <geo:Commune rdf:about="COM_14035">
      <geo:code_commune>14035</geo:code_commune>
      <geo:nom xml:lang="fr">Balleroy</geo:nom>
      <geo:voisin rdf:resource="COM_14050"/>
      <geo:voisin rdf:resource="COM_14140"/>
      ...
    </geo:Commune>
  </geo:subdivision>
</geo:subdivision>
  <geo:Commune rdf:about="COM_14140">
    <geo:code_commune>14140</geo:code_commune>
    <geo:nom xml:lang="fr">Castillon</geo:nom>
    <geo:voisin rdf:resource="COM_14468"/>
    <geo:voisin rdf:resource="COM_14035"/>
    ...
  </geo:Commune>
</geo:subdivision>
...
</geo:Canton>
```

- Christine PORQUET - ENSICAEN

- 67 -

Exemples d'inférences en OWL (1/5)

Transitivité de `owl:TransitiveProperty`

A partir de :

(rdf:type ancêtre owlTransitiveProperty)

(ancêtre Bernard Amaury)

(ancêtre Lucy Bernard)

On peut déduire que :

(ancêtre Lucy Amaury)

- Christine PORQUET - ENSICAEN

- 68 -

Exemples d'inférences en OWL (2/5)

Utilisation de `owl:inverseOf`

A partir de :

```
owl:inverseOf est_mère_de a_pour_mère)  
(a_pour_mère Amaury Chantal)
```

On peut déduire que :

```
(est_mère_de Chantal Amaury)
```

Exemples d'inférences en OWL (3/5)

Utilisation de `owl:sameAs`

A partir de :

```
(owl:sameAs Humain Personne)  
(rdf:type Humain rfs:Class)  
(rdf:type Personne rfs:Class)
```

On peut déduire que :

```
(owl:equivalentClass Humain Personne)
```

Exemples d'inférences en OWL (4/5)

Utilisation de `owl:FunctionalProperty`

A partir de :

```
(rdf:type a_pour_mère
  owl:FunctionalProperty)
(a_pour_mère Lise Isabelle)
(a_pour_mère Lise Isa)
```

On peut déduire que :

```
(owl:sameAs Isabelle Isa)
```

- Christine PORQUET - ENSICAEN

- 71 -

Exemples d'inférences en OWL (5/5)

Les vaches folles, ça n'existe pas ! (inconsistance)

Une vache est végétarienne.

Les plantes ou les parties de plantes sont disjointes des animaux ou parties d'animaux.

Les végétariens ne mangent que des choses qui ne se sont pas des animaux ou des parties d'animaux.

Un mouton est un animal qui ne mange que de l'herbe.

Une vache folle est une vache qui a mangé de la cervelle de mouton.

donc une vache folle a mangé une partie d'un animal ce qui est en contradiction avec la définition du concept de végétarien

```
Class (vache partial végétarien)
```

```
DisjointClasses(
  unionOf(animal restriction(partie_de someValuesFrom (animal)))
) unionOf(plante restriction(partie_de someValuesFrom (plante)))
```

```
Class (végétarien complete intersectionOf(animal
  restriction(mange allValuesFrom (complementOf
    (restriction(partie_de someValuesFrom (animal)))))
  restriction(mange allValuesFrom (complementOf(animal)))))
```

```
Class (mouton partial restriction (mange allValuesFrom herbe))
```

```
Class (vache_folle complete intersectionOf(vache
  restriction(mange someValuesFrom (intersectionOf(cerveau
    restriction(partie_de someValuesFrom (mouton)))))
```

- Christine PORQUET - ENSICAEN

- 72 -

OWL 2.0 – nouveautés

<http://www.w3.org/TR/owl2-primer/>

Class: **Person**

SubClassOf: hasAge exactly 1
and hasGender exactly 1
and hasGender only {female , male}

ObjectProperty: **hasWife**

Characteristics: Functional, InverseFunctional, Irreflexive,
Asymmetric

Domain: Person, Man

Range: Person, Woman

SubPropertyOf: hasSpouse, loves

Class: **Teenager**

EquivalentTo: Person that hasAge some integer[>= 13 , < 20]

Intervalles de valeurs: la classe Teenager

Manchester Syntax

Class: Teenager SubClassOf: hasAge **some integer[<= 13 , >= 19]**

RDF/XML Syntax

```
<owl:Class rdf:about="Teenager">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="hasAge"/>
      <owl:someValuesFrom>
        <rdfs:Datatype>
          <owl:onDatatype rdf:resource="&xsd;integer"/>
          <owl:withRestrictions rdf:parseType="Collection">
            <xsd:minExclusive rdf:datatype="&xsd;integer"> 12 </xsd:minExclusive>
            <xsd:maxInclusive rdf:datatype="&xsd;integer"> 19 </xsd:maxInclusive>
          </owl:withRestrictions>
        </rdfs:Datatype>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Propriétés disjointes:

les mariages entre parents et enfants sont illégaux

Manchester Syntax

DisjointProperties: hasParent hasSpouse

RDF/XML Syntax

```
<rdf:Description rdf:about="hasParent">  
  <owl:propertyDisjointWith rdf:resource="hasSpouse"/>  
</rdf:Description>
```

Syntaxe de Manchester

http://www.co-ode.org/resources/reference/manchester_syntax/

- hasChild some Man
- hasChild exactly 3
- Man or Woman
- Person and
hasChild some (Person and (hasChild only Man)
and (hasChild some Person))
describes the set of people who have at least one child that has some
children that are only men
(i.e. grandparents that only have grandsons).
- hasAge value "21"^^long
- Person and hasAge some int[>= 18, <= 30]

Editeurs d'ontologie

- Protégé

<http://protege.stanford.edu/>

- Owlsight

<http://pellet.owldl.com/ontology-browser/>

SPARQL Query Language for RDF

- Le langage SPARQL définit la syntaxe et la sémantique nécessaire à l'expression de requêtes sur une base de données de type **RDF** et la forme possible des résultats.
- SPARQL permet d'exprimer des requêtes interrogatives ou constructives :
 - Une requête **SELECT**, de type interrogative, permet d'extraire du graphe RDF un sous-graphe correspondant à un ensemble de ressources vérifiant les conditions définies dans une clause **WHERE**.
 - Une requête **CONSTRUCT**, de type constructive, engendre un nouveau graphe qui complète le graphe interrogé.

Ex: sur un graphe RDF contenant des informations **généalogiques**, on pourra par une requête **SELECT** trouver les parents ou grands-parents d'une personne donnée, et par des requêtes **CONSTRUCT** ajouter des relations frère-sœur, cousin-cousine, oncle-neveu, qui ne seraient pas explicitement déclarées dans le graphe initial.

Exemple : Données RDF utilisant l'ontologie FOAF

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rss="http://purl.org/rss/1.0/"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <foaf:Person rdf:about="http://example.net/Paul_Dupont">
    <foaf:name>Paul Dupont</foaf:name>
    <foaf:img rdf:resource="http://example.net/Paul_Dupont.jpg"/>
    <foaf:knows
      rdf:resource="http://example.net/Pierre_Dumoulin"/>
  </foaf:Person>
  <foaf:Person rdf:about="http://example.net/Pierre_Dumoulin">
    <foaf:name>Pierre Dumoulin</foaf:name>
    <foaf:img rdf:resource="http://example.net/Pierre_Dumoulin.jpg"/>
  </foaf:Person>
  <foaf:Image rdf:about="http://example.net/Paul_Dupont.jpg">
    <dc:description>Photo d'identité de Paul Dupont
  </dc:description>
  </foaf:Image>
  <foaf:Image rdf:about="http://example.net/Pierre_Dumoulin.jpg">
    <dc:description>Photo d'identité de Pierre Dumoulin
  </dc:description>
  </foaf:Image>
</rdf:RDF>
```

- Christine PORQUET - ENSICAEN

- 79 -

Exemple : Requête SPARQL

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
```

```
SELECT DISTINCT ?nom ?image ?description
WHERE {
    ?personne rdf:type foaf:Person .
    ?personne foaf:name ?nom .
    ?image rdf:type foaf:Image .
    ?personne foaf:img ?image .
    ?image dc:description ?description }
```

- On remarque la déclaration des **espaces de noms** en début, suivis de la requête proprement dite. Le nom des **variables** est précédé d'un point d'interrogation ?.
- La ligne SELECT permet de sélectionner l'ensemble des tuples, ou lignes de variables (*nom*, *image*, *description*) correspondant aux contraintes de la clause WHERE.
- La première ligne de la clause WHERE se lit : la variable *personne* est de type *Person* au sens de l'ontologie FOAF. La seconde ligne permet de définir la variable *nom* en tant que propriété *name* de la variable *personne*.

- Christine PORQUET - ENSICAEN

- 80 -

Le résultat SPARQL

```
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  <head>
    <variable name="nom"/>
    <variable name="image"/>
    <variable name="description"/> </head>
  <results ordered="false" distinct="true">
    <result>
      <binding name="nom"> <literal>Pierre Dumoulin</literal>
      </binding>
      <binding name="image">
        <uri>http://example.net/Pierre\_Dumoulin.jpg</uri> </binding>

      <binding name="description">
        <literal>Photo d'identité de Pierre Dumoulin</literal>
      </binding>
    </result>
    <result>
      <binding name="nom"> <literal>Paul Dupont</literal> </binding>
      <binding name="image">
        <uri>http://example.net/Paul\_Dupont.jpg</uri> </binding>
      <binding name="description">
        <literal>Photo d'identité de Paul Dupont</literal> </binding>
    </result>
  </results>
</sparql>
```

- Christine PORQUET - ENSICAEN

- 81 -

- Langage d'interrogation SPARQL pour RDF (traduction française)
<http://www.yoyodesign.org/doc/w3c/rdf-sparql-query/>
- SPARQL par l'exemple
<http://www.cambridgesemantics.com/2008/09/sparql-by-example>

- Christine PORQUET - ENSICAEN

- 82 -

GRDDL

Gleaning Resource Descriptions from Dialects of Languages

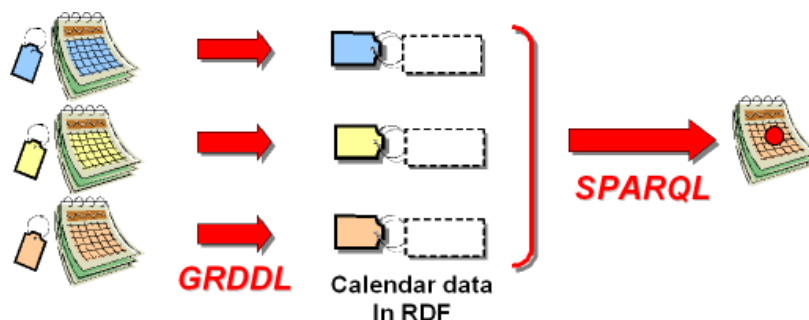
<http://www.w3.org/TR/grddl-primer/>

- Question: comment faire entrer les données disponibles aux formats XHTML et XML dans le monde de RDF et du Web sémantique ?
- Réponse: par un mécanisme **explicite** liant l'espace de nom à son algorithme de traitement (par ex. **XSLT**).

Use case #1 - Scheduling :

Jane is trying to coordinate a meeting.

- Robin publishes his schedule on his home page using the [hCalendar microformat](#).
- David publishes his in [Embedded RDF](#) using some RDF calendar properties.
- Kate uses a blog engine that encodes her diary as [RDFa](#).
- Jane uses an online calendaring service that publishes an [RSS 1.0](#) feed of her schedule.



L'agenda de Robin

1. 2006

1. **Fashion Expo** in Paris, France: Oct 20 to 22
2. **New line review** in Cologne, Germany: Oct 26 to 27
3. **Clothing 2006** in Rome, Italy: Dec 1 to 5

2. 2007

1. **Web Design Conference** in Edinburgh, UK: Jan 8 to 10
2. **Board Review** in New York, USA: Feb 23 to 24

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>
    <title>Robin's Schedule</title>
  </head>
  <body>
    <ol class="schedule">
      <li>2006
        <ol>
          <li class="vevent">
            <strong class="summary">Fashion Expo</strong> in
            <span class="location">Paris, France</span>:
            <abbr class="dtstart" title="2006-10-20">Oct 20</abbr> to
            <abbr class="dtend" title="2006-10-23">22</abbr>
          </li>

          <li class="vevent">
            <strong class="summary">New line review</strong> in
            <span class="location">Cologne, Germany</span>:
            <abbr class="dtstart" title="2006-10-26">Oct 26</abbr> to
            <abbr class="dtend" title="2006-10-28">27</abbr>
          </li>

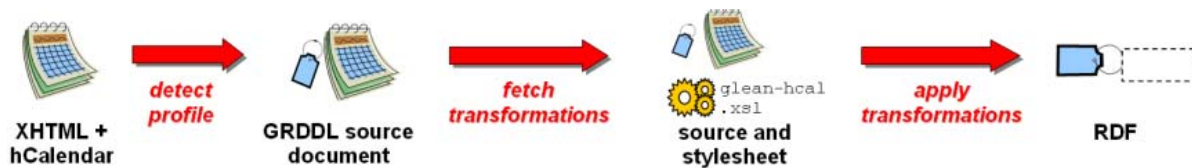
          ...
        </ol>
      </li>

      ...
    </ol>
  </body>
</html>
```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
    "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
    lang="en">
  <head profile="http://www.w3.org/2003/g/data-view">
    <title>Robin's Schedule</title>
    <link rel="transformation"
      href="http://www.w3.org/2002/12/cal/glean-hcal"/>
  </head>
  <body>
    ...

```



```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:_6="http://www.w3.org/2002/12/cal/icaltzd#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <rdf:Description rdf:nodeID="jTCXOrie4">
    <rdf:type rdf:resource="http://www.w3.org/2002/12/cal/icaltzd#Vcalendar"/>
    <_6:component rdf:nodeID="jTCXOrie5"/>
    <_6:component rdf:nodeID="jTCXOrie8"/>
    <_6:component rdf:nodeID="jTCXOrie7"/>
    <_6:component rdf:nodeID="jTCXOrie6"/>
    <_6:component rdf:nodeID="jTCXOrie9"/>
    <_6:prodid>-//connolly.w3.org//palmagent 0.6 (BETA)//EN</_6:prodid>
    <_6:version>2.0</_6:version>
  </rdf:Description>

  <rdf:Description rdf:nodeID="jTCXOrie5">
    <_6:location xml:lang="en">Paris, France</_6:location>
    <rdf:type rdf:resource="http://www.w3.org/2002/12/cal/icaltzd#Vevent"/>
    <_6:url rdf:resource="http://www.w3.org/2001/sw/grddl-wg/doc29/robin-hcal-grddl.html"/>
    <_6:summary xml:lang="en">Fashion Expo</_6:summary>
    <_6:dtstart rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2006-10-20</_6:dtstart>
    <_6:dtend rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2006-10-23</_6:dtend>
  </rdf:Description>

  <rdf:Description rdf:nodeID="jTCXOrie6">
    <_6:location xml:lang="en">Cologne, Germany</_6:location>
    <rdf:type rdf:resource="http://www.w3.org/2002/12/cal/icaltzd#Vevent"/>
    <_6:url rdf:resource="http://www.w3.org/2001/sw/grddl-wg/doc29/robin-hcal-grddl.html"/>
    <_6:summary xml:lang="en">New line review</_6:summary>
    <_6:dtstart rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2006-10-26</_6:dtstart>
    <_6:dtend rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2006-10-28</_6:dtend>
  </rdf:Description>
  ...
</rdf:RDF>

```

<http://www.w3.org/2002/12/cal/glean-hcal.xsl>

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
...>
<xsl:output indent="yes" method="xml" />
...
<xsl:template match="/">
  <r:RDF>
    <c:Vcalendar r:about="{ $Source }">
      <c:prodid> <xsl:value-of select="$prodid" /> </c:prodid>
      <c:version>2.0</c:version>
      <xsl:apply-templates />
    </c:Vcalendar>
  </r:RDF>
</xsl:template>
...
<!-- don't pass text thru -->
<xsl:template match="text()" />
</xsl:stylesheet>
```

- Christine PORQUET - ENSICAEN

- 89 -

SKOS (1/2)

Simple Knowledge Organisation System

<http://www.w3.org/TR/skos-primer/>

“OWL is a Harley-Davison, SKOS is a mountain bike”

Tom Baker (SW deployment working group)

- SKOS is geared towards specific (though large) use cases, like taxonomies, glossaries, thesauri...
- SKOS is based on a very simple usage of OWL
 - the emphasis is on *organization* and not on logical inferences.
 - a **bridging technology**, providing the missing link between the rigorous logical formalism of ontology languages such as OWL and the chaotic, informal and weakly-structured world of Web-based collaboration tools, as exemplified by social tagging applications.

- Christine PORQUET - ENSICAEN

- 90 -

SKOS (2/2)

- . SKOS is based on a simple structure
 - the central concept is a **SKOS concept**
 - concepts can have preferred and alternate labels
 - a concept may be **narrower** or **broader** than another one
 - concepts may be related to one another
 - concepts can be collected in “concept schemes”
- . Other resources can then refer to these concepts as, eg, their subject.

Exemple

```
ex:animals rdf:type skos:Concept;  
  skos:prefLabel "animals"@en;  
  skos:prefLabel "animaux"@fr;  
  skos:altLabel "creatures"@en;  
  skos:narrower ex:mammals.
```

```
ex:mammals rdf:type skos:Concept;  
  skos:prefLabel "mammals"@en;  
  skos:broader ex:animals.
```

```
ex:birds rdf:type skos:Concept;  
  skos:prefLabel "birds"@en;  
  skos:related ex:ornithology.
```

```
ex:ornithology rdf:type skos:Concept;  
  skos:prefLabel "ornithology"@en.
```



As of July 2009

UNE INTRODUCTION AU WEB SEMANTIQUE ET AUX ONTOLOGIES RESSOURCES

Le texte fondateur de Tim Berners-Lee et al. paru dans le Scientific American (mai 2001) :

<http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2>

et sa traduction française : <http://www.urfist.cict.fr/archive/lettres/lettre28/lettre28-22.html>

Activités du W3C liées au développement du web sémantique. On trouve notamment à cette adresse toutes les recommandations du W3C concernant RDF, RDFS et OWL : <http://www.w3.org/2001/sw/>

Voir en particulier :

- la liste des outils : <http://esw.w3.org/topic/SemanticWebTools>
- la « pièce montée » qui n'en est plus une : <http://www.w3.org/2007/03/layerCake.png>

Liste de toutes les traductions françaises officielles de documents du W3C :

<http://www.w3.org/2003/03/Translations/byLanguage?language=fr>

Diaporamas des chapitres 3 (sur RDF et RDFS) et 4 (sur OWL) du livre *A Semantic Web Primer* - Grigoris Antoniou & Frank van Harmelen :

- <http://www.ics.forth.gr/isl/swprimer/presentations/Chapter3.ppt>
- <http://www.ics.forth.gr/isl/swprimer/presentations/Chapter4.ppt>

Tutoriel de l'équipe Edelweiss (INRIA Sophia-Antipolis) :

<http://www-sop.inria.fr/teams/edelweiss/wiki/wakka.php?wiki=CoreseTutorial>

Linking Open Data : <http://esw.w3.org/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

Dublin Core Metadata Initiative : <http://dublincore.org/documents/2000/07/16/usageguide/references.shtml>

FOAF : Friend Of A Friend: <http://www.foaf-project.org/>

Voir aussi :

- Spécification technique : *FOAF Vocabulary Specification* : <http://xmlns.com/foaf/0.1/>
- L'article de Wikipedia : <http://fr.wikipedia.org/wiki/FOAF>

DBpedia : extraction d'information structurée à partir de Wikipedia : <http://dbpedia.org>

Editeur d'ontologie Protégé – documentation, tutoriaux, etc. : <http://protege.stanford.edu/doc/users.html>

Bibliothèque d'ontologies : http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library#OWL_ontologies

Communauté de développement autour de OWL : Projets CO-ODE et HyOntUse :

<http://www.co-ode.org/resources/>

Voir en particulier :

- Le tutoriel : *Ontology Design Patterns and Problems Practical Ontology Engineering using Protege-OWL*. <http://www.co-ode.org/resources/tutorials/iswc2005/ISWC-Tutorial-Best-practice-2005.ppt>

Langage d'interrogation SPARQL pour RDF : <http://www.yoyodesign.org/doc/w3c/rdf-sparql-query/>

Introduction à GRDDL : <http://www.yoyodesign.org/doc/w3c/grddl-primer/>

SKOS : Simple Knowledge Organization System Primer : <http://www.w3.org/TR/skos-primer/>

OWL 2 Web Ontology Language Primer : <http://www.w3.org/TR/2009/WD-owl2-primer-20090611/>

API Java OWL API : <http://owlapi.sourceforge.net/index.html>

Navigateur OwlSight : <http://pellet.owldl.com/ontology-browser/>