

Ingénierie ontologique

Yannick Prié

UFR Informatique

Université Claude Bernard Lyon 1

Objectifs

- Comprendre ce que sont les ontologies formelles
- Découvrir les logiques de description comme langage de représentation des connaissances
- Découvrir OWL (Ontology Web Language) dans le contexte du Web sémantique
- Concevoir une ontologie en suivant une méthodologie

Plan

- Ingénierie des connaissances
- Ontologies
- Logiques de description
- Introduction au Web sémantique
- OWL
- Conception d'ontologies

Connaissance ?

- Une connaissance se manifeste en acte, est liée à une action
 - elle est révélée par une capacité à agir dans un monde
- Au niveau Intelligence Artificielle
 - Connaissance = information (donnée) utilisée dans une inférence ou un raisonnement
 - Une inscription qui permet une inférence est une inscription de connaissances
- Exemples de connaissances
 - Connaissances du domaine
 - Concepts manipulés et leur relations. Description d'un domaine. Contraintes
 - Connaissances stratégiques, savoirs faire, procédures
 - Comment une tâche va être effectuée

Représentation des connaissances

- Approche d'inscription symbolique
- Approche de modélisation et de formalisation
 - Modéliser pour éliciter des connaissances
 - construire des modèles de connaissances
 - Formaliser pour pouvoir
 - représenter des connaissances en machine
 - raisonner automatiquement dessus
- Représentation de connaissances
 - Inscription de connaissances formelles à l'aide d'un langage de représentation de connaissances possédant une sémantique formelle permettant de faire des inférences

Représentation de connaissances

- Système définissant une série de symboles et une série d'opérations sur ces symboles + interprétation
 - En tant que représentations, les symboles ont un contenu sémantique mais les opérations qu'on leur associe ne sont déterminées que par leur syntaxe
 - Les inférences sont des déductions formelles donnant de nouvelles représentations
- Enjeu de la représentation des connaissances :
 - Que les représentations données soient significatives pour un humain
 - Que les représentations déduites soient toujours significatives pour un humain !

Systemes à base de connaissances

- Systemes informatiques mobilisant explicitement des modèles de connaissances
 - Manipulation d'expressions symboliques (ou inscriptions), règles de réécriture symboliques
 - L'utilisateur utilise (en les interprétant) les inscriptions dans le cadre d'une tâche
- SBC = « Systemes sémiotiques de manipulation d'inscriptions symboliques dont le fonctionnement informatique doit permettre à l'utilisateur d'interpréter et de comprendre le système dans le cadre de son activité, en utilisant les termes du domaine »

Ingénierie des connaissances

- SBC = SI
 - systèmes techniques plongés dans des systèmes d'usage
- « L'ingénierie des connaissances correspond à l'étude de modèles symboliques formels plongés dans des systèmes d'usage : c'est l'ingénierie informatique et logique des modèles en fonction des usages qu'ils rendent possibles et des appropriations qu'ils permettent »

Concevoir des SBC

- Trouver les expressions linguistiques permettant d'explicitier le problème à traiter
- Proposer un modèle conceptuel au « niveau des connaissances » permettant de décrire les objets et les raisonnements
- Proposer une opérationnalisation en machine des connaissances
- Et le SBC qui utilise cette dernière

Questions de l'IC

- Quels modèles pour les bases de connaissances ?
 - Type de modèle (domaine, raisonnement, etc.)
- Quels langages et formalismes pour décrire les modèles ?
 - Langages de modélisation formelle : logique, LD, GC...
- Comment construire les modèles nécessaires à une application, suivant quelles étapes, avec quels outils ?
 - Méthodes et techniques : KADS...
- Comment construire une application à base de connaissances liée à un modèle ?
 - Navigation, inférences, explications
- Comment faire évoluer un modèle ?
 - Maintenance et gestion
- ...

Domaines associés

- Linguistique
 - Traitement Automatique des Langues, terminologie
- Logique
 - Raisonnement, logiques formelles
- Ergonomie
 - IHM
- Epistémologie
 - Connaissances et inscriptions
- Anthropologie, sociologie
 - Usages, partage
- Sciences de gestion
 - Gestion des connaissances
- Génie logiciel
- ...

CFP conférence IC 2009

- Ontologies : conception, représentations et exploitations
- Web de connaissances
- Dimensions individuelle, collective et sociale des connaissances
- Représentation des connaissances
- Conception et génération de modèles de connaissances
- Traitements et raisonnement sur des connaissances
- Le temps et l'espace dans la gestion de connaissances
- Evolution et historique des modèles à base de connaissances
- Conception d'interactions et interfaces avec des systèmes à base de connaissances
- Propriété, sécurité et confidentialité dans les systèmes à base de connaissances
- Applications et retour d'expérience en ingénierie de connaissances
- Ingénierie des systèmes à base de connaissances
- Développements théoriques et interdisciplinaires de l'ingénierie des connaissances

Plan

- Ingénierie des connaissances
- **Ontologies**
- Logiques de description
- Introduction au Web sémantique
- OWL
- Conception d'ontologies

Qu'est ce qu'une ontologie ?

- Philosophie
 - Partie de la métaphysique qui s'applique à « l'être en tant qu'être » (Aristote), indépendamment de ses déterminations particulières (Petit Robert)
 - étude des propriétés de l'être
- Informatique / ingénierie des connaissances
 - « Specification of a shared conceptualisation » (Gruber)
 - Conceptualisation : choix sur la description de quelque chose
 - Spécification : formalisation

Histoire de définitions

- Une des premières définitions :
 - « 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 et al. 91]
- Suivie par une définition plus large :
 - « an ontology is an explicit specification of a conceptualization » [Gruber 93]
- Définition de Gruber plus élaborée :
 - « Ontologies are defined as a formal specification of a shared conceptualization » [Borst et al. 97]
 - « Conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used, and the constraints on their use are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared refers to the notion that an ontology captures consensual knowledge, that is, is not primitive to some individual, but accepted by a group. [Studer et al. 98]

Histoire de définitions (2)

- Autre définition
 - « An ontology may take a variety of forms, but necessarily it will include a vocabulary of terms, and some specification of their meaning. This includes definitions and an indication of how concepts are inter-related which collectively impose a structure on the domain and constrain the possible interpretation of terms. An ontology is virtually always the manifestation of a shared understanding of a domain that is agreed between a number of agents. Such agreement facilitates accurate and effective communication of meaning, which in turn leads to other benefits such as inter-operability, reuse and sharing ». [Uschold 98]
- Définition plus large
 - « We consider ontologies to be domain theories that specify a domain-specific vocabulary of entities, classes, properties, predicates, and functions, and to be a set of relationships that necessarily hold among those vocabulary terms. Ontologies provide a vocabulary for representing knowledge about a domain and for describing specific situations in a domain » [Fikes et al. 99]

Gruber 2007

- "In the context of computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class members). The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application..." (<http://tomgruber.org/writing/ontology-definition-2007.htm>)
- La littérature est pleine de définitions différentes du terme ontologie. Chaque communauté adopte sa propre interprétation selon l'usage qui en est fait et le but visé.

En bref, une ontologie est

- Un produit d'ingénierie :
 - composé d'un certain vocabulaire utilisé pour décrire de façon explicite et consensuelle une certaine réalité (conceptualisation),
 - et d'un ensemble d'hypothèses explicites sur le sens des termes,
 - exprimé comme un ensemble d'objets et de relations entre eux, décrit avec un langage formel (spécification)
- utilisé comme outil de communication
 - entre humains
 - entre humain et machine
 - entre machines

Différents degrés ontologiques

- Plus ou moins
 - de formalisation
 - de sémantique formelle
 - de possibilités inférentielles
- Quelques variations
 - Vocabulaire contrôlé
 - Référentiel métier
 - Liste d'autorité
 - Terminologie
 - Thésaurus
 - Système de classes d'une représentation par objet
 - Ontologie formelle

Vocabulaire contrôlé / terminologie

- Vocabulaire contrôlé
 - Ensemble de termes reconnus, fixés, inaltérables, normalisés et validés par un groupe (une communauté de pratiques) utilisés pour indexer ou analyser le contenu et pour rechercher de l'information dans un domaine d'information défini.
 - Les définitions doivent être non ambiguës et non redondantes. Le vocabulaire contrôlé garantit ainsi qu'un sujet sera décrit avec les mêmes termes préférentiels. Si plusieurs termes désignent un même concept, un seul d'entre eux sera choisi et identifié comme le "terme préféré" (descripteur autorisé), les autres seront listés comme synonymes.
 - Utilisation : indexation, traduction, etc.
- Terminologie :
 - Vocabulaire contrôlé accompagné de définitions qui en fixent le sens.

Exemple de vocabulaire contrôlé

Accessibilité (pour personne handicapée)

- **Employer pour**
 - Accès aux fauteuils roulants
- [Barrier free access](#)

Accident

- **Employer pour**
 - Collision
- [Accidents](#)

Accident vasculaire cérébral

- [Stroke](#)

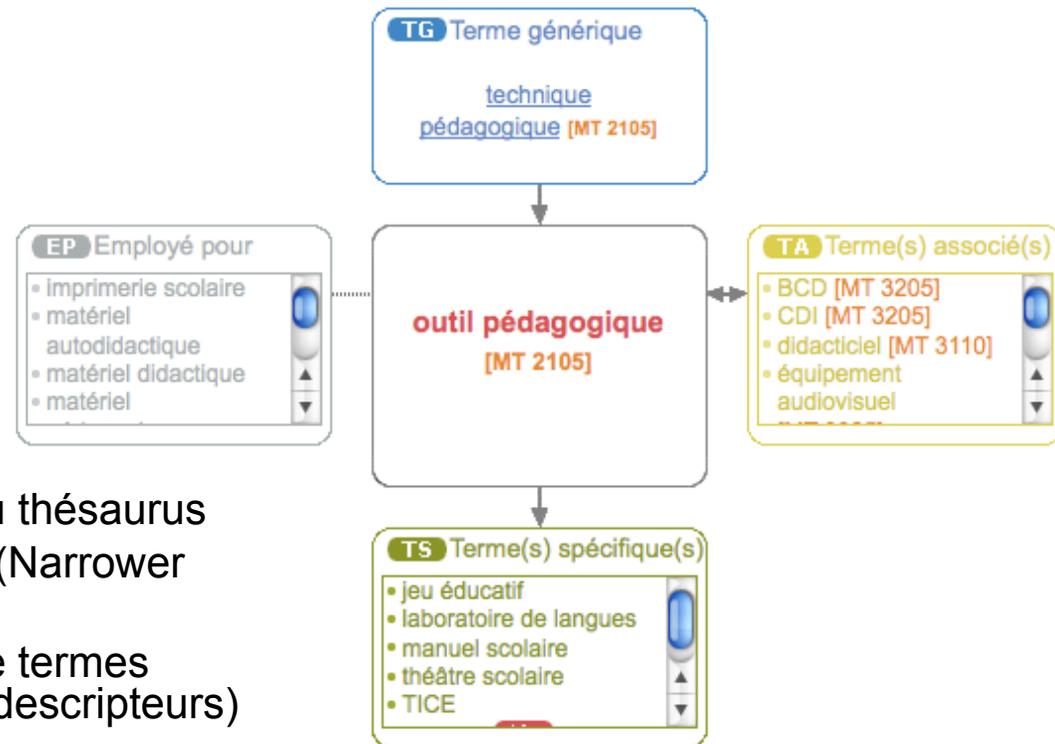
Accord

- **Employer pour**
 - Alliance
 - Convention
 - Entente
 - Protocole d'entente
 - Traité
- **Voir aussi**
 - **Accord international**
- [Agreements](#)

<http://www.hc-sc.gc.ca/>

Thesaurus

- Un réseau de termes organisé avec plusieurs types de relations
 - relation hiérarchique
 - base de la hiérarchie du thésaurus
 - BT (Broader Term), NT (Narrower Term), TT (Top Term)
 - relation d'équivalence (entre termes descripteurs et termes non-descripteurs)
 - base de l'univocité
 - USE (Preferred Term), UF (Use for, non preferred synonym, quasi synonym)
 - relation d'association (entre descripteurs)
 - enrichissement sémantique ; sujets connexes.
 - RT (Related Term - other than BT, NT, TT, etc.), SN (Scope Note, note pour expliquer un terme)



<http://www.thesaurus.motbis.cndp.fr/>

- Différentes normes ISO
- Côté WS, voir SKOS

Ontologie formelle

- Modélisation de connaissances exprimée dans un langage de représentation des connaissances
 - sémantique formelle
- Exemple (cf. Acacia / INRIA)

```
concept Object
```

```
concept Engine < Object
```

```
concept Vehicle < Object
```

```
relation partOf :
```

```
Object -> Object
```

```
Engine e1 partOf Vehicle v2
```

```
Engine breakdown => Vehicle stop
```

Construire une ontologie pour...(1)

- *Partager la compréhension commune de la structure de l'information entre des personnes ou entre des agents logiciels*
 - Standardisation, réutilisation pour indexation, base de connaissances, etc.
 - Exemple
 - une ontologie médicale permet de décrire de la même manière des sites web ou des services médicaux en ligne
 - des agents logiciels peuvent agréger l'information, offrir des services de recherche, réutiliser l'information, etc.

Construire une ontologie pour...(2)

- *Permettre la réutilisation du savoir sur un domaine*
 - En tant que spécification d'un conceptualisation d'un domaine, une ontologie peut être réutilisée telle quelle et améliorée, plutôt qu'être reconstruite à partir de rien.
 - ontologie de domaine
 - eg. médecine
 - ontologie trans-domaine
 - représentation du temps
 - haut-niveau

Construire une ontologie pour...(3)

- *Expliciter ce qui est considéré comme implicite sur un domaine*
 - une spécification explicite permet de construire des systèmes qui dépendent explicitement d'une modélisation des connaissances (vs. des assumptions implicites dispersées dans le code)
 - il est donc possible de changer cette modélisation sans changer le système
 - on peut plus facilement comprendre la modélisation que comprendre ce qui est sous-jacent dans le code
- *Distinguer le savoir de domaine du savoir opérationnel*
 - Permet de décrire des tâches indépendamment des objets sur lesquels elles s'appliquent
 - Exemple : configuration à partir de composants.

Composants d'une ontologie : concepts / classes

- Un concept représente un ensemble d'objets et leurs propriétés communes.
- Décrit par un terme
 - eg. Voiture, Vache, Violon
- Un concept peut avoir
 - une définition intensionnelle : CNS pour appartenir au concept
 - eg. Véhicule de transport automobile conçu et aménagé pour le transport d'un petit nombre de personnes
 - une définition extensionnelle : description exhaustive de tout ce qui obéit à la définition
 - eg. la liste de toutes les voitures du monde
- A lier à une approche triadique du concept (Aristote, Frege, etc.)
 - terme, intension, extension : mot, sens, objets

Composants d'une ontologie : individus

- Instances de concepts
- Les éléments décrits par les classes
- Exemples :
 - la voiture de Jean
 - Blanchette, Marguerite, Rousette

Composants d'une ontologie : propriétés

- On différencie en général
 - attributs : propriétés simples
 - eg. age, nom, nombre-de-roues, etc.
 - rôles, relations sémantiques : association entre concepts
 - eg. parent-de, partie-de, proche-de, sous, contient, connecté à, etc.
- On peut y associer des facettes
 - valeurs possibles d'attributs
 - eg. $0 < \text{age} < 150$
 - restrictions de rôles
 - eg. pas plus de 2 parents

Composants d'une ontologie

relations de spécialisation

- Relation binaire entre un concept général et un concept plus spécifique.
- Relation inverse = généralisation
- Expression d'une inclusion ensembliste en terme sémantique
- Noms variés
 - is-a, est un, est une sorte de
- Induit une hiérarchie de spécialisation, ou une taxinomie
- Remarque
 - les propriétés peuvent aussi être organisées en hiérarchie de spécialisation

Taxinomie des pains



Pain

Pain spécial

Pain régional



Pain complet



Pain aux lardons



Pain blanc

Pain de seigle

Pain au son

Pain aux noix et aux raisins

Pain auvergnat

Couronne



Pain marguerite

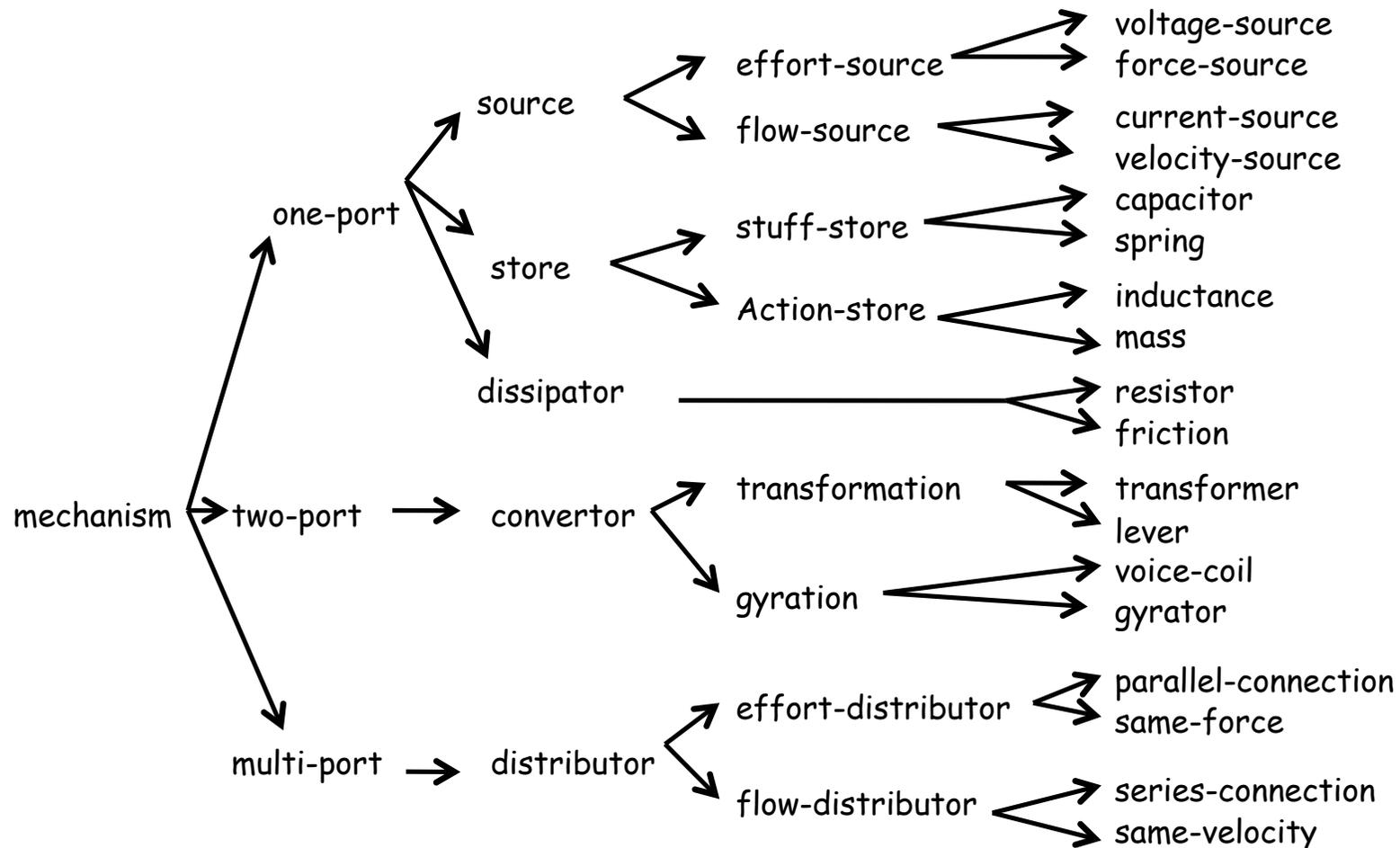
Lyonnaise

Pain vaudois



Taxinomie de mécanismes physiques

(Borst)



Connectivity

Type

Effort/flow

Domain (e.g. el, me)

Taxinomies et ontologies

- Taxinomie/Taxonomie : classification d'éléments (Petit Robert)
- Origines
 - arbre de Porphyre, classification des êtres vivants et des fossiles, etc.
- Naturelle pour l'homme qui fonctionne souvent par abstraction et association
- Structure à la base de deux inférences élémentaires que nous faisons tous les jours
 - l'identification : capacité à reconnaître la classe d'un objet à partir de ses caractéristiques
 - la spécialisation : capacité à prendre en compte des catégories de niveaux de précision variables

Les composants d'une ontologie

restrictions, règles, axiome

- Pour exprimer ce qui ne peut l'être comme concept ou propriété
- Restriction : chose qui doit être vraie pour que ce qui est exprimé soit valable
 - cf. restrictions de rôles
- Règle : affirmation sous la forme antécédent > conséquent décrivant des inférences possibles
 - "Louis XIV" est le même individu que "Roi soleil"
 - une voiture rare est chère
 - $\text{uncle}(x, y) \leftarrow \text{brother}(x, z) \wedge \text{father}(z, y)$
- Axiome : assertion générale sur les fondements de l'ontologie
 - *partOf* est transitive
 - *parent-de* est l'inverse de *enfant-de*

Diversité des ontologies (1)

- Ontologies de représentation
 - définissent un ensemble de primitives de représentation
 - exemple : la définition des éléments de RDF(S) : classe, propriétés, relations de sous-classe, de sous-propriété, etc.
- Ontologies génériques
 - exemples : ontologies spatiale, temporelle
- Ontologies de domaine : les plus classiques, description structurelle, fonctionnelle, causale, etc.
 - Applications précises
 - Energies marines (EDF), Textes juridiques (Mondeca pour Lexis-Juris), Dossier patient (Menelas)...
 - Famille d'application
 - Documents audiovisuels (INA), STEP (conception), TOVE (modélisation de l'entreprise)...
 - Communauté « mondiale » d'utilisateurs qui peut être très large
 - CRM (Musées), Gene ontology (génétique)...

Diversité des ontologies (2)

- Ontologies de tâche / méthode - connaissances de résolution de problème
 - le rôle joué par chaque concept dans une méthode particulière est rendu explicite : conception, diagnostic, évaluation, planifications
- Ontologies d'application
 - peuvent être vues comme une double spécialisation : d'une ontologie du domaine et d'une ontologie de méthode

Diversité des ontologies (3)

- Autres distinctions
 - Ontologies générales, abstraites, de haut niveau
 - eg. catégories conceptuelles : objet, événement, état, processus, action, temps, espace
 - vs Ontologies spécialisées
 - Ontologies théoriques
 - eg. physique, mathématique, cinématique
 - vs Ontologies pragmatiques
- Beaucoup de différences
 - objectifs de construction
 - conditions de construction et d'évolution
 - niveaux d'expressivité
 - niveaux de formalisation
 - fondations logiques, capacités inférentielles
 - niveaux de réutilisation
 - ...

Geonames ontology

- <http://www.geonames.org/ontology/>
- The GeoNames Ontology makes it possible to add geospatial semantic information to the Word Wide Web. All over 6.2 million geonames toponyms now have a unique URL with a corresponding RDF web service. Other services describe the relation between toponyms.
- Depending on applicability the following documents are available for a Feature :
 - The children (countries for a continent, administrative subdivisions for a country, ...).
 - The neighbours (neighbouring countries).
 - Nearby features. Nearby to the Eiffel Tower are Champ de Mars, Trocadéro - Palais de Chaillot...
 - isOfficialName, locatedIn, etc.
- Exemple
 - <http://geotree.geonames.org/2968254/>

Wordnet

- <http://wordnet.princeton.edu/>
- WordNet® is a large lexical database of English, developed under the direction of George A. Miller. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The resulting network of meaningfully related words and concepts can be navigated with the browser. WordNet is also freely and publicly available for download. WordNet's structure makes it a useful tool for computational linguistics and natural language processing.
- Relations (<http://www.shiffman.net/teaching/a2z/wordnet/>)
 - All parts of speech
 - Synonymy / Antonymy
 - Nouns only
 - Hypernymy. / Hyponymy
 - Meronymy. Meronymy refers to a part/whole relationship. For example, paper is a meronym of book, since paper is a part of a book
 - Verbs only
 - Troponymy. Troponymy is the semantic relationship of doing something in the manner of something else. For example, “walk” is a troponym of “move” and “limp” is a troponym of “walk.”
 - Entailment. Entailment refers to the relationship between verbs where doing something requires doing something else. If you are snoring, you must be sleeping so sleeping is entailed by snoring.
- Une "Ontologie linguistique"
 - pas vraiment de sémantique formelle
- EuroWordnet

Wordnet (suite)

- Dog
 - Noun
 - S: (n) dog, domestic dog, *Canis familiaris* (a member of the genus *Canis* (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds) "the dog barked all night"
 - S: (n) frump, dog (a dull unattractive unpleasant girl or woman) "she got a reputation as a frump"; "she's a real dog"
 - S: (n) dog (informal term for a man) "you lucky dog"
 - S: (n) cad, bounder, blackguard, dog, hound, heel (someone who is morally reprehensible) "you dirty dog"
 - S: (n) frank, frankfurter, hotdog, hot dog, dog, wiener, wienerwurst, weenie (a smooth-textured sausage of minced beef or pork usually smoked; often served on a bread roll)
 - S: (n) pawl, detent, click, dog (a hinged catch that fits into a notch of a ratchet to move a wheel forward or prevent it from moving backward)
 - S: (n) andiron, fire dog, dog, dog-iron (metal supports for logs in a fireplace) "the andirons were too hot to touch"
 - Verb
 - S: (v) chase, chase after, trail, tail, tag, give chase, dog, go after, track (go after with the intent to catch) "The policeman chased the mugger down the alley"; "the dog chased the rabbit"

CYC - OpenCyc

- <http://www.cyc.com/> / <http://www.opencyc.org/>
- Tentative d'ontologie universelle, démarrée en 1984 par Doug Lenat
 - The Cyc knowledge base (KB) is a formalized representation of a vast quantity of fundamental human knowledge: facts, rules of thumb, and heuristics for reasoning about the objects and events of everyday life. The medium of representation is the formal language [Cycl](#), described below. The KB consists of terms--which constitute the vocabulary of Cycl--and assertions which relate those terms. These assertions include both simple ground assertions and rules. Cyc is not a frame-based system: the Cyc team thinks of the KB instead as a sea of assertions, with each assertion being no more "about" one of the terms involved than another.
 - The Cyc KB is divided into many (currently thousands of) "microtheories", each of which is essentially a bundle of assertions that share a common set of assumptions; some microtheories are focused on a particular domain of knowledge, a particular level of detail, a particular interval in time, etc. The microtheory mechanism allows Cyc to independently maintain assertions which are *prima facie* contradictory, and enhances the performance of the Cyc system by focusing the inferencing process.
 - At the present time, the Cyc KB contains nearly two hundred thousand terms and several dozen hand-entered assertions about/involving each term. New assertions are continually added to the KB by human knowledge enterers. Additionally, term-denoting functions allow for the automatic creation of millions of non-atomic terms, such as (`LiquidFn Nitrogen`); and Cyc adds a vast number of assertions to the KB by itself as a product of the inferencing process.
- Une version pour le Web : OpenCYC
 - OpenCyc is the open source version of the [Cyc](#) technology, the world's largest and most complete general knowledge base and commonsense reasoning engine. OpenCyc can be used as the basis of a wide variety of intelligent applications such as:

UMLS

- Unified Medical Language System
- <http://www.nlm.nih.gov/research/umls/>
- The purpose of NLM's Unified Medical Language System[®] (UMLS) is to facilitate the development of computer systems that behave as if they "understand" the meaning of the language of biomedicine and health. To that end, NLM produces and distributes the UMLS Knowledge Sources (databases) and associated software tools (programs) for use by system developers in building or enhancing electronic information systems that create, process, retrieve, integrate, and/or aggregate biomedical and health data and information, as well as in informatics research. By design, the UMLS Knowledge Sources are multi-purpose. They are not optimized for particular applications, but can be applied in systems that perform a range of functions involving one or more types of information, e.g., patient records, scientific literature, guidelines, and public health data. The associated UMLS software tools assist developers in customizing or using the UMLS Knowledge Sources for particular purposes. The lexical tools work more effectively in combination with the UMLS Knowledge Sources, but can also be used independently.

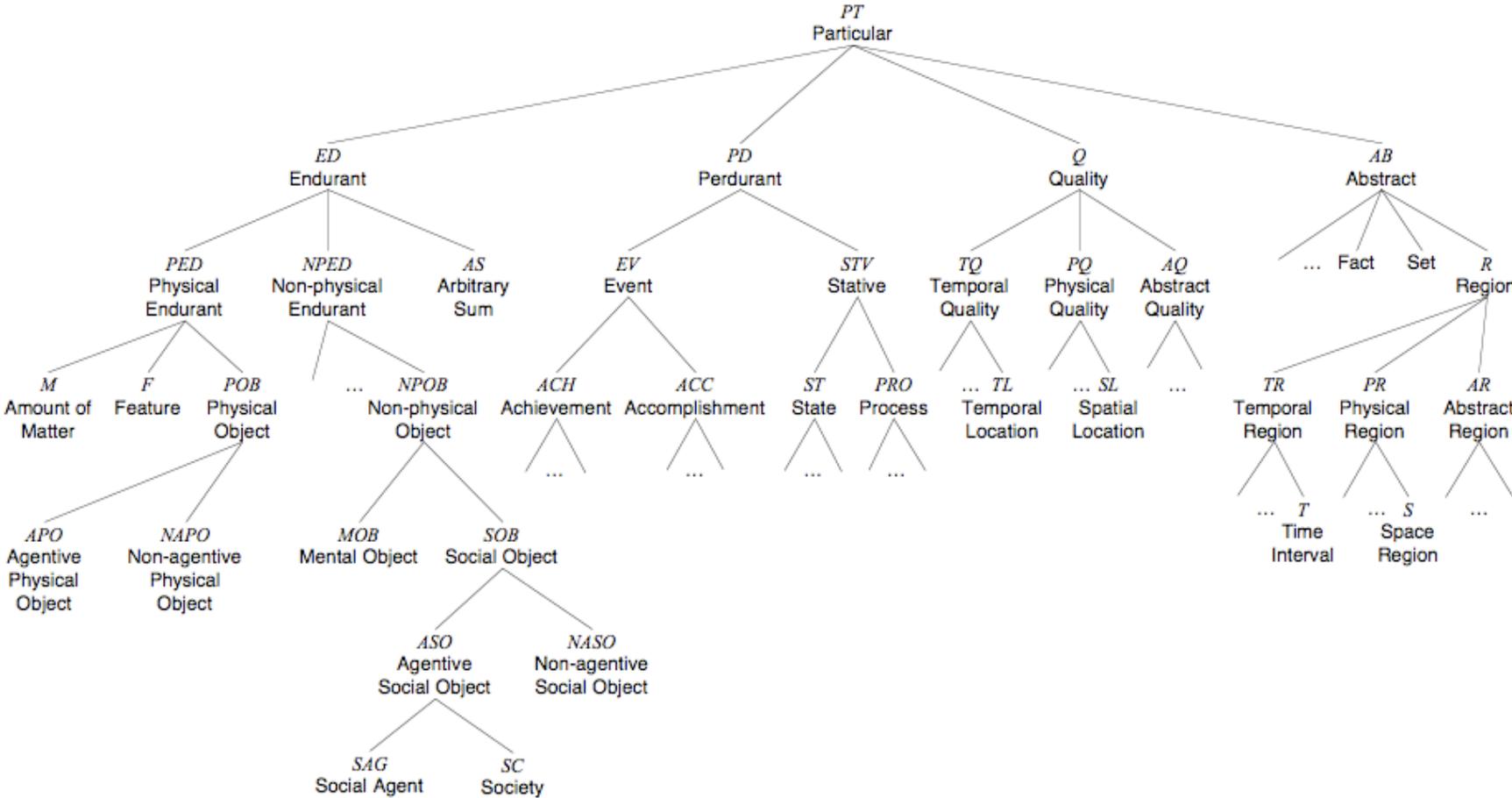
UMLS (suite)

- Metathesaurus
 - The Metathesaurus is a very large, multi-purpose, and multi-lingual vocabulary database that contains information about biomedical and health related concepts, their various names, and the relationships among them. It is built from the electronic versions of many different thesauri, classifications, code sets, and lists of controlled terms used in patient care, health services billing, public health statistics, indexing and cataloging biomedical literature, and /or basic, clinical, and health services research. In this documentation, these are referred to as the "source vocabularies" of the Metathesaurus. In the Metathesaurus, all the source vocabularies are available in a single, fully-specified database format.
- Semantic Network
 - The purpose of the Semantic Network is to provide a consistent categorization of all concepts represented in the UMLS Metathesaurus and to provide a set of useful relationships between these concepts. All information about specific concepts is found in the Metathesaurus; the Network provides information about the set of basic semantic types, or categories, which may be assigned to these concepts, and it defines the set of relationships that may hold between the semantic types. The current release of the Semantic Network contains 135 semantic types and 54 relationships. The Semantic Network serves as an authority for the semantic types that are assigned to concepts in the Metathesaurus. The Network defines these types, both with textual descriptions and by means of the information inherent in its hierarchies.

DOLCE

- <http://www.loa-cnr.it/DOLCE.html>
- WONDERWEB Foundational Ontologies Library are to serve as:
 - starting point for building new ontologies. One of the most important and critical questions when starting a new ontology is determining what things there are in the domain to be modeled. Adopting a high level view provides an enormous jump start in answering this question;
 - a reference point for easy and rigorous comparisons among different ontological approaches;
 - a common framework for analyzing, harmonizing and integrating existing ontologies and metadata standards (by manually mapping existing categories into the categories assumed by some module(s) in the library).
- In addition, we intend the library to be:
 - minimal – as opposed to other comprehensive ontology efforts, we intend the library to be as general as possible, including only the most reusable and widely applicable upper-level categories;
 - rigorous – where possible, the ontologies in the libraries will be characterized by means of rich axiomatisations, and the formal consequences (theorems) of such characterizations will be explored in detail;
 - extensively researched – each module in the library will be added only after careful evaluation by experts and consultation with canonical works. The basis for ontological choices will be documented and referenced.

DOLCE



SWEET

- Semantic Web for Earth and Environmental Terminology
- <http://sweet.jpl.nasa.gov/ontology>
- The SWEET project « provides a common semantic framework for various Earth science initiatives. The semantic web is a transformation of the existing web that will enable software programs, applications, and agents to find meaning and understanding on web pages. SWEET developed these capabilities in the context of finding and using Earth science data and information »
- Sweet 1.0
 - Earth Realm
 - Physical Phenomena
 - Physical Process
 - Physical Property
 - Physical Substance
 - Sun Realm
 - Biosphere
 - Data
 - Data Center
 - Human Activity
 - Material Thing
 - Numerics
 - Sensor
 - Space
 - Time
 - Units

Enterprise Ontology

- <http://www.aiai.ed.ac.uk/project/enterprise/enterprise/ontology.html>
- The Enterprise Ontology is a collection of terms and definitions relevant to business enterprises
- Activity
 - Activity Specification, Execute, Executed Activity Specification, T-Begin, T-End, Pre-Conditions, Effect, Doer, Sub-Activity, Authority, Activity Owner, Event, Plan, Sub-Plan, Planning, Process Specification, Capability, Skill, Resource, Resource Allocation, Resource Substitute.
- Organisation
 - Person, Machine, Corporation, Partnership, Partner, Legal Entity, Organisational Unit, Manage, Delegate, Management Link, Legal Ownership, Non-Legal Ownership, Ownership, Owner, Asset, Stakeholder, Employment Contract, Share, Share Holder.
- Strategy
 - Purpose, Hold Purpose, Intended Purpose, Strategic Purpose, Objective, vision, Mission, Goal, Help Achieve, Strategy, Strategic Planning, Strategic Action, Decision, Assumption, Critical Assumption, Non-Critical Assumption, Influence Factor, Critical Influence Factor, Non-Critical Influence Factor, Critical Success Factor, Risk.
- Marketing
 - Sale, Potential Sale, For Sale, Sale Offer, Vendor, Actual Customer, Potential Customer, Customer, Reseller, Product, Asking Price, Sale Price, Market, Segmentation Variable, Market Segment, Market Research, Brand Image, Feature, Need, Market Need, Promotion, Competitor.
- Time
 - Time Line, Time Interval, Time Point.

SBML

- Systems Biology Markup Language (SBML)
- http://sbml.org/Main_Page
- The Systems Biology Markup Language (SBML) is a computer-readable format for representing models of biochemical reaction networks. SBML is applicable to metabolic networks, cell-signaling pathways, regulatory networks, and many others. »
- The starting point is an appreciation that computational modeling of biological systems is no longer a fringe activity—it's a requirement for us to make sense of our vast and ever-expanding quantities of data. At its most basic, it is no different from modeling as it's practiced by all scientists, whether in biology or elsewhere. The extra but crucial step is casting the model into a formal, computable form that can be analyzed rigorously using simulation and other methods.
- Different representations are useful for different purposes. Graphical diagrams of biochemical reaction networks are useful for visual presentation to humans, but at the level of software, a different format is needed for quantifying a model to the point where it can be simulated and analyzed. This is where the Systems Biology Markup Language (SBML) comes in.
- Simply put, SBML is a machine-readable format for representing models. It's oriented towards describing systems of biochemical reactions of the sort common in research on a number of topics, including cell signaling pathways, metabolic pathways, biochemical reactions, gene regulation, and many others.

Discussion (1)

- Les instances font-elle partie de l'ontologie ?
 - pour certains oui
 - pour d'autres
 - ontologie = classes
 - ontologie et instances = base de connaissances
- Comment concevoir une ontologie ?
 - conception toujours difficile, comme pour n'importe quel produit d'ingénierie
 - dépend des objectifs
 - voir la dernière partie du cours

Discussion (2)

- Différence entre une ontologie et un schéma de base de données ?
 - Les ontologies jouent un rôle analogue, mais
 - Un langage de spécification d'ontologies est syntaxiquement et sémantiquement plus riche qu'un langage de base de données
 - avec une BD on a une sémantique opérationnelle (programmation) de traitement des données. Avec une ontologie on a la possibilité de tenir des raisonnements indépendamment de l'utilisation effective des données de l'ontologie (inférence)
 - Une ontologie doit être une terminologie consensuelle et partagée, elle est naturellement orientée vers le partage et l'échange d'information
 - Une ontologie fournit une théorie du domaine et pas seulement la structure du « container » des données

Discussion (3)

- Que devient le langagier dans une ontologie ?
 - les concepts humains existent par les faits langagiers, les mots, les termes
 - les phénomènes de synonymie et d'ambiguïté existent partout dans la langue
 - chambre d'hôtel, chambre d'écho, chambre des députés, chambre d'enregistrement, chambre noire, chambre funéraire
 - voiture, auto, tacot...
 - construire une ontologie (une terminologie, un thesurus, etc.) oblige à se mettre d'accord sur le sens des termes employés dans une organisation, une communauté, un métier
 - un terme doit être un signifié normé (Rastier)
 - le choix du nom d'un concept est très important
 - à l'opposé : les tags
 - individualisation et puissance statistique

Discussion (4)

- Qu'est ce que l'engagement ontologique ?
 - Mettre en place et utiliser une ontologie nécessitent
 - un accord sur ce qui est exprimé par l'ontologie
 - au niveau du cadre
 - » vocabulaire (e.g. concepts, quantificateurs...)
 - au niveau signification / domaine
 - » modélisation, signification capturée par l'ontologie
 - donc un accord sur la conceptualisation partagée
 - engagement ontologique (commitment)
 - L'engagement ontologique est vérifié par le fait que l'agent (humain, machine) qui utilise l'ontologie a un comportement correspondant à ce qui y est exprimé

Discussion (6)

- Qu'est ce qu'une ontologie universelle ?
 - Un vieux rêve
 - décrire toute la connaissance humaine en machine avec une ontologie unique
 - une intéropérabilité générale entre tous les systèmes (!)
 - Problème :
 - impossible en théorie
 - l'humain est toujours là pour utiliser au final
 - ce qu'est le monde ne cesse de varier
 - on préfère construire des ontologies liées à des domaines plus restreints
 - certains cherchent à définir des ontologies de haut-niveau (universelles) à spécialiser
 - La question de l' "impérialisme ontologique"
 - en imposant mes schémas de données, j'impose ma vision du monde

Discussion (7)

- Dans une ontologie de domaine, qu'est ce qu'un domaine ?
 - Ce sur quoi porte l'accord de description
 - lié à une communauté de pratique
 - Les domaines peuvent être très variés
 - les technologies, l'informatique, les réseaux informatiques, les protocoles
 - des concepts différents avec les mêmes termes
 - eg. "feu" pour le service voirie d'une ville, "feu" pour le service lié à la circulation
 - deux personnes indépendantes feront une ontologie différente avec le même domaine
 - c'est quand on a fini l'ontologie qu'on connaît le domaine (explicitation de l'implicite)
 - Seul critère de validation : l'ontologie sert à quelque chose

Discussion (8)

- Quels sont les langages permettant de décrire des ontologies formelles ?
 - logique du 1er ordre
 - graphes conceptuels
 - logiques de description
 - programmes objet
 - ...
 - Outils
 - KIF (Knowledge Interchange Format), RDF Schema, DAML+OIL, OWL...

Plan

- Ingénierie des connaissances
- Ontologies
- Logiques de description
- Introduction au Web sémantique
- OWL
- Conception d'ontologies

Logiques de description

- **Famille** de langages de représentation des connaissances, offrant de bons compromis entre
 - lisibilité et formalisation
 - réseaux sémantiques, *frames*
 - *communication homme - machine*
 - expressivité et décidabilité/complexité
 - \subseteq logique du premier ordre
 - communication machine – machine
- Bons candidats pour représenter des ontologies

Principes de base (1)

Les termes du langage sont séparés en familles
disjointes :

- Les concepts (ou classes)
 - interprétés comme des sous-ensembles de Δ^3
 - *exemples : Personne, Étudiant, Document*
- Les rôles (ou propriétés)
 - interprétés comme des relations binaires sur Δ^3
 - *exemples : père, auteur*

Principes de base (2)

- Dans certaines LD, d'autres familles de termes sont utilisées :
 - Les individus
 - interprétés comme des éléments de Δ
 - exemples : pa_champin, cette_présentation
 - Les valeurs concrètes (rare)
 - interprétés comme des éléments de ΔD
 - ΔD Domaine concret
 - exemple : 42, 3.1415, "hello world"

Différentes LD

- Les diverses LD se différencient par :
 - Les termes complexes que l'on peut construire à partir de termes de bases
 - aussi appelés concepts définis vs concepts primitifs
 - exemple : « toute personne qui est l'auteur d'un document »
 - Les axiomes que l'on peut exprimer (« phrases »)
 - exemple : « un étudiant est toute personne qui suit un cours »
 - Les mécanismes d'inférence qu'elles offrent

Concepts complexes (1)

- constructeurs ensemblistes

- $C \cap D \xrightarrow{\mathfrak{S}} C^{\mathfrak{S}} \cap D^{\mathfrak{S}}$

- $C \sqcup D \xrightarrow{\mathfrak{S}} C^{\mathfrak{S}} \cup D^{\mathfrak{S}}$

- $\neg C \xrightarrow{\mathfrak{S}} \Delta^{\mathfrak{S}} \setminus C^{\mathfrak{S}}$

- quantificateurs

- $\exists R.C \xrightarrow{\mathfrak{S}} \{x \mid \exists (x,y) \in R^{\mathfrak{S}}, y \in C^{\mathfrak{S}}\}$

- $\forall R.C \xrightarrow{\mathfrak{S}} \{x \mid \forall (x,y) \in R^{\mathfrak{S}}, y \in C^{\mathfrak{S}}\}$

- restrictions de cardinalité

- $\leq n R \xrightarrow{\mathfrak{S}} \{x \mid \text{card}\{(x,y) \in R^{\mathfrak{S}}\} \leq n\}$

- $\geq n R \xrightarrow{\mathfrak{S}} \{x \mid \text{card}\{(x,y) \in R^{\mathfrak{S}}\} \geq n\}$

- $= n R \xrightarrow{\mathfrak{S}} \{x \mid \text{card}\{(x,y) \in R^{\mathfrak{S}}\} = n\}$

Concepts complexes (2)

- restrictions de cardinalité qualifiées

$$- \leq n \text{ R.C } \xrightarrow{\mathfrak{J}} \{ x \mid \text{card}\{(x,y) \in R^{\mathfrak{J}} \text{ et } y \in C^{\mathfrak{J}}\} \leq n \}$$

$$- \geq n \text{ R.C } \xrightarrow{\mathfrak{J}} \{ x \mid \text{card}\{(x,y) \in R^{\mathfrak{J}} \text{ et } y \in C^{\mathfrak{J}}\} \geq n \}$$

$$- = n \text{ R.C } \xrightarrow{\mathfrak{J}} \{ x \mid \text{card}\{(x,y) \in R^{\mathfrak{J}} \text{ et } y \in C^{\mathfrak{J}}\} = n \}$$

- extension

$$- \{ i, j, k, \dots \} \xrightarrow{\mathfrak{J}} \{ i^{\mathfrak{J}}, j^{\mathfrak{J}}, k^{\mathfrak{J}}, \dots \}$$

Exemples de concepts complexes

- \neg (Étudiant \sqcup Salarié)
- Groupe $\sqcap \exists$ membre . Étudiant
 - Peut contenir un *groupe sans aucun membre* ?
- Groupe $\sqcap \forall$ membre . Étudiant
 - Peut contenir un *groupe sans aucun membre* ?
- Groupe $\sqcap (\geq 10$ membre)
- (= 1 auteur)
- (= 1 auteur.Étudiant)
- { john, paul, george, ringo }

Rôles complexes (certaines LD)

- *constructeurs ensemblistes*

- $R \sqcap S \xrightarrow{\exists} R^{\exists} \cap S^{\exists}$

- $R \sqcup S \xrightarrow{\exists} R^{\exists} \cup S^{\exists}$

- $\neg R \xrightarrow{\exists} \Delta^{\exists} \times \Delta^{\exists} \setminus R^{\exists}$

- *composition*

- $R \circ S \xrightarrow{\exists} \{ (x,y) \mid \exists z, (x,z) \in R^{\exists} \text{ et } (z,y) \in S^{\exists} \}$

- *inverse*

- $R^{-} \xrightarrow{\exists} \{ (x,y) \mid (y,z) \in R^{\exists} \}$

- *fermeture transitive*

- $R^* \xrightarrow{\exists} \{ (x,y) \mid (x,y) \in (R^{\exists})^* \}$

Exemples de rôles complexes

- père \sqcup mère
- collègue \sqcap ami
- père^omère
- parent -
- parent *

Axiomes

- Définition de concept atomique
 - $A \sqsubseteq C \xrightarrow{\mathfrak{S}} A\mathfrak{S} \subseteq C\mathfrak{S}$ (subsumption, CN)
 - Etudiant \sqsubseteq Personne
 - $A \equiv C \xrightarrow{\mathfrak{S}} A\mathfrak{S} = C\mathfrak{S}$ (équivalence, CNS)
 - Etudiant \equiv Student
- General Inclusion Axiom ($C \sqsubseteq D, C \equiv D$)
 - entre concept complexes, limitations
 - Etudiant \sqsubseteq Personne $\sqcap \forall$ statut . Majeur

Autres axiomes

- Hiérarchie de rôles ($R \sqsubseteq S$, $R \equiv S$)
 - auteur \sqsubseteq contributeur
- Rôle transitif, rôle réflexif
 - ancêtre : transitif
- ...

A-Box

- Dans les LD qui supportent les individus :
- T-Box (axiomes sur les classes et les rôles)
 - niveau terminologique
- A-Box (axiomes sur les individus)
 - niveau assertion
 - $i:C \xrightarrow{\mathfrak{S}} i\mathfrak{S} \in C\mathfrak{S}$
 - $i,j:R \xrightarrow{\mathfrak{S}} (i\mathfrak{S}, j\mathfrak{S}) \subseteq R\mathfrak{S}$
 - $i = j \xrightarrow{\mathfrak{S}} i\mathfrak{S} = j\mathfrak{S}$
 - $i \neq j \xrightarrow{\mathfrak{S}} i\mathfrak{S} \neq j$
- NB : pas de distinction dans les LD qui supportent l'extension
 - $i:C \Leftrightarrow \{i\} \sqsubseteq C$, $i,j:R \Leftrightarrow \{i\} \sqsubseteq \exists R.\{j\}$, etc.

Inférences dans les LD

- Inférences de base
 - satisfiabilité : $\exists \mathfrak{S}, C^{\mathfrak{S}} \neq \emptyset$
 - un concept peut-il avoir des instances ?
 - subsomption : $\forall \mathfrak{S}, C^{\mathfrak{S}} \subseteq D^{\mathfrak{S}}$
 - un concept en subsume-t'il un autre ?
 - classification
 - où placer un concept dans la hiérarchie ?
- Inférence sur les individus
 - identification ou test à l'instanciation
 - quel est le concept le plus spécifique décrivant un individu ?
- Raisonnement sur une base de connaissance
 - *T-Box seule* ou *T-Box + A-Box*

Inférence dans les LD

- On peut tout ramener à un problème de satisfiabilité, donc au test d'existence d'un modèle pour une expression
 - C est subsumé par $D \Leftrightarrow C \sqcap \neg D$ est insatisfiable
 - C et D sont équivalents $\Leftrightarrow C \sqcap \neg D$ et $D \sqcap \neg C$ sont *insatisfiables*
 - C et D sont disjoints $\Leftrightarrow C \sqcap D$ est insatisfiable
 - a est une instance de $C \Leftrightarrow ABox \cup \{a : \neg C\}$ est insatisfiable
- La complexité dépend de la logique de description choisie, qui dépend des constructeurs utilisés
 - <http://www.cs.man.ac.uk/~ezolin/dl/>

LD – Méthode des tableaux

- Un tableau est une représentation d'un ensemble de modèles
 - arbre fini étiqueté
 - chaque branche mémorise une série d'évaluations possibles pour les énoncés testés
- L'application de règles de transformation (adaptation de la méthode des tableaux en logique des propositions) garantit qu'on explorera tous les modèles possibles
- Principe : raisonnement par réfutation
 - on suppose qu'une instance x existe
 - on déduit tout ce qu'on peut sur cette instance
 - si on arrive à une contradiction ou clash : $x^{\mathfrak{S}} \in C^{\mathfrak{S}} \cap \neg C^{\mathfrak{S}}$, il n'y a pas de modèle
 - si on arrive à une branche complète : on a un modèle

LD – Méthode des tableaux (1)

- Soit une T-box

Homme $\sqsubseteq \neg$ Femme

Personne $\sqsubseteq \exists$ père Homme \sqcap

\exists mère Femme \sqcap

\forall père Homme \sqcap

parent ≤ 2

père \sqsubseteq parent

mère \sqsubseteq parent

- Question

Personne \sqcap *père* ≥ 2 est-il satisfiable ?

LD – Méthode des tableaux (2)

Personne, père ≥ 2

(x)

- Homme $\sqsubseteq \neg$ Femme
- **Personne \sqsubseteq**
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- **Personne \sqcap père ≥ 2**
est il satisfiable ?

LD – Méthode des tableaux (3)

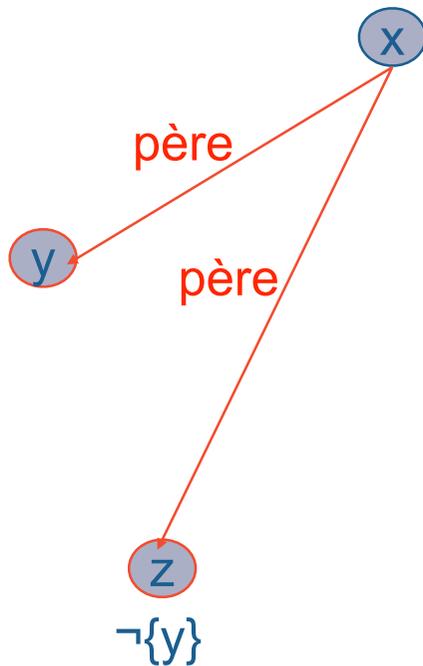
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme, \forall
père Homme, parent ≤ 2

X

- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
est il satisfiable ?

LD – Méthode des tableaux (4)

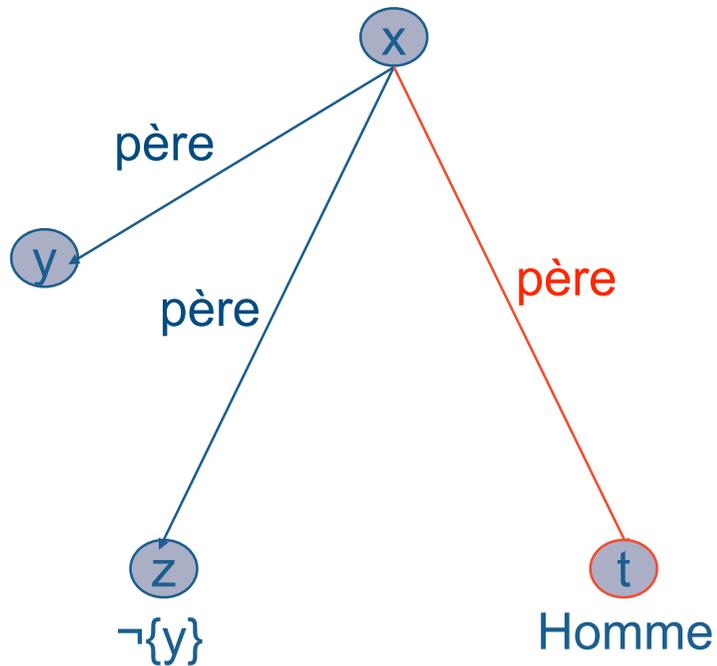
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme, \forall
père Homme, parent ≤ 2



- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
est il satisfiable ?

LD – Méthode des tableaux (5)

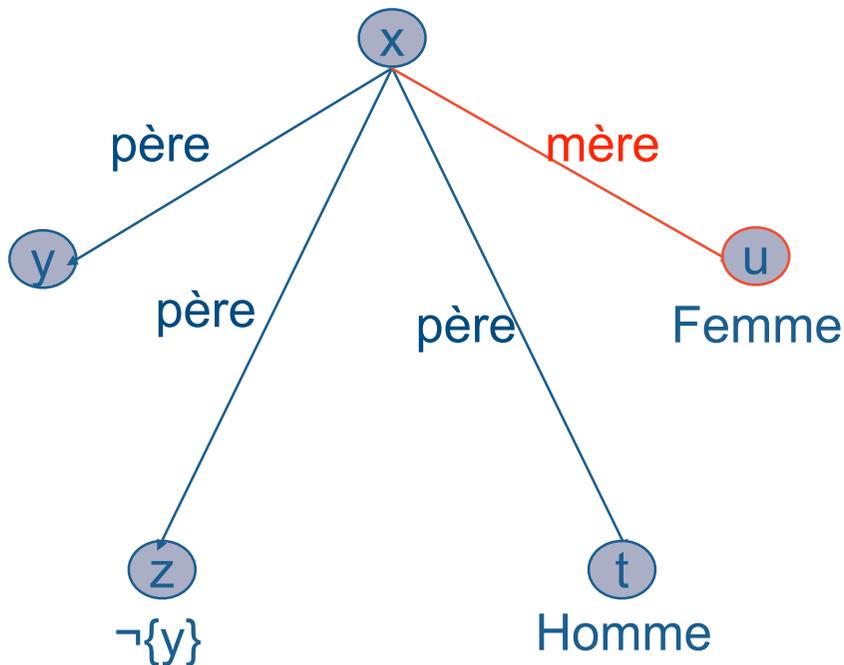
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme,
 \forall père Homme, parent ≤ 2



- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
est-il satisfiable ?

LD – Méthode des tableaux (6)

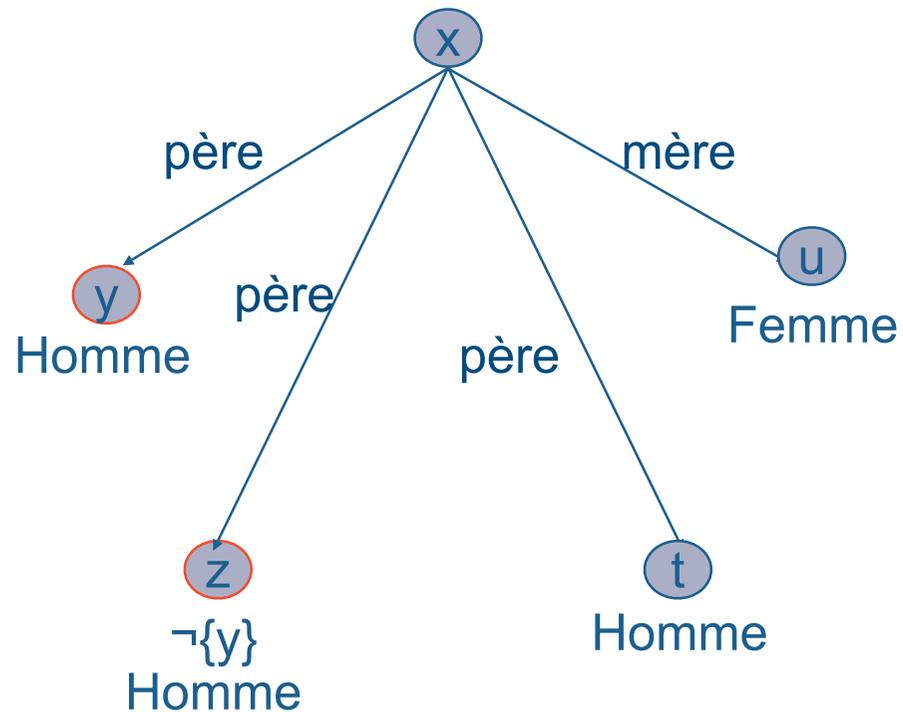
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme, \forall
 père Homme, parent ≤ 2



- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
 parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
 est il satisfiable ?

LD – Méthode des tableaux (7)

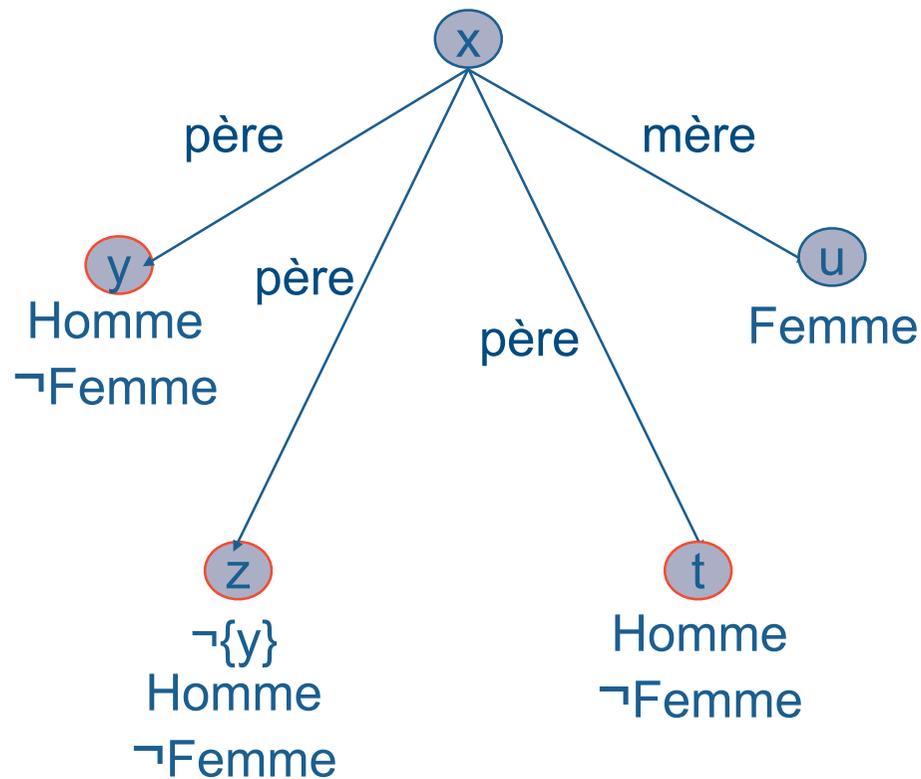
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme, \forall
 père Homme, parent ≤ 2



- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
 parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
 est il satisfiable ?

LD – Méthode des tableaux (8)

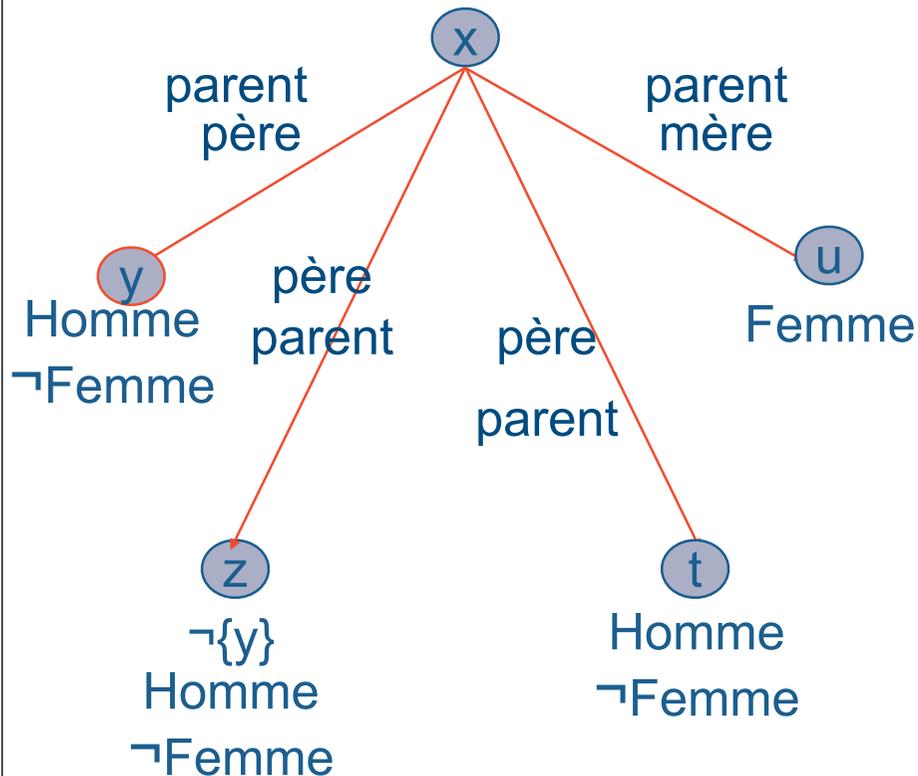
Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme, \forall
 père Homme, parent ≤ 2



- Homme \sqsubseteq ¬ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
 parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
 est il satisfiable ?

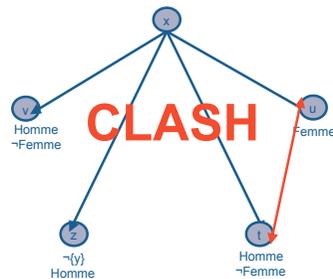
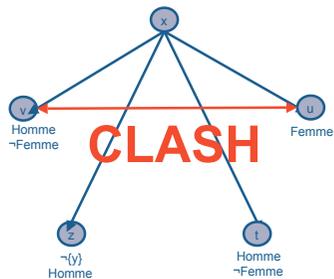
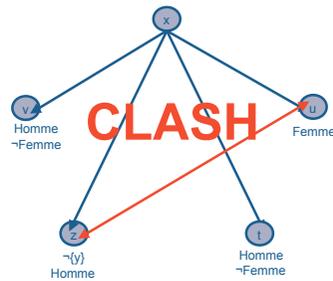
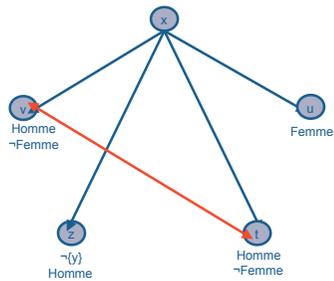
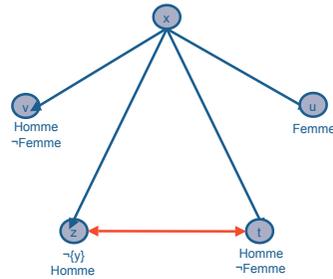
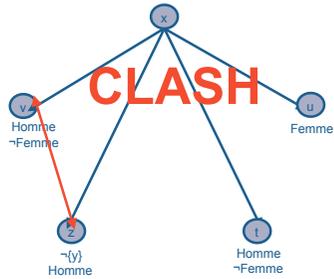
LD – Méthode des tableaux (9)

Personne, père ≥ 2 ,
 \exists père Homme, \exists mère Femme,
 \forall père Homme, parent ≤ 2



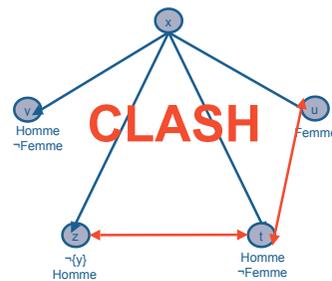
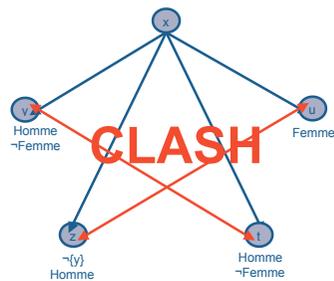
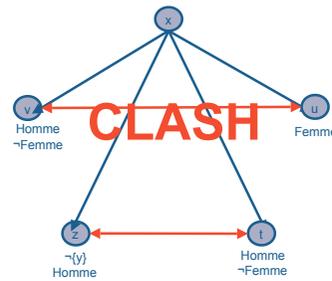
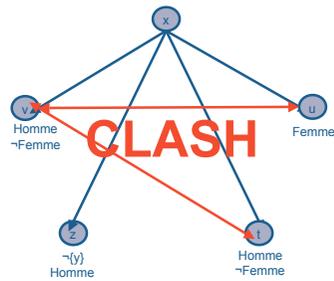
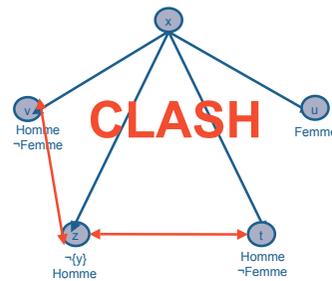
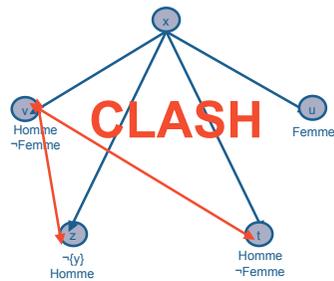
- Homme $\sqsubseteq \neg$ Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
 parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
 est il satisfiable ?

LD – Méthode des tableaux (10)



- Homme \sqsubseteq \neg Femme
- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2
- père \sqsubseteq parent
- mère \sqsubseteq parent
- Personne \sqcap père ≥ 2
est il satisfiable ?

LD – Méthode des tableaux (II)



- Homme \sqsubseteq \neg Femme

- Personne \sqsubseteq
 \exists père Homme \sqcap
 \exists mère Femme \sqcap
 \forall père Homme \sqcap
parent ≤ 2

- père \sqsubseteq parent

- mère \sqsubseteq parent

- Personne \sqcap père ≥ 2
n'est pas satisfiable

Familles de logiques de description

- Différentes familles se distinguent par les opérateurs qu'elles offrent
- Les opérateurs sont indiqués par des lettres
 - S : LD basique (ALC) plus rôles transitifs (e.g., ancestor R+)
 - H : hiérarchie de rôles (e.g., hasDaughter v hasChild)
 - O : description extentionnelle des classes (e.g., {Italy, France})
 - I : rôles inverses (e.g., isChildOf \equiv hasChild–)
 - N : restriction de nombre (e.g., >2 hasChild)
- DL de base + hiérarchie de rôles + nominaux + inverse + NR = SHOIN
 - base de OWL-DL
 - SHOIN est très expressive, et encore décidable
 - possibilité de définir des raisonneurs efficaces

Raisonneurs sur les LD

- Cerebra Engine
 - is a commercial C++-based reasoner. It implements a tableau-based decision procedure for general TBoxes (subsumption, satisfiability, classification) and ABoxes (retrieval, tree-conjunctive query answering using a XQuery-like syntax). It supports the OWL-API and comes with numerous other features.
- FaCT++
 - is a free (GPL/LGPL) open-source C++-based reasoner for SROIQ with simple datatypes (i.e., for OWL 2). It implements a tableau-based decision procedure for general TBoxes (subsumption, satisfiability, classification) and ABoxes (retrieval). It supports the OWL-API, the lisp-API and the DIG interface.
- KAON2
 - is a free (free for non-commercial usage) Java reasoner for SHIQ extended with the DL-safe fragment of SWRL. It implements a resolution-based decision procedure for general TBoxes (subsumption, satisfiability, classification) and ABoxes (retrieval, conjunctive query answering). It comes with its own, Java-based interface, and supports the DIG interface.

Raisonneurs sur les LD

- Pellet
 - is a free open-source Java-based reasoner for SROIQ with simple datatypes (i.e., for OWL 1.1). It implements a tableau-based decision procedure for general TBoxes (subsumption, satisfiability, classification) and ABoxes (retrieval, conjunctive query answering). It supports the OWL-API, the DIG interface, and the Jena interface and comes with numerous other features.
- RacerPro
 - is a commercial (free trials and research licenses are available) lisp-based reasoner for SHIQ with simple datatypes (i.e., for OWL-DL with qualified number restrictions, but without nominals). It implements a tableau-based decision procedure for general TBoxes (subsumption, satisfiability, classification) and ABoxes (retrieval, nRQL query answering). It supports the OWL-API and the DIG interface and comes with numerous other features.
- Liste à jour sur <http://www.cs.man.ac.uk/~sattler/reasoners.html>

DIG

- DL Implementation Group
- <http://dig.cs.manchester.ac.uk/>
- The DIG Interface provides a specification of an interface for description logic reasoners. It is intended to be a lightweight mechanism providing access to reasoning functionality. There are many things that a reasoning service may be expected to provide that the DIG interface does not provide — it is expected that a DIG reasoner will be one component within a larger architecture.

Plan

- Ingénierie des connaissances
- Ontologies
- Logiques de description
- **Introduction au Web sémantique**
- OWL
- Conception d'ontologies

Web humain / web machine

- "Le Web sémantique est une vision du Web à venir, où les informations reçoivent une signification facilitant le traitement et l'intégration automatiques par les machines des informations disponibles sur le Web"
- « L'expression Web sémantique, due à Tim Berners-Lee au sein du W3C, fait d'abord référence à la vision du Web de demain comme un vaste espace d'échange de ressources entre êtres humains et machines permettant une exploitation, qualitativement supérieure, de grands volumes d'informations et de services variés. Espace virtuel, il devrait voir, à la différence du Web que nous connaissons aujourd'hui, les utilisateurs déchargés d'une bonne partie de leurs tâches de recherche, de construction et de combinaison des résultats, grâce aux capacités accrues des machines à accéder aux contenus des ressources et à effectuer des raisonnements sur ceux-ci. »
- « Le Web actuel est essentiellement syntaxique, dans le sens que la structure des documents (ou ressources au sens large) est bien définie, mais que son contenu reste quasi inaccessible aux traitements machines. Seuls les humains peuvent interpréter leurs contenus. La nouvelle génération de Web – Le Web sémantique – a pour ambition de lever cette difficulté. Les ressources du Web seront plus aisément accessibles aussi bien par l'homme que par la machine, grâce à la représentation sémantique de leurs contenus. »

Le Web aujourd'hui

- Universalité
 - Homogénéité des techniques utilisées
 - HTTP, HTML, intranet
 - Puissance de l'hypertexte
 - toute ressource peut être liée à toute autre ressource
- Ressources du Web
 - Principalement constituées pour une exploitation humaine
 - Ecrites par des humains ou générées par des machines
- Nécessité de médiateurs
 - Moteurs de recherche plus ou moins spécialisés, plus ou moins efficaces
- Mais
 - trouver les bonnes ressources, analyser le contenu des pages, dégager les bonnes informations, combiner les différents résultats est coûteux en temps... et parfois fastidieux

Agents logiciels sur le Web

- Les moteurs de recherche par mots clés laissent beaucoup de travail à l'utilisateur
- Peu de descriptions structurées des informations (méta-données) utilisables et utilisées par des agents logiciels
- Problème
 - HTML ne donne que des pages et des liens
 - dont les étiquettes sont interprétables seulement par leurs utilisateurs
 - sans sémantique utilisable par les machines
- Conclusion
 - l'information et les services sur le Web sont aujourd'hui peu exploitables par des machines,
 - mais peut-être de moins en moins exploitables sans l'aide des machines !

Exemples de requêtes complexes

- Trouver des résumés d'articles écrits par des chercheurs que connaît YP, après 2004 portant sur le Web sémantique (ou les extraits intéressants signalés par d'autres chercheurs)
- Trouver des entreprises du secteur informatique faisant des pertes, avec un chiffre d'affaire entre ... et ... et ayant eu des contrats avec d'autres entreprises logicielles dans les derniers 90 jours
- Trouver des locations de vacances familiales dans telle région avec des activités de plein-air, quelques musées et églises touristiques et un festival de jazz dans la période du au...

Le Web de demain (W3C, 2000)

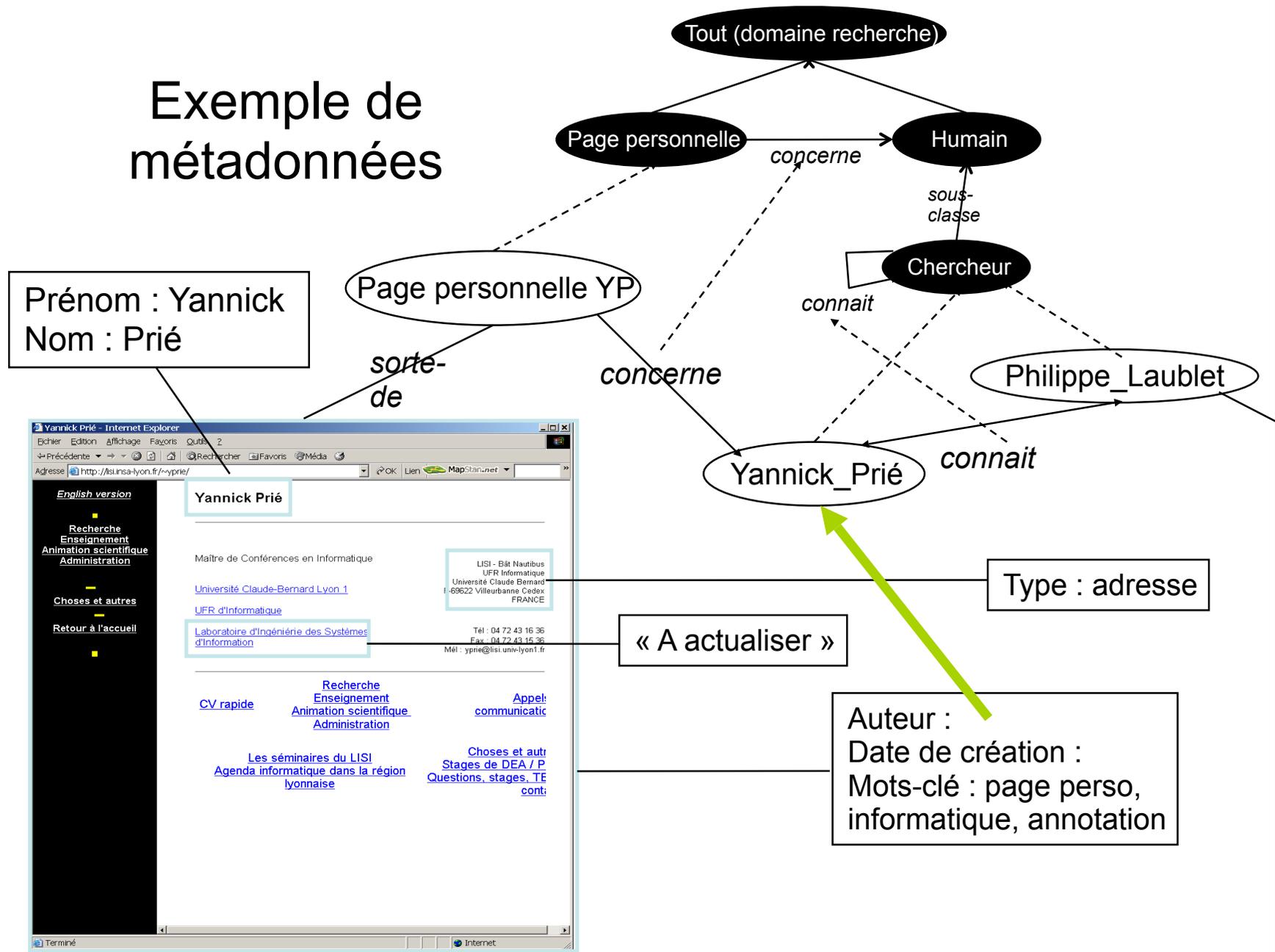
- « Un vaste espace d'échanges de ressources entre machines permettant l'exploitation de grands volumes d'informations et de services variés, aidant les utilisateurs en les libérant d'une (bonne) partie de leur travail de recherche, et de combinaison de ces ressources »
- « Le Web sémantique n'est pas un Web séparé, mais une extension du Web actuel dans lequel l'information est munie d'une signification bien définie permettant aux ordinateurs et aux personnes de mieux travailler en coopération » (T. Berners-Lee , J. Hendler et O.Lassila, Scientific American 2001)
- Les ressources sont interprétées et utilisées au moins autant par des logiciels que par des utilisateurs humains au service des utilisateurs
- Deux approches aujourd'hui
 - ressources = documents
 - le WS apporte les métadonnées sémantiques
 - ressources = données distribuées
 - le WS apporte la possibilité de tout faire interopérer

Approche 1

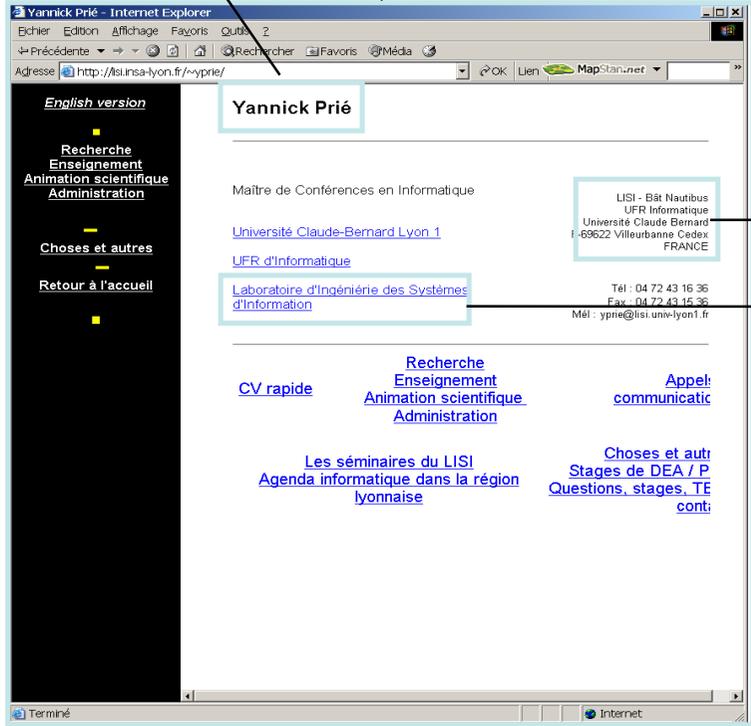
annotations et documents

- Métadonnées / annotations
 - « information associée à une ressource du web, permettant d'en favoriser l'utilisation par un agent humain, du fait de son exploitation par un agent logiciel »
- Points importants
 - Séparation de la ressource et de l'annotation
 - Information structurée descriptive de la ressource
 - Traitements plus ou moins évolués par des programmes
 - Utilisateur plus ou moins conscient de l'exploitation des métadonnées (de leur existence)

Exemple de métadonnées



Prénom : Yannick
Nom : Prié



« A actualiser »

Type : adresse

Auteur :
Date de création :
Mots-clé : page perso,
informatique, annotation

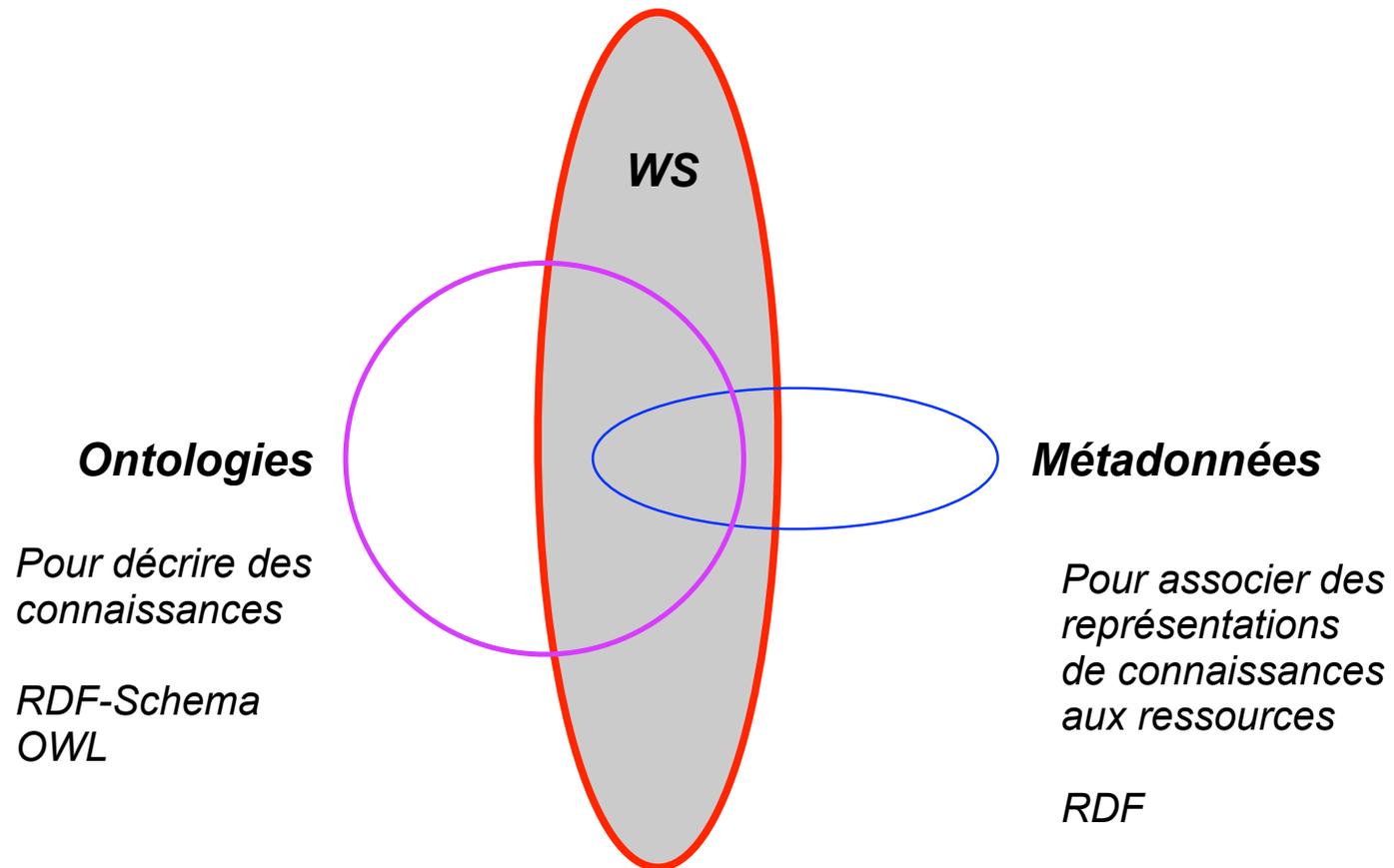
Ressources sémantiques pour

- Décrire des métadonnées documentaires
 - Conditions de productions, ... qui, quand, où, comment
 - Conditions d'utilisation
- Décrire des documents (métadonnées, annotations ...)
 - Cela parle de quoi
 - C'est structuré comment
 - Cela fait référence à quoi ...
 - ...
- Collaborer entre communautés (de travail, relationnelles,)
 - Notes de travail
 - Commentaires
 - Notes de « réputation »
- Structurer des ressources (liens hypertextes étiquetés) ... avec un certain niveau de formalisation

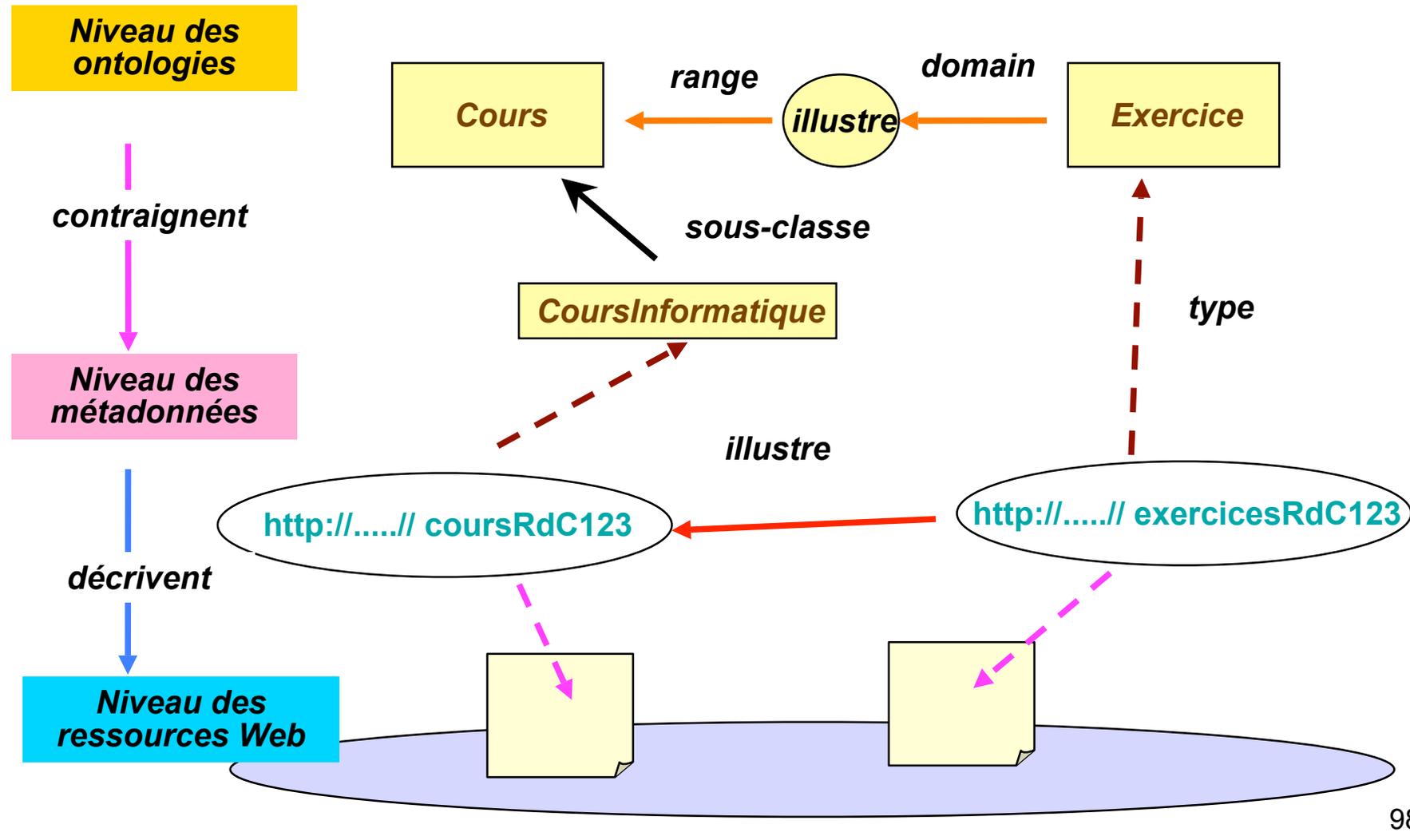
L'approche initiale du Web sémantique

- Mise en place d'une couche « sémantique »
 - permettant la description des ressources et des contenus
 - et la structuration des ressources (liens entre ressources étiquetés) avec un certain niveau de formalisation
 - utilisables par des agents logiciels
- De tels logiciels peuvent utiliser
 - les réseaux de concepts et de relations présentes dans ces descriptions
 - combiner ces significations
- Ces descriptions associées à une ressource lors de sa création ou plus tard
 - Et codées dans des langages standards dédiés
 - RDF, RDF(S), OWL

Web sémantique



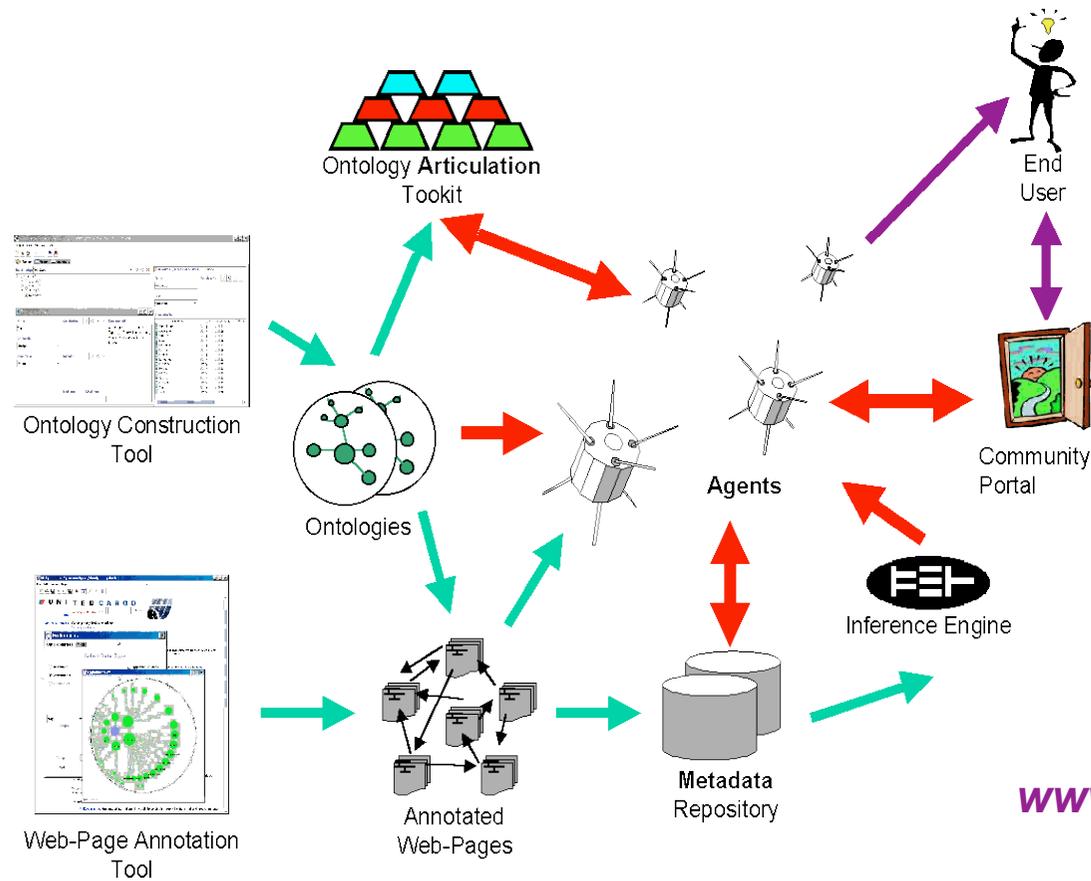
Ontologies, métadonnées et ressources



Familles d'application

- Recherche de ressources basée sur les structures relationnelles entre concepts
 - versus recherche basée sur des mots-clés
- Navigation sémantique
 - versus navigation basée sur des liens “anonymes” pour les logiciels
- Génération sémantique de ressources, personnalisation
 - versus une seule “taille” pour tout le monde
- Question - réponse en combinant des résultats
 - versus simple recherche de document
- ...

Ontologies, métadonnées, agents logiciels et ... utilisateur



Approche 2

intégration de données

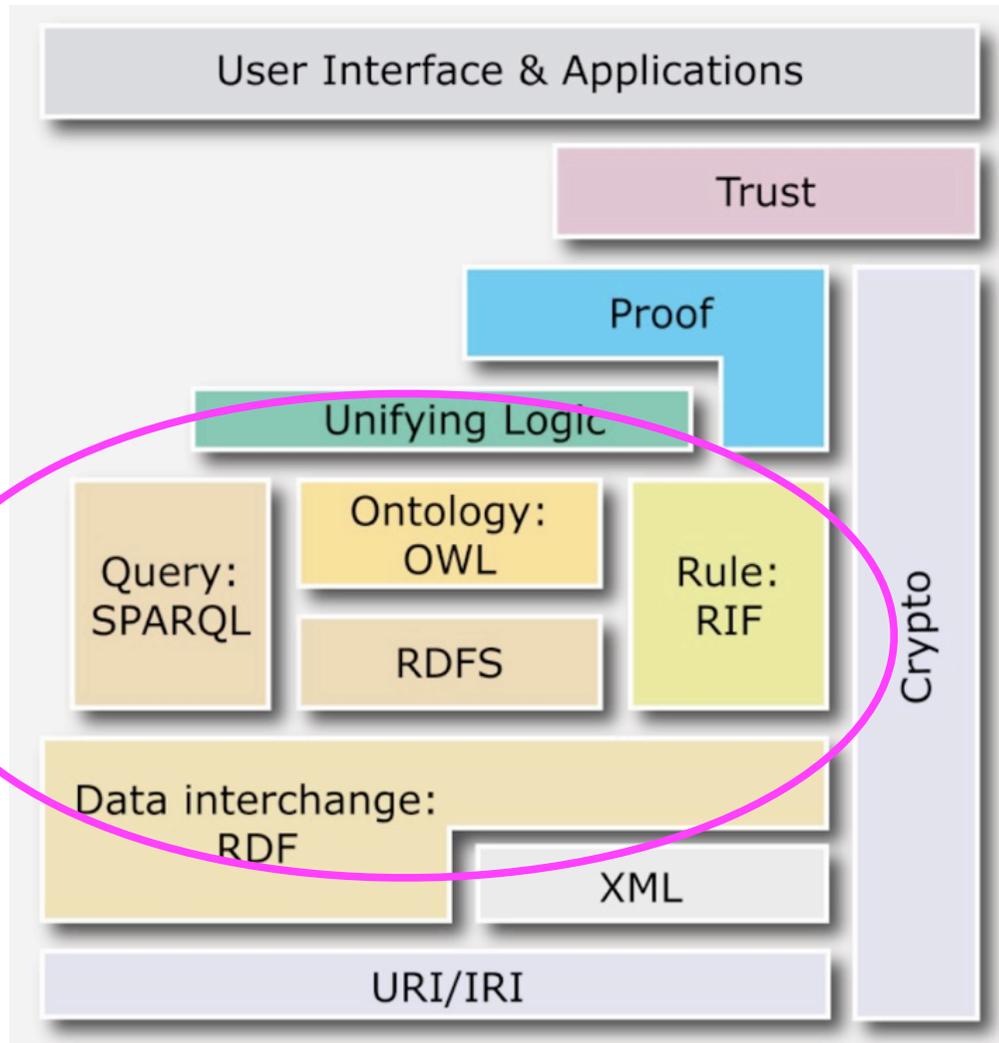
- What is the Semantic Web? in 17th XBRL International Conference, Eindhoven, the Netherlands 5st May, 2008
- par Ivan Herman, W3C

Enfin, le WS c'est / ça sera

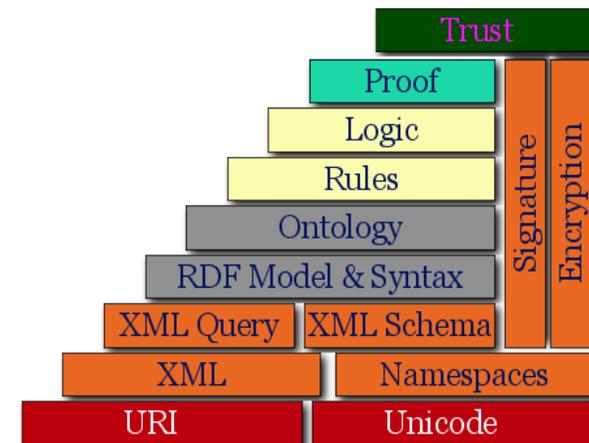
(Ivan Herman)

- Un moyen standard de spécifier des données et leurs relations
- Un moyen d'étendre les principes du Web et des documents aux données : créer un Web de données
- Un moyen de découvrir de nouvelles relations par le biais du Web (tout comme pour les documents)

La pile des langages (W3C)



Un gâteau tout chaud



Le vieux gâteau

Les « couches » du WS (1)

- XML / RDF : couche de transport syntaxique
 - transformation, échange, stockage, import/export
- RDF : réseau sémantique de base
 - triplets relationnels...
 - modèle de connaissances : description, intégration,
- RDFS : langage de classes, simple
 - représentation de structures conceptuelles et d'ontologies simples avec description hiérarchique des concepts et des propriétés
- OWL : langage de représentation d'ontologies plus complexes
 - raisonnement en plus

Les « couches » du WS (2)

- SPARQL
 - langage d'interrogation de données RDF
- RIF
 - langage d'expression de règles
- Unifying logic
 - Complète et unifie
 - Connaissance déclarative spécifique à l'application
- Plus prospectif
 - Preuve
 - Génération de preuves
 - Validation
 - Confiance
 - Signatures numériques
 - recommandations, organismes d'évaluation et de certification

Normes du web sémantique

RDF

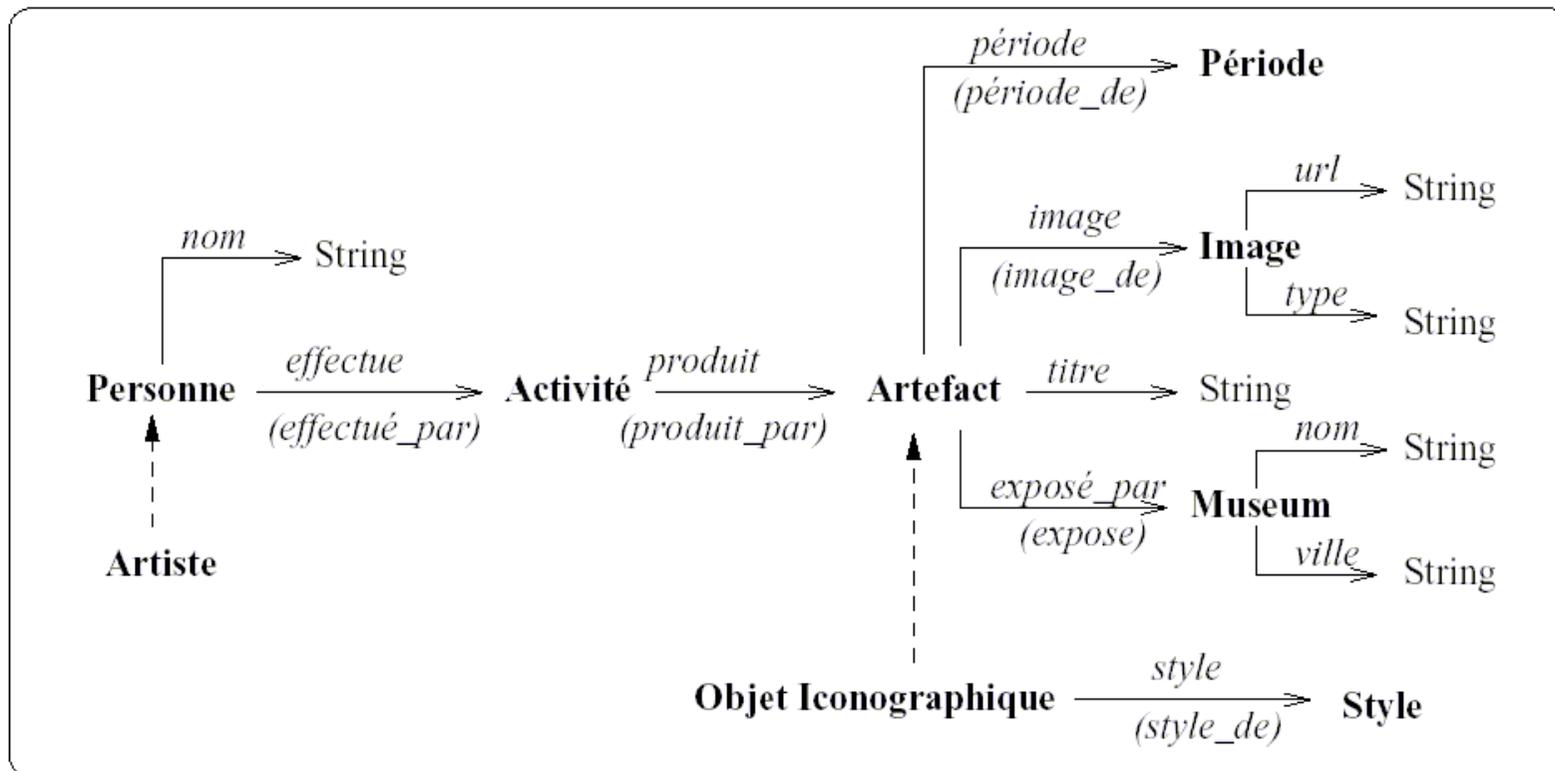
- Les données / connaissances sont exprimées
 - Sous forme de triplets
 - Comme le sujet, verbe et objet d'une phrase élémentaire
- Les assertions sont faites sur une ressource Web (URI)
 - Elle a des propriétés avec certaines valeurs : une valeur de base ou une autre ressource Web
- Aspect relationnel premier
 - Possibilité d'ajouter des triplets descriptifs sans difficulté
 - Des représentations plus complexes sont proposées

Normes du Web sémantique

RDFS

- RDF Schema = extension de RDF
- Langage de description des classes ou des types des éléments de RDF
- Décrit des classes et les types de propriétés autorisées entre les instances de classes (en RDF)
- Fournit des hiérarchies de généralisation de propriétés ou de classes

Modèle conceptuel simple exprimable en RDFs



D'après Bernd Amman

Normes du Web sémantique

OWL

- Ontology Web Language
- Pour décrire des ontologies
- "à côté de RDF-Schema"
- Basé sur les logiques de description
 - trois logiques de description
 - OWL-DL, OWL-Lite, OWL-Full

Composants typiques d'une ontologie RDFS ou OWL

- Concepts ou classes
 - *Enseignant, Cours, CoursInformatique*
 - Relations de généralisation / spécialisation entre concepts
- Relations entre concepts ou « propriétés objets »
 - *X enseigne Y*
 - ObjectProperties en OWL
- Les attributs des concepts ou « propriétés types de données »
 - *X a comme e-mail « philippe.lauble@paris4.sorbonne.fr »*
 - DataTypesProperties en OWL

Composants typiques d'une ontologie OWL

- Contraintes diverses ou axiomes restreignant l'interprétation des éléments
 - *Seuls les membres de l'université peuvent enseigner des cours*
 - *Professeur et Etudiant sont disjoints*
 - *Chaque département doit avoir au moins 10 enseignants*
- Construction de classes complexes
 - Utilisation de connecteurs logiques
 - *EtablissementEnseignement et PoleDeRecherche*
 - Définitions ...
 - *Université dans un grande ville avec au moins un cours de linguistique*
 - *Ces constructions sont utilisées par les mécanismes inférentiels*

Normes du web sémantique

SPARQL, GRDDL

- SPARQL
 - Simple Protocol and RDF Query Language
 - interrogation à base de motifs de graphes
 - un protocole pour les requêtes
 - un format XML de résultat
- GRDDL
 - Gleaning Resource Descriptions from Dialects of Languages
 - accès RDF à des données (X)HTML

Outils du WS

- Système de gestion de bases de triplets RDF
- Moissonneurs RDF
- Moteurs d'inférence
- Convertisseurs
- Moteurs de recherche
- Middleware
- CMS
- Navigateurs web sémantique
- Environnements de développement
- Outils de création d'ontologies RDFS, OWL
- Outils d'annotation automatique ou semi-automatique
- Wikis sémantiques

Navigateur sémantique E-Culture MultimediaN

- <http://e-culture.multimedien.nl/demo/search>

E-Culture MultimediaN *cultural heritage search*

search
annotate
Q search

This cultural search engine will give you access to artworks from several museum collections.
Type a keyword, for example: *Derain*, *calligraphy*, or *1867*.

SEARCH



Artchive.com (>3,000 objects)



Rijksmuseum.nl (>16,000 objects)



Volkenkunde.nl (>10,000 objects)



Tropenmuseum.nl (>78,000 objects)



Bibliopolis.nl (>1,600 objects)

Vocabularies and thesauri



Getty AAT (>31.000)



Getty ULAN (>130.000)



Getty TGN (>890.000)



SVCN (Dutch ethnology, >11.000)



Princeton Wordnet (>115.000)

RP-P-1964-479
<http://e-culture.multimedien.nl/hs/rijksmuseum#RP-P-1964-479>



Links

- permanent link
- annotate

De Heilige Hiëronymus zit in een rotsachtige omgeving te bidden bij een crucifix. Naast hem staat een knotwilg, aan zijn voeten ligt een leeuw.

Property	Value
id	33165
engraver	Dürer, Albrecht
physical material	droge naald
pm	Dürer, Albrecht
prentmaker	Dürer, Albrecht
subject collection	prenten
subject iconclass	St. Jerome as penitent in the desert, half naked, kneeling before a crucifix and holding a stone in his hand to beat his breast; a skull (and other 'Vanitas' symbols), sometimes a scorpion and a snake beside him
after	1512
before	1512
earliest date	1512
latest date	1512
inscription	F...
inscription method	gestempeld
Current Repository	Rijksmuseum, Amsterdam
Material	papier
Source	Rijksmuseum Amsterdam
Style/Period	16e eeuw, eerste kwart 16e eeuw
Technique	2219
Type	91
type	Work

used as metadata in:

Property	Subject
relation.depicts	M-RP-P-1964-479-00

SKOS

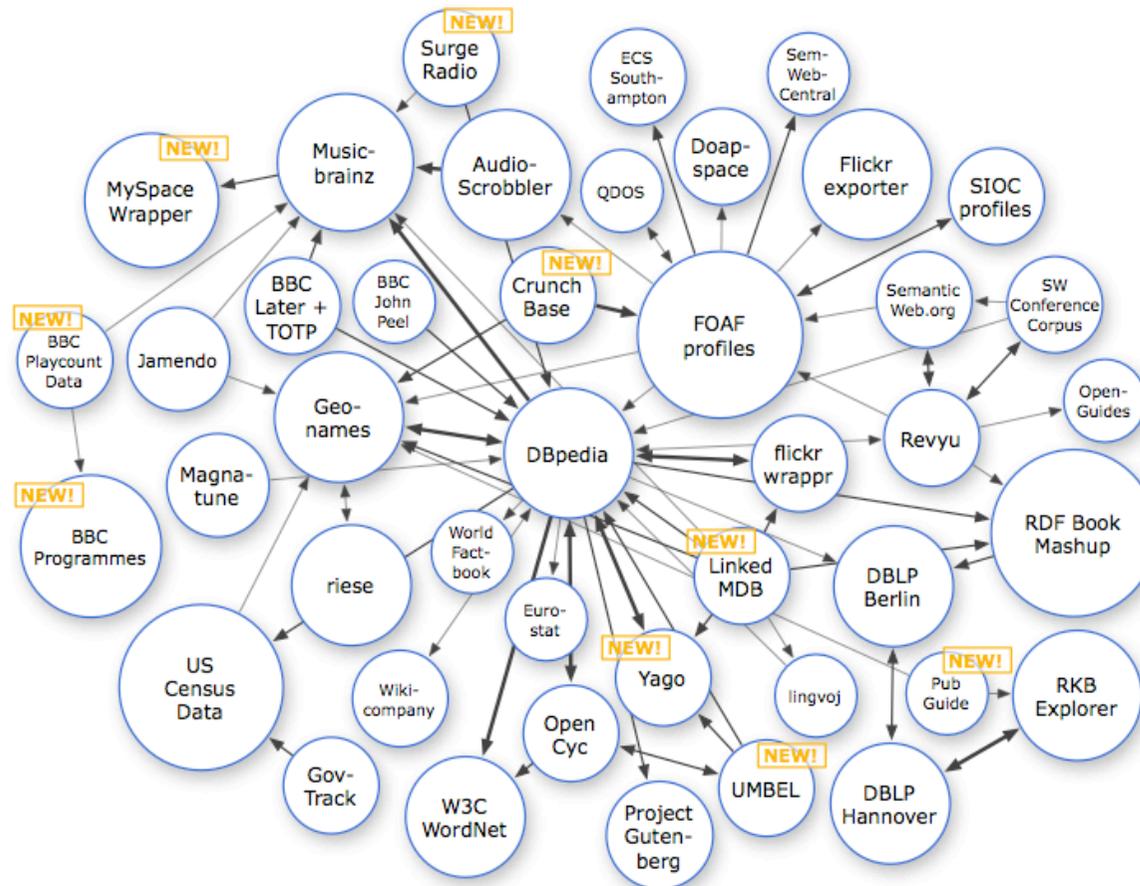
- Simple Knowledge Organisation Systems
- Vocabulaire RDF/RDFS pour exprimer des vocabulaires contrôlés structurés utilisés dans les bibliothèques et les sciences de l'information en général
 - thésaurus, taxonomies, vedettes-matières, etc.
- En cours de standardisation au W3C
- Nombres de *thesaurii* en cours de traduction

Exemple DBPedia

- DBpedia is a community effort to extract structured information from Wikipedia and to make this information available on the Web. DBpedia allows you to make sophisticated queries against Wikipedia, and to link other data sets on the Web to Wikipedia data.
- Des triplets RDF extraits de wikipedia
 - plusieurs dizaines de millions
- Des API d'accès et de recherche
- Des outils de recherche
 - "People influenced by Friedrich Nietzsche"
 - <http://wikipedia.aksw.org/index.php?qid=14>
 - "People who were born in Berlin before 1900"
 - [http://dbpedia.org/snorql/?query=SELECT+%3Fname+%3Fbirth+%3Fdeath+%3Fperson+WHERE+{%0D%0A+++++%3Fperson+dbpedia2%3AbirthPlace+%3Chttp%3A%2F%2Fdbpedia.org%2Fresource%2FBerlin%3E+. %0D%0A+++++%3Fperson+dbpedia2%3Abirth+%3Fbirth+. %0D%0A+++++%3Fperson+foaf%3Aname+%3Fname+. %0D%0A+++++%3Fperson+dbpedia2%3Adeath+%3Fdeath%0D%0A+++++FILTER+\(%3Fbirth+%3C+%221900-01-01%22^^xsd%3Adate\)+. %0D%0A}%0D%0AORDER+BY+%3Fname](http://dbpedia.org/snorql/?query=SELECT+%3Fname+%3Fbirth+%3Fdeath+%3Fperson+WHERE+{%0D%0A+++++%3Fperson+dbpedia2%3AbirthPlace+%3Chttp%3A%2F%2Fdbpedia.org%2Fresource%2FBerlin%3E+. %0D%0A+++++%3Fperson+dbpedia2%3Abirth+%3Fbirth+. %0D%0A+++++%3Fperson+foaf%3Aname+%3Fname+. %0D%0A+++++%3Fperson+dbpedia2%3Adeath+%3Fdeath%0D%0A+++++FILTER+(%3Fbirth+%3C+%221900-01-01%22^^xsd%3Adate)+. %0D%0A}%0D%0AORDER+BY+%3Fname)
 - "Les sites inscrits au patrimoine mondial de l'Unesco"
 - <http://www.lespetitescases.net/semweblabs/dbpedia/dbpedia-categorie-ville.php>
 - etc.

Linking open data project

- <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>
- Exposer des données en RDF, les lier



As of September 2008

Cas d'utilisation du Web sémantique

<http://www.w3.org/2001/sw/sweo/public/UseCases/>

- Active Knowledge Management for Integrated Operations, Computas AS et Statoil
- B2B Integration with Semantic Mediation, BT Research and Venturing
- Drug Ontology Project for Elsevier (DOPE), Elsevier, Aduna, et Vrije Universiteit Amsterdam
- Enhancing Web Searches within the Principality of Asturias, CTIC Foundation, University of Oviedo, et Principality of Asturias
- Helping New Judges Answer Complex Legal Questions, Spanish General Council of the Judiciary
- How Ontologies and Rules Help to Advance Automobile Development, AUDI AG, Achievo Inproware GmbH, et ontoprise GmbH
- Mobile Content Recommendation System, Saltlux
- Ontology-Driven Information Integration and Delivery; A Survey of Semantic Web Technology in the Oil and Gas Industry, Chevron Information Technology Company
- Semantic Web Technologies in Automotive Repair and Diagnostic, Renault
- Using Semantic Web and Proof Technologies to Reduce Errors in Radiological Procedure Orders, Agfa Healthcare
- Using the Semantic Web for Internal Compound Repurposing, Pfizer Ltd.
- WEASEL, Vodafone R&D Corporate Semantic Web, Vodafone Group Research & Development

Etat du web sémantique

- Les langages variés
 - RDF, OWL, SPARQL, GRDDL, RDFa
- Des vocabulaires utilisés et réutilisés
 - DC, FOAF
- Des applications et des outils
- Une migration de la thématique de l'annotation sémantique à celle de l'interopérabilité de données
- SPARQL comme point d'unification

Travaux en cours au W3C

- Intégration règles et WS
 - Rule Interchange Format (RIF)
- OWL 2.0
- SKOS
- Web sémantique pour communautés précises
 - eGouvernement, santé, etc

Défis du WS pour la représentation des connaissances

- Convergence / standardisation / pragmatisme
 - Apport de « millions d'ingénieurs »
 - Solutions partielles déjà intéressantes
 - Tests instantanés
- Variété / richesse du web
 - Taille du web
 - de plus en plus de ressources RDF/OWL
 - Utilisation parfois de mécanismes inférentiels
 - Variété
 - Beaucoup d'auteurs
 - Qui fait « autorité » ? Questions de confiance
 - Beaucoup d'ontologies / vocabulaires, qualité variée
 - Comment les articuler ?
 - Utilisation imprévisible
 - Décentralisation et évolution permanente du web

Plan

- Ingénierie des connaissances
- Ontologies
- Logiques de description
- Introduction au Web sémantique
- **OWL**
- Conception d'ontologies

RDF

- Resource Description Framework
 - Recommandation du W3C
 - Description de métadonnées, proche des réseaux sémantiques
 - XML : unification de la syntaxe
RDF : unification de la sémantique
- Modèle
 - Termes du langage :
 - URI (≈URL)
 - littéraux
 - nœuds vierges (*blank*)
 - Structure d'un énoncé : sujet prédicat objet

RDF (triplet)

<http://liris.cnrs.fr/labo>

<http://xmlns.com/foaf/0.1/member>

<http://liris.cnrs.fr/yannick.prie/moi>

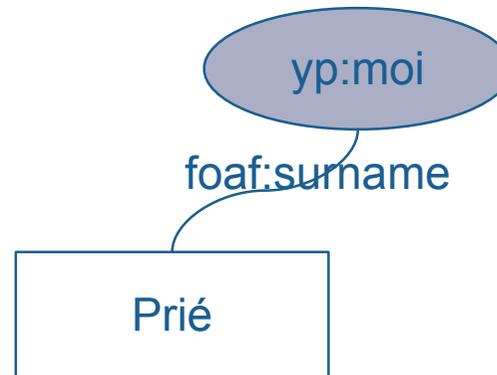


RDF (notation)



foaf: <http://xmlns.com/foaf/0.1/>
liris: <http://liris.cnrs.fr/>
yp: <http://liris.cnrs.fr/yannick.prie/>

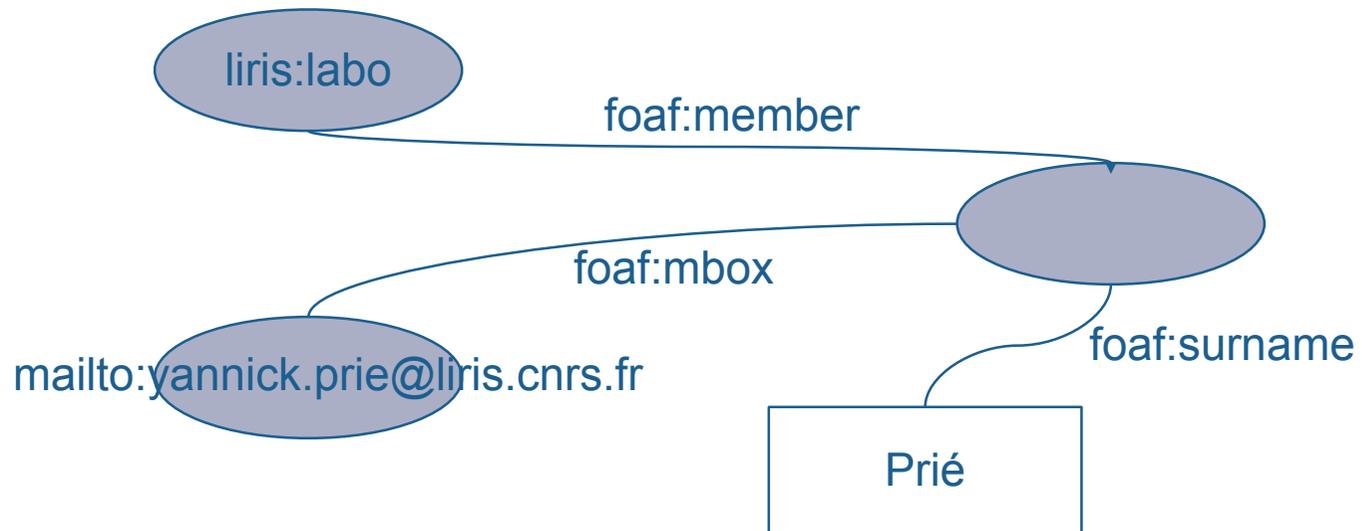
RDF (littéral)



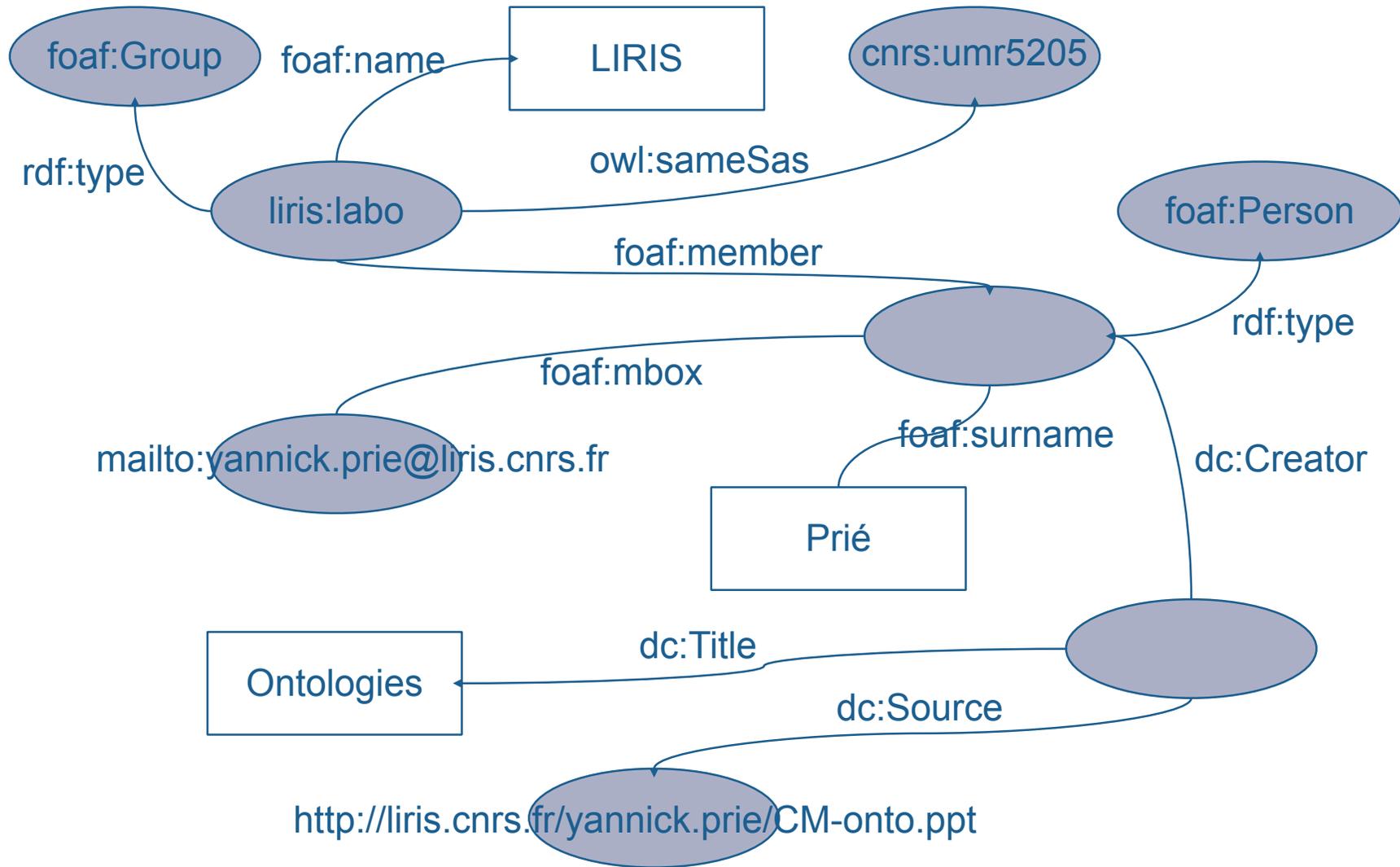
RDF (nœud vierge)



RDF (exemple)



RDF (exemple)



RDF (syntaxe RDF/XML)

```
<foaf:Group rdf:about="&liris;labo">  
  <foaf:name>LIRIS</foaf:name>  
  <owl:sameAs rdf:resource="&cnrs;umr5205" />  
  <foaf:member>  
    <foaf:Person rdf:nodeID="yp">  
      <foaf:surname>Prié</foaf:surname>  
    </foaf:Person>  
  </foaf:member>  
</foaf>
```

```
<rdf:Description>  
  <dc:Title>  
    Ontologies  
  </dc:Title>  
  <dc:Creator rdf:nodeId="yp" />  
  <dc:Source rdf:resource=  
    "http://liris.cnrs.fr/ens/CM-onto.ppt">
```

RDF (syntaxe Notation3)

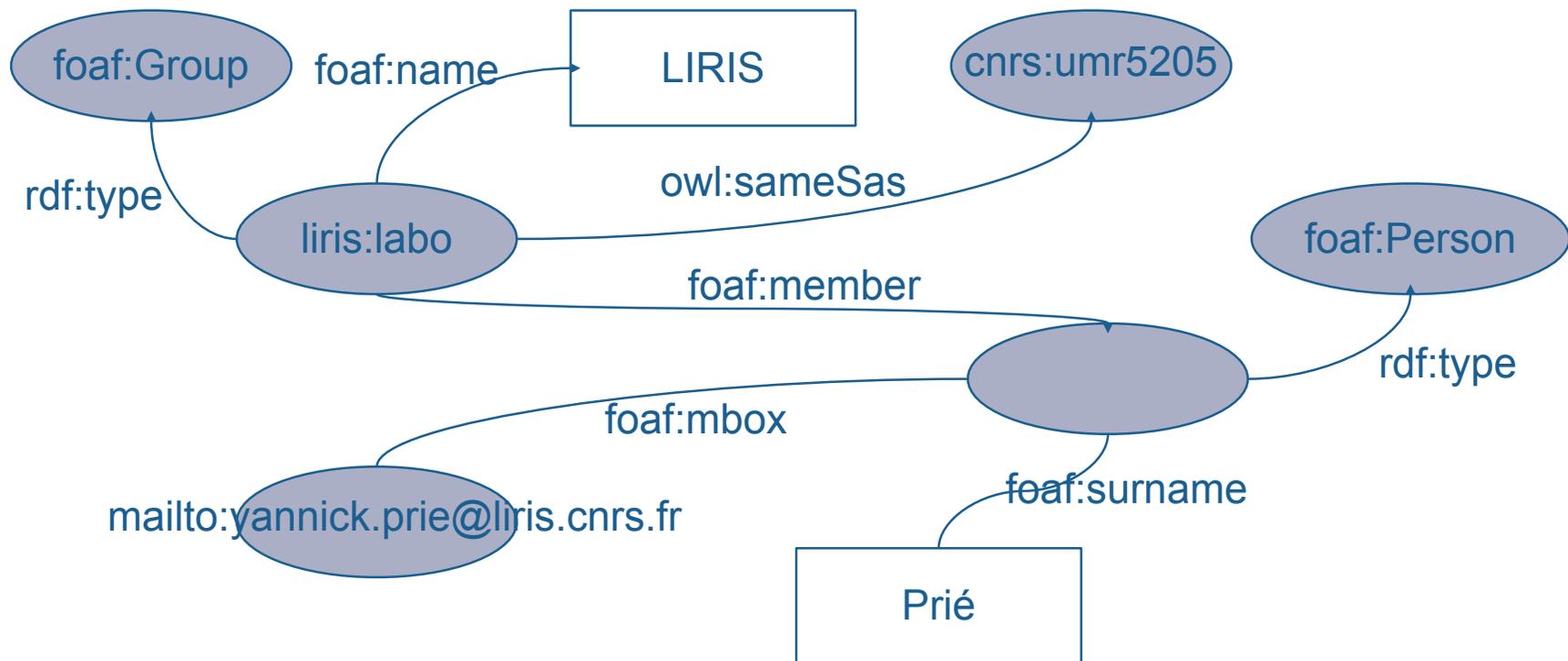
```
liris:labo rdf:type foaf:Group ;
            foaf:name "LIRIS" ;
            owl:sameAs cnrs:umr5205 ;
            foaf:member _:pac .

_:pac rdf:type foaf:Person ;
       foaf:surname "Champin" .

[
  dc>Title "Interopérabilité sémantique..." ;
  dc:Creator _:pac ;
  dc:Source <http://liris.cnrs.fr/mccir-050127.sxi>
].
```

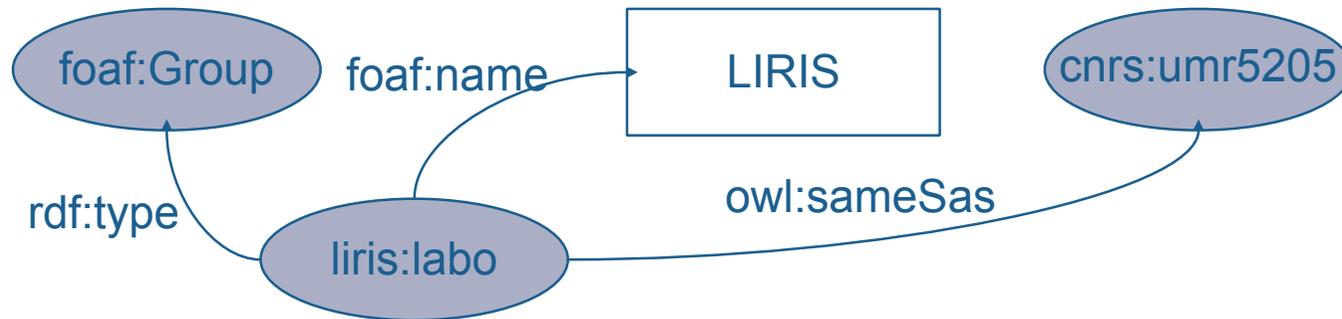
RDF – Sémantique (1)

- Graphe conséquence d'un autre graphe



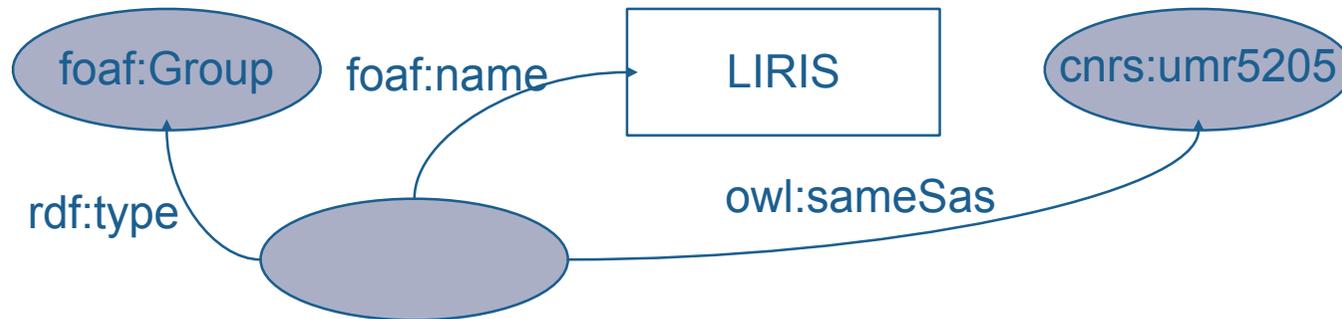
RDF – Sémantique (2)

- Tout sous-graphe est une conséquence :
sémantique **monotone**



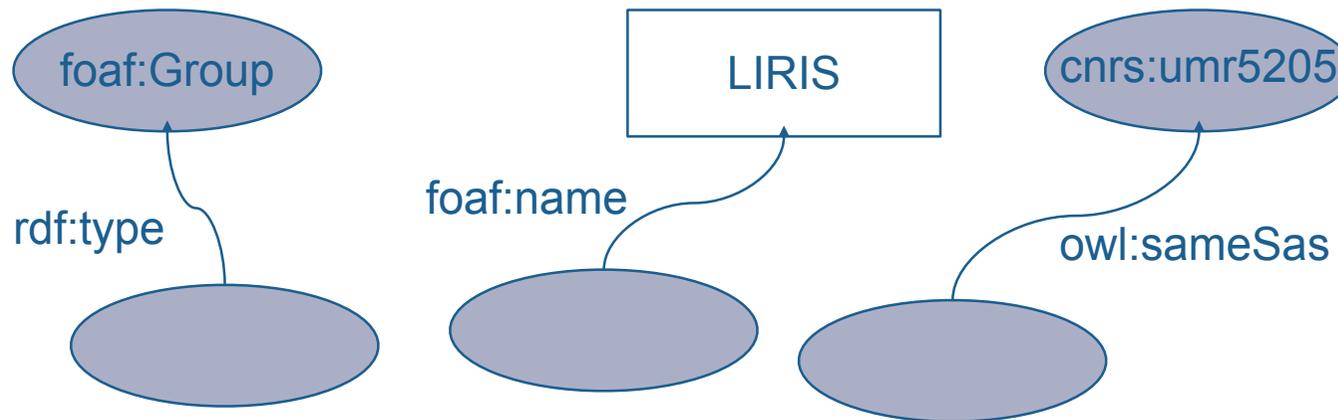
RDF – Sémantique (3)

- Substitution des nœuds vierges



RDF – Sémantique (3)

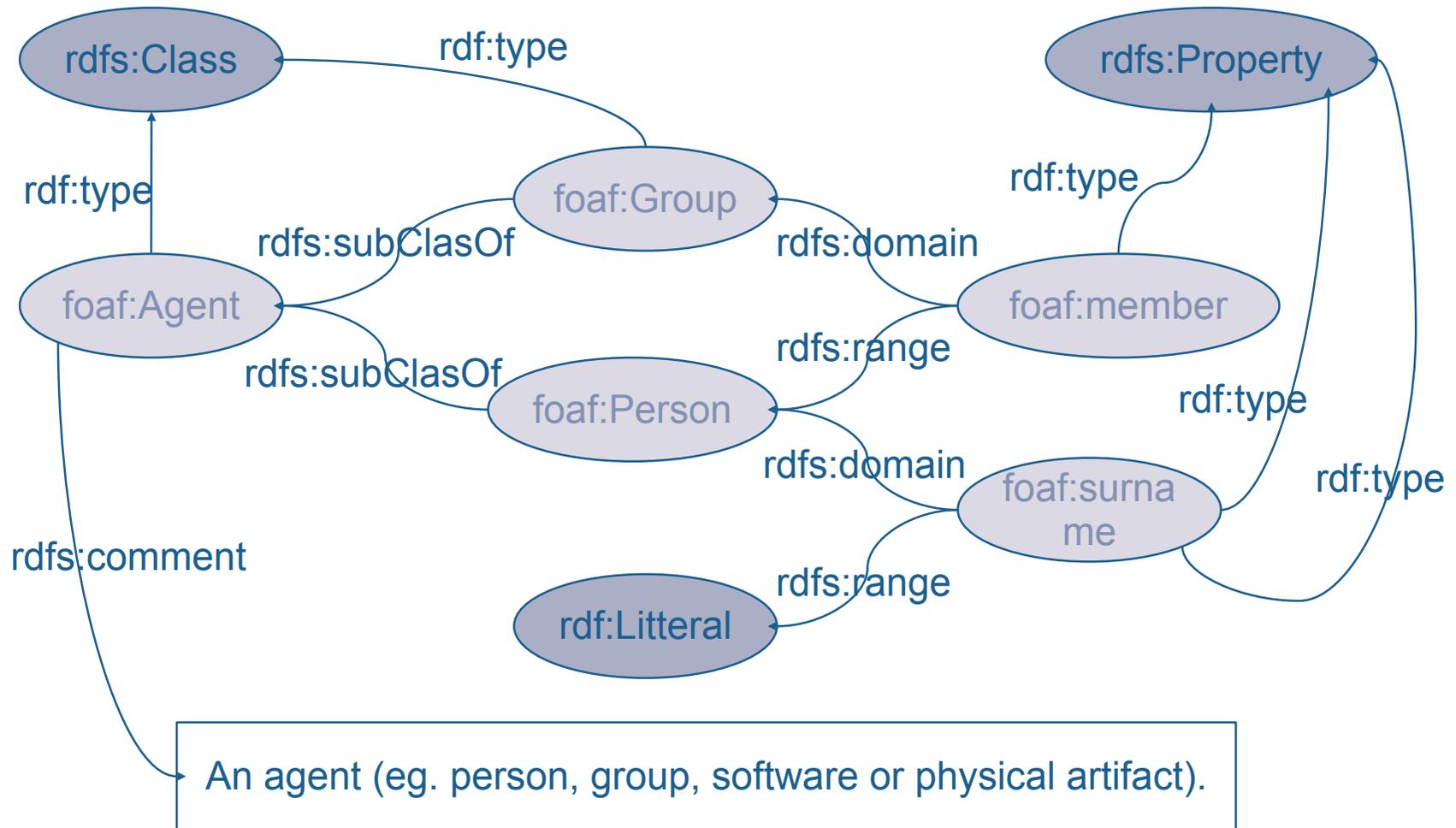
- Éclatement des nœuds vierges



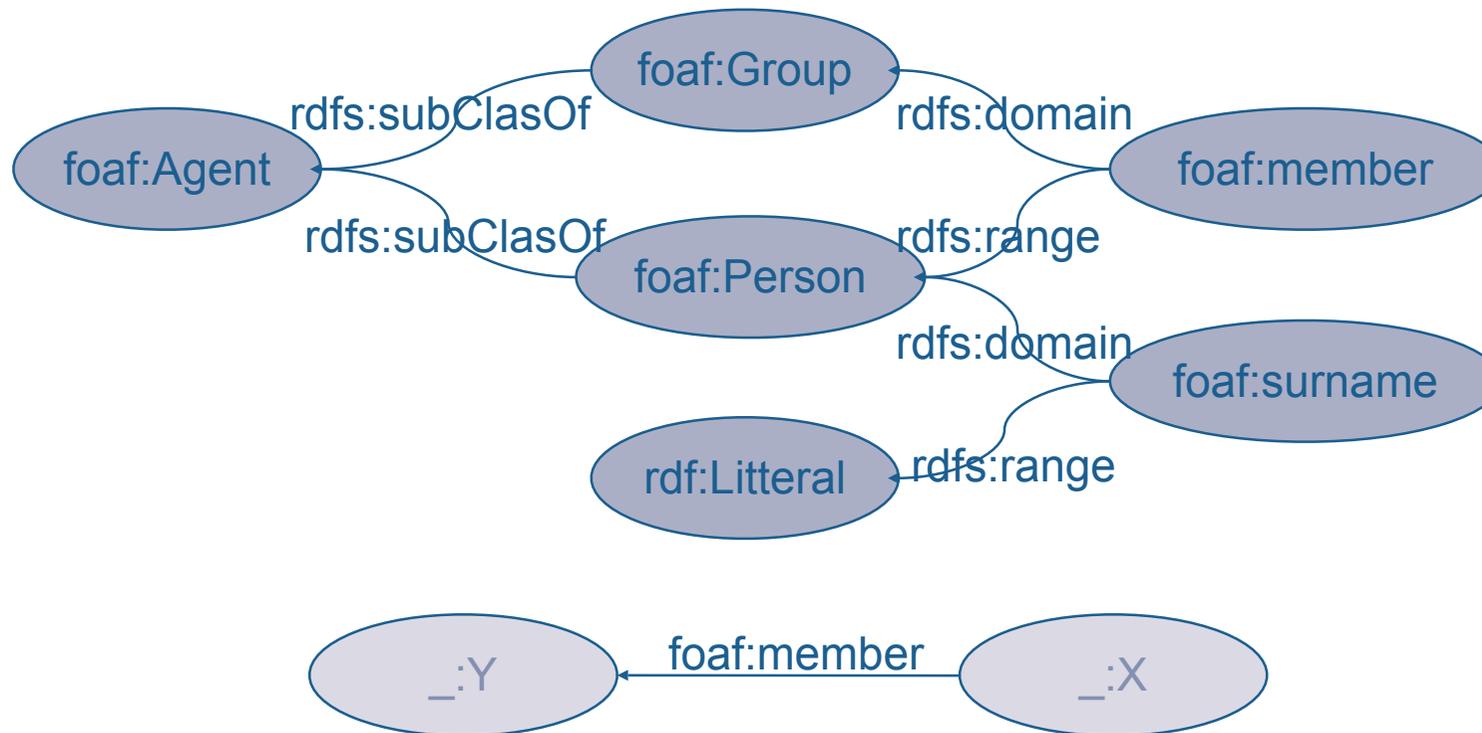
RDF Schema

- Vocabulaire pour
 - décrire des classes (concepts),
 - décrire des propriétés (rôles),
 - les structurer en hiérarchie de spécialisation (\sqsubseteq),
 - contraindre le domaine et la portée des propriétés (\forall limité).
 - peupler une telle ontologie (A-Box).
- Sémantique d'une LD très simple

RDF Schema (exemple)



RDF Schema (exercice)



X est un Groupe ? Y est une Personne ? X est un Agent ?

<http://www.w3.org/TR/rdf-mt/>

RDF Schema

- En fait, RDF Schema n'est pas strictement une LD : les familles de termes (classes, propriétés, individus) ne sont **pas** distinctes
 - :Dumbo rdf:type :Éléphant
 - :Éléphant rdfs:subClassOf :Mammifère
 - :Éléphant rdf:type :Espèce
 - :Espèce rdf:type rdfs:Class
- Pas gênant car il n'y a pas de termes complexes, et notamment pas de négation
 - on ne peut pas exprimer de contradiction en RDFS
 - les mécanismes d'inférence restent décidables

OWL – Présentation (1)

- Recommandation du W3C
<http://www.w3.org/2004/OWL/>
- Expressivité des LD en RDF
 - réutilise le vocabulaire de RDF Schema
 - logique très expressive (famille SH)
 - domaines concrets (littéraux RDF)
 - implémentations performantes disponibles
 - FaCT <http://www.cs.man.ac.uk/~horrocks/FaCT/>
 - Racer <http://www.racer-systems.com/>
 - Pellet <http://www.mindswap.org/2003/pellet/>

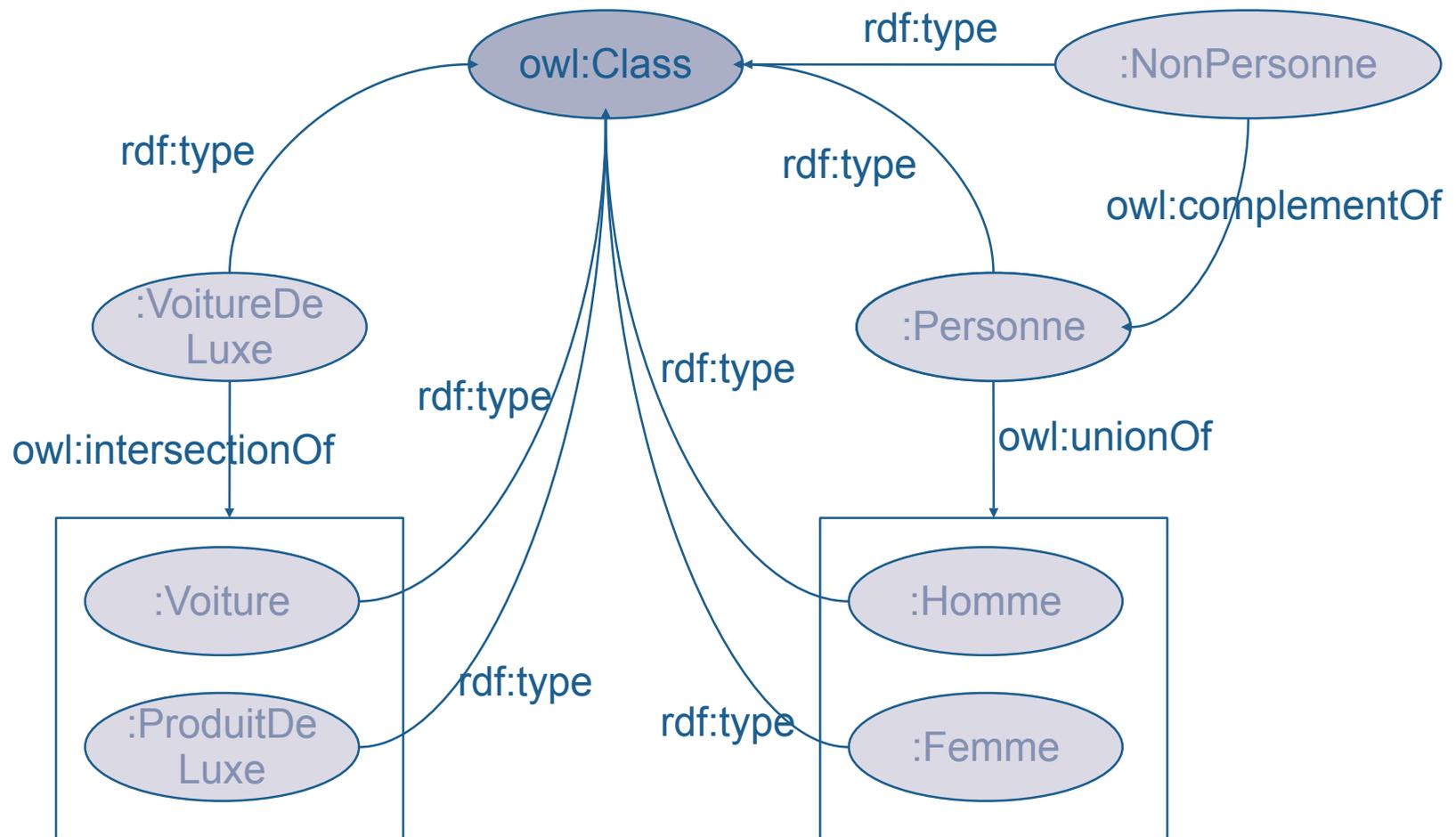
OWL – Présentation (2)

- Différences avec les LD
 - utilise les URIs comme termes
 - différence de méta-terminologie (RDF Schema)
 - concept → classe
 - rôle → propriété
 - les axiomes s'expriment en RDF
 - non atomiques (triplets)
 - sémantique d'un triplet isolé ?

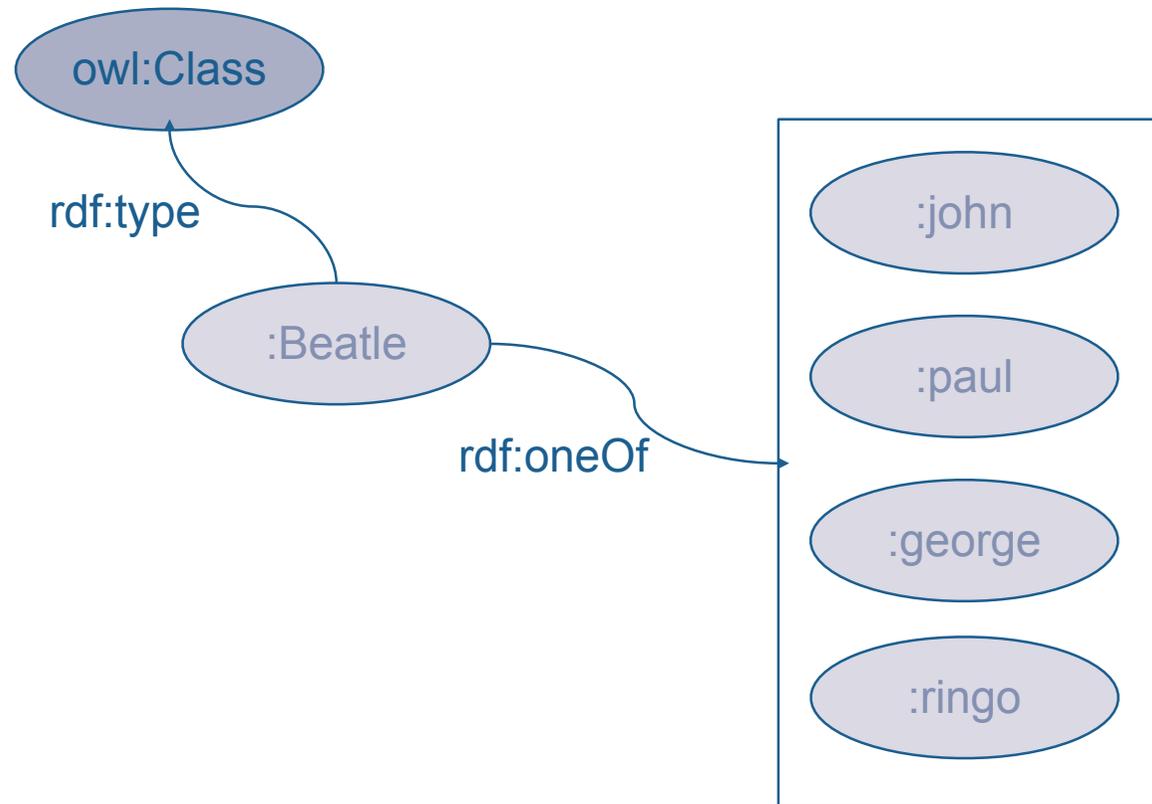
OWL

- Hiérarchies de Classes
- Hiérarchies de Propriétés
 - Types de données simples
 - « Propriétés objets (relations entre classes) »
- Descriptions de classes
 - Peuvent être utilisées au lieu de classes nommées
 - Énumérations
 - une EstimationDeCrise est soit normal, soit dommagesMatériels, soit grave, soit trèsGrave
 - Restrictions
 - une Ville est un LieuGéo qui aCommeHabitants au moins 3000
 - Enoncés logiques
 - quelquechose qui soit RouteSecours et pas RouteTerrestre

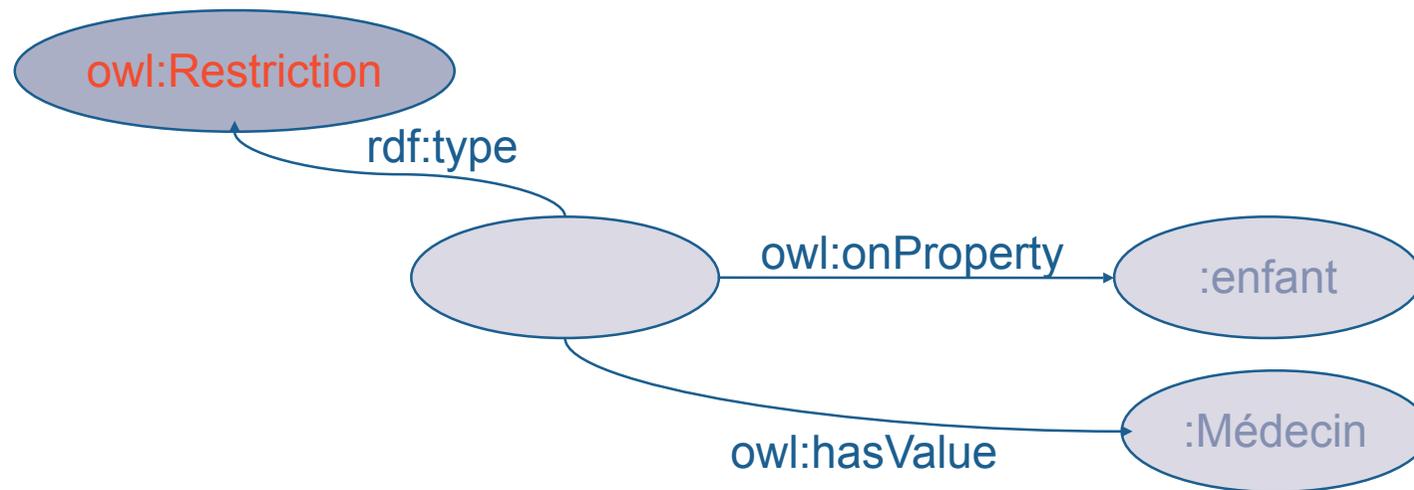
Classes OWL (op. ensemblistes)



Classes OWL (extension)

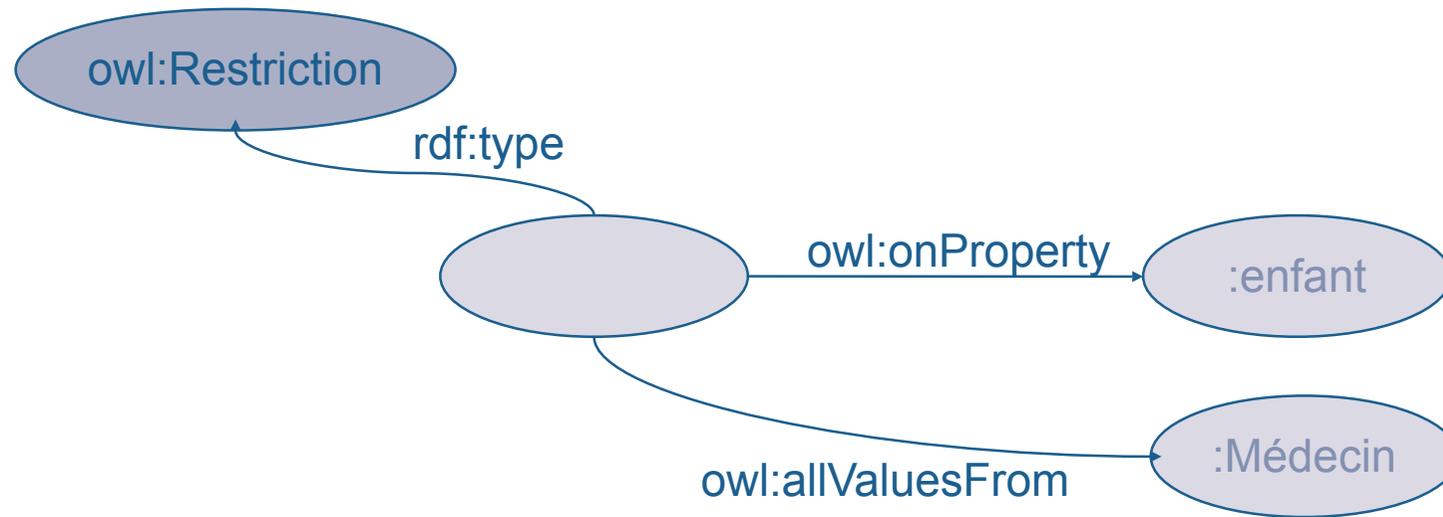


Classes OWL (restrictions)



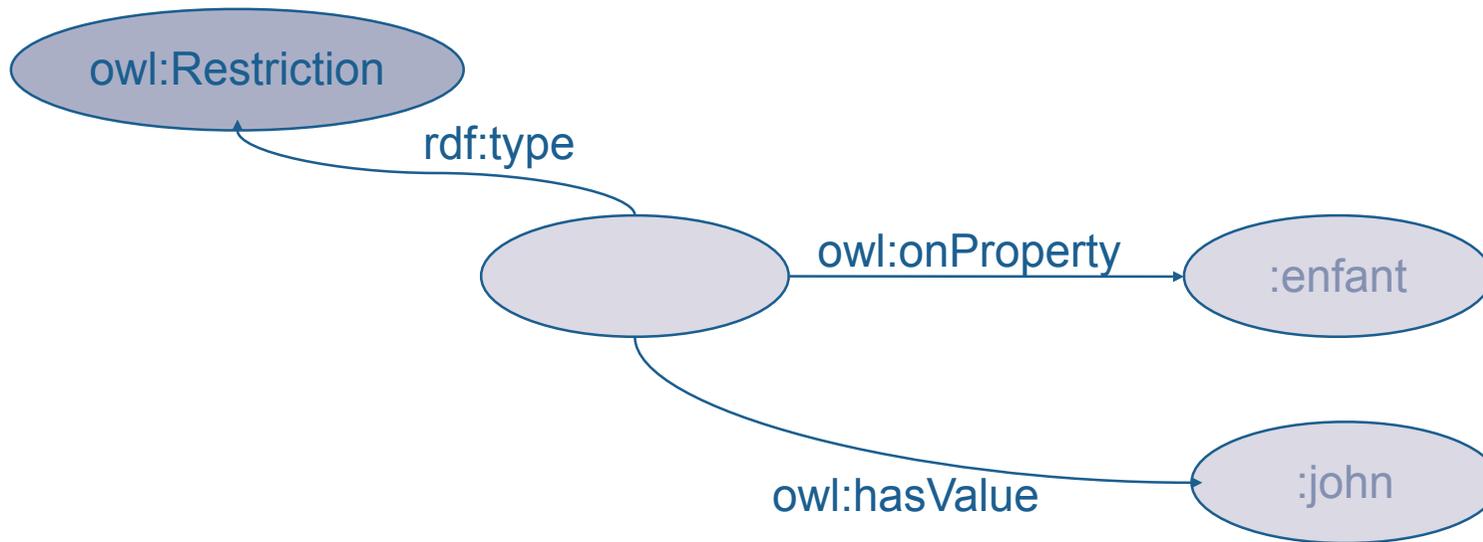
\exists enfant.Médecin

Classes OWL (restrictions)



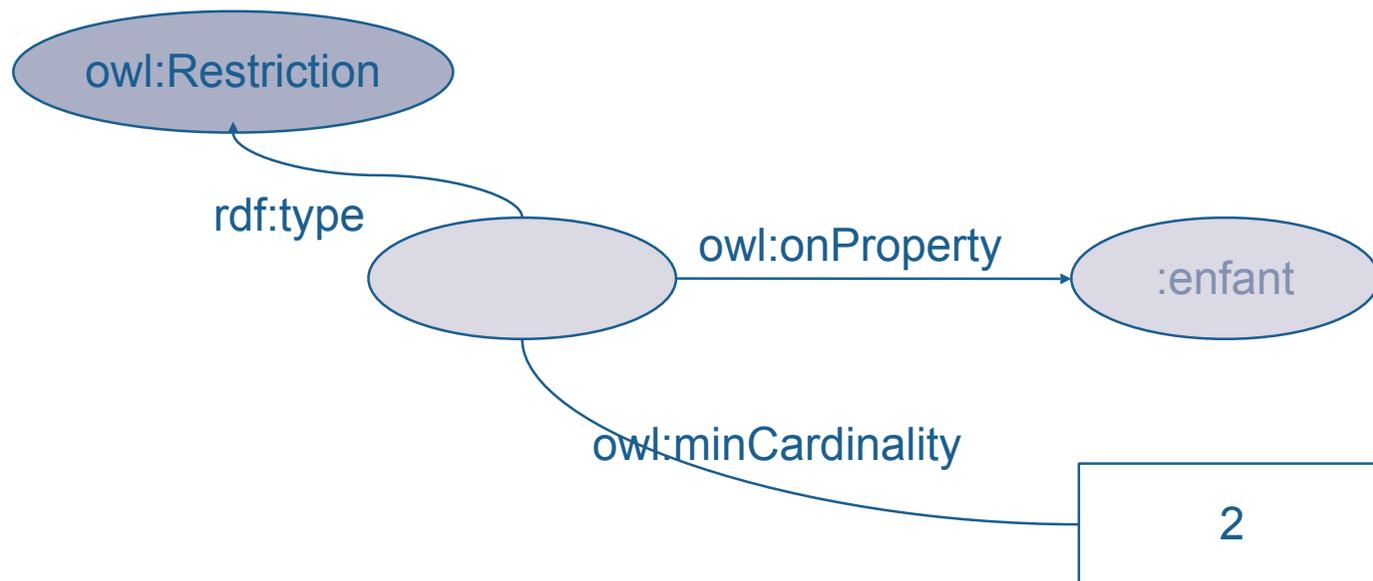
∀ enfant.Médecin

Classes OWL (restrictions)



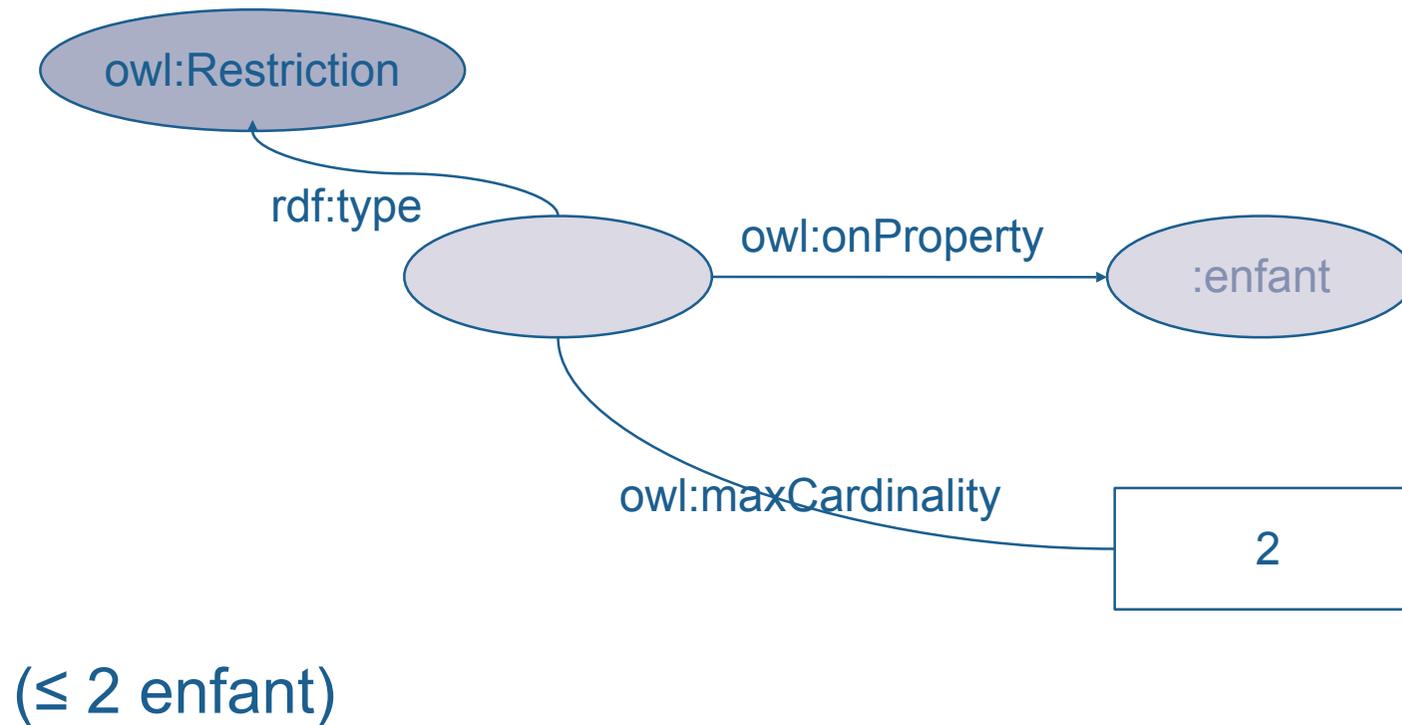
\exists enfant.{john}

Classes OWL (restrictions)

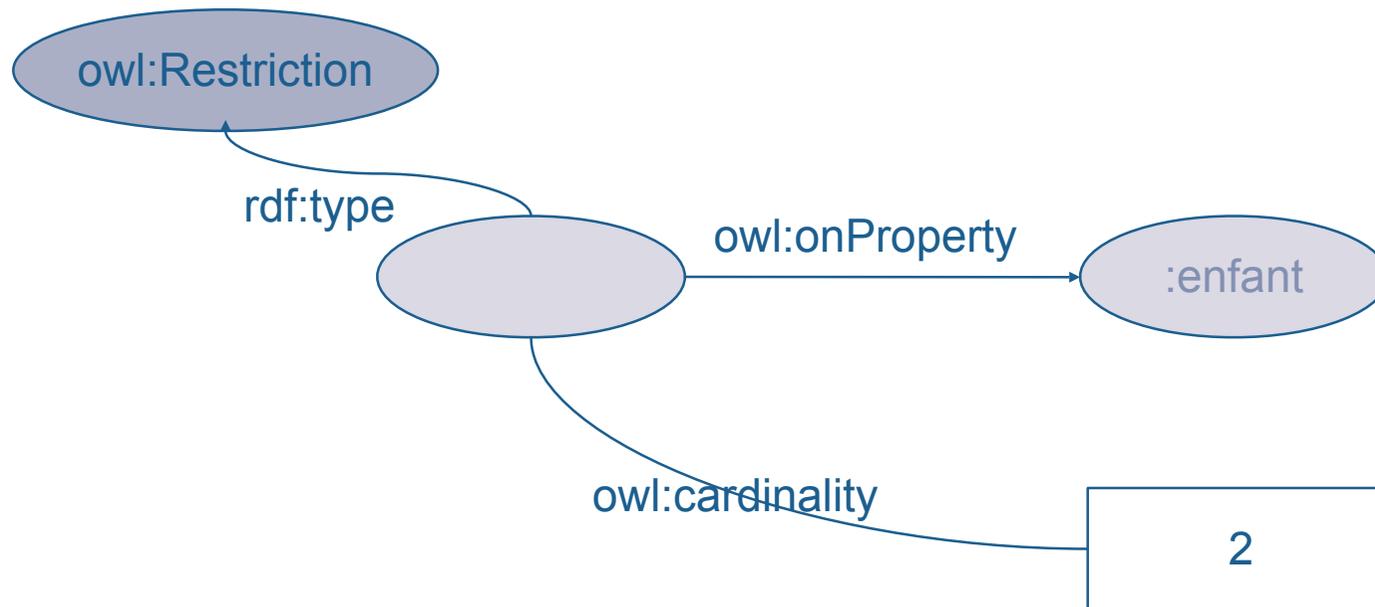


(≥ 2 enfant)

Classes OWL (restrictions)

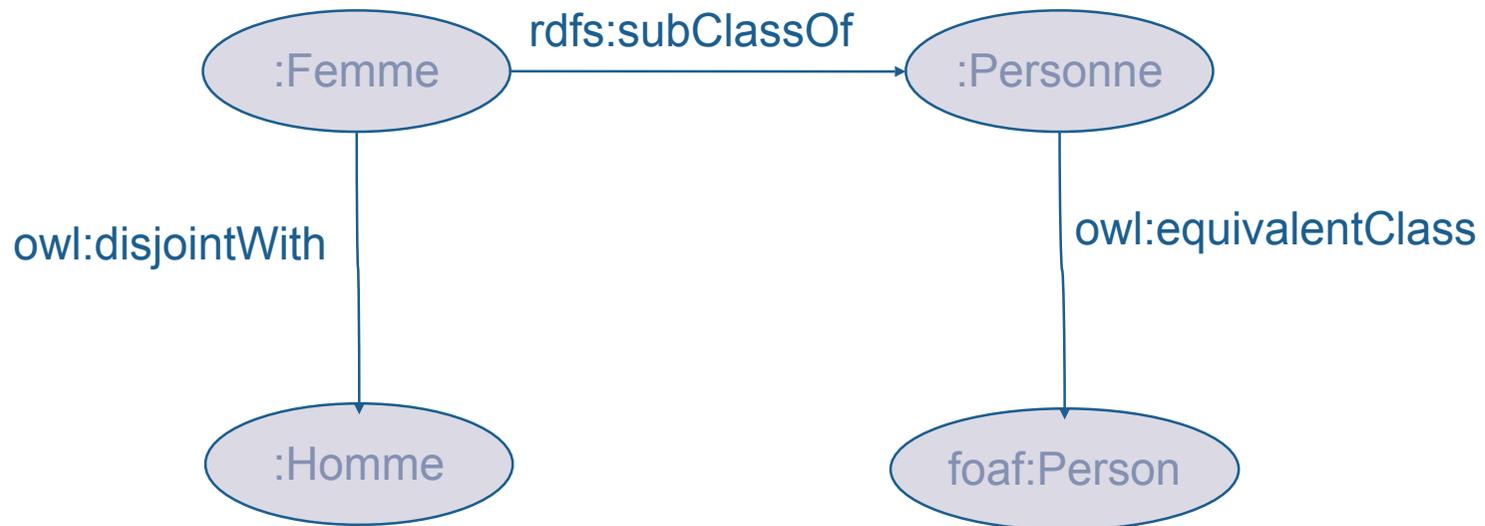


Classes OWL (restrictions)



(= 2 enfant)

Relations entre classes OWL



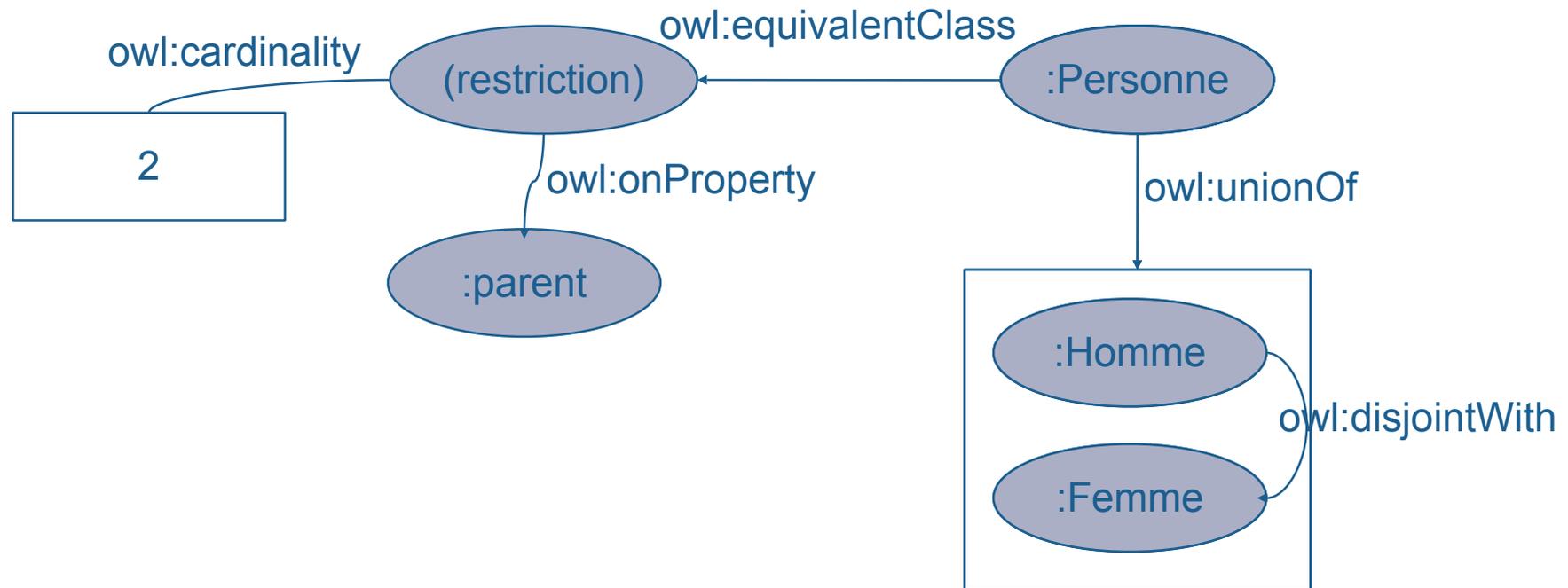
Propriétés

- Relation entre propriétés
 - Inverse : owl:inverseOf
 - Equivalence : owl:equivalentProperty
- Propriétés spéciales
 - Transitivité : owl:TransitiveProperty
 - Symétrie : owl:SymmetricProperty
 - Fonctionnelle : owl:FunctionalProperty
 - un seul individu en relation par cette propriété
 - Inverse fonctionnelle : owl:InverseFunctionalProperty
 - une ressource unique pour une valeur de la propriété

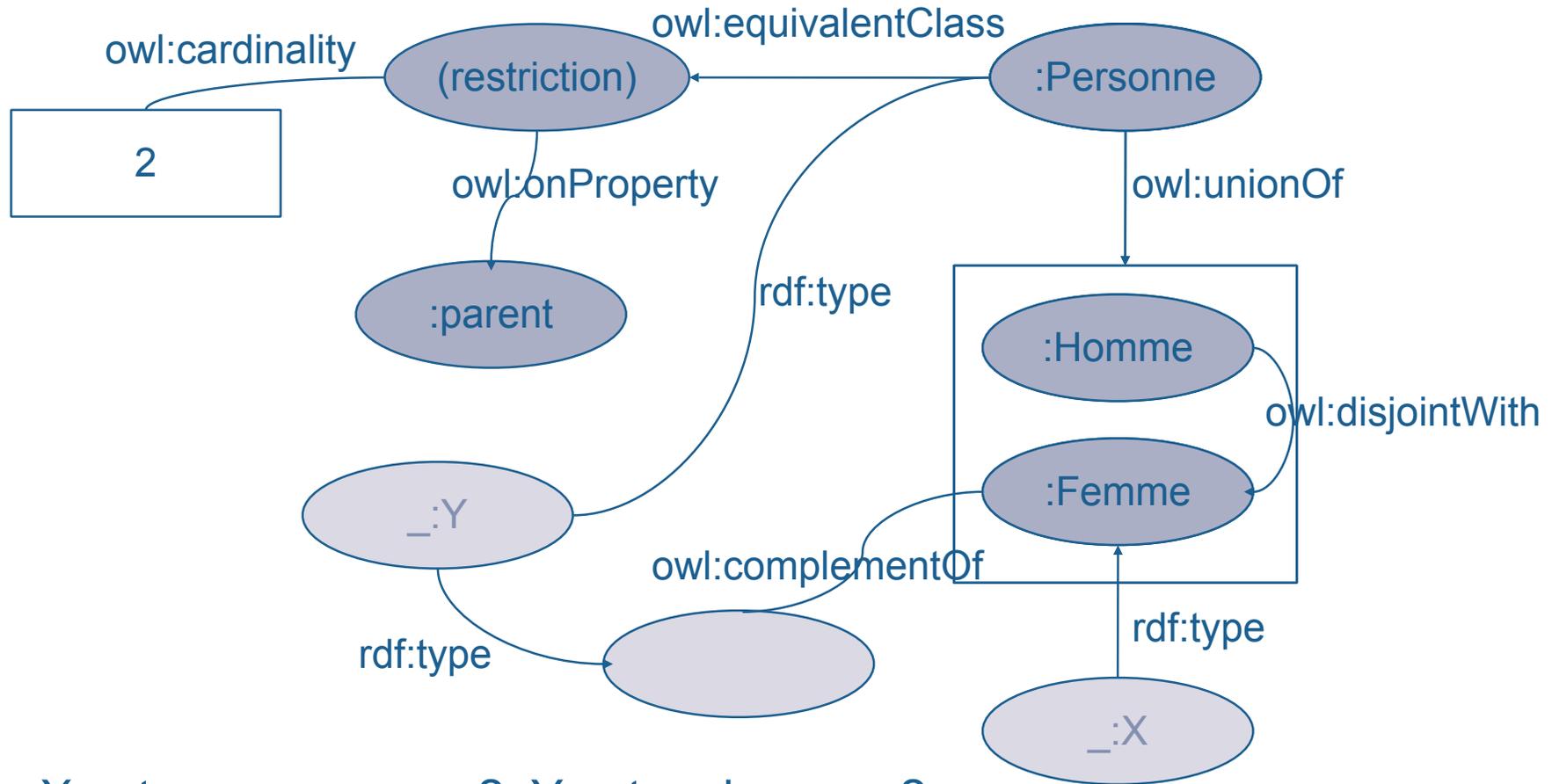
Individus

- Identité : owl:sameAs
- Différence : owl:differentFrom

Classes OWL (exercices)

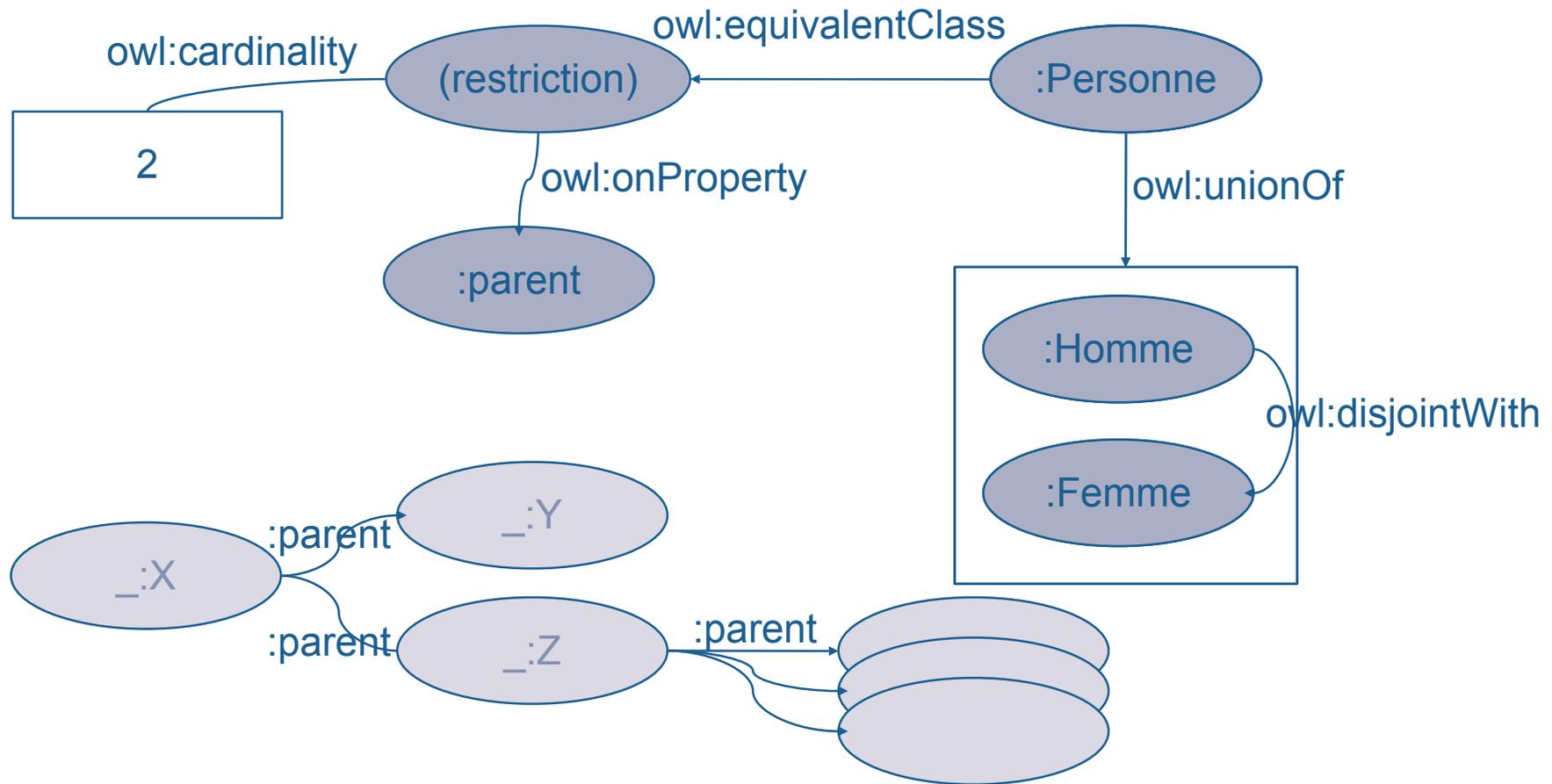


Classes OWL (exercices)



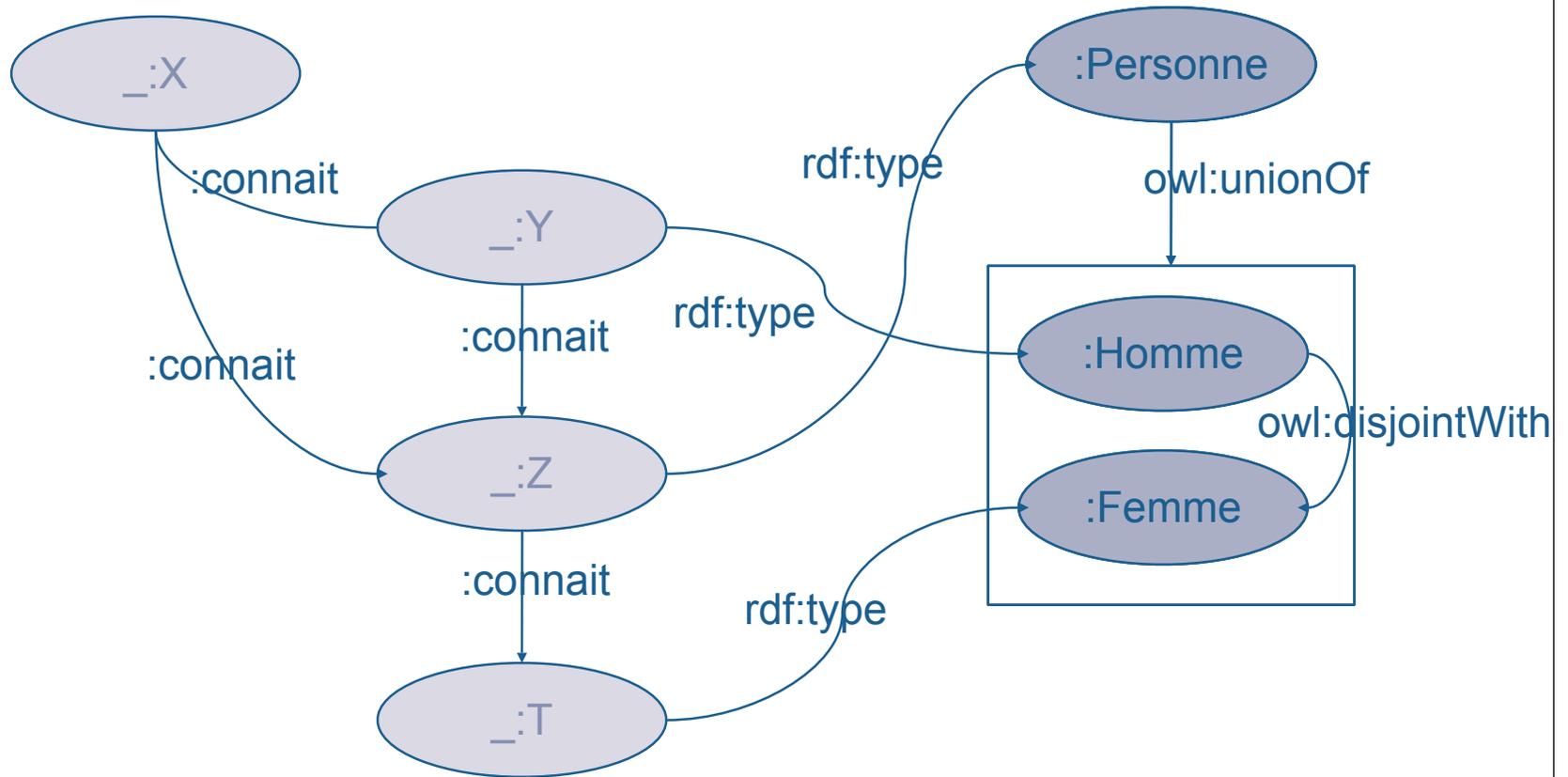
X est une personne ? Y est un homme ?

Classes OWL (exercices)



X est une personne ? Y est une personne ? Z est une personne ?

Classes OWL (exercices)



X connaît un homme qui connaît une femme ?

Syntaxe XML / OWL

```
<owl:Class rdf:ID= "Latitude">
  <rdfs:subClassOf rdf:resource="#MeasureGeo" />
</owl:Class>
<owl:ObjectProperty rdf:ID="aPourEstimation">
  <rdf:type rdf:resource="&owl;FunctionalProperty" />
  <rdfs:domain rdf:resource="#LieuGeo"/>
  <rdfs:range rdf:resource="#EstimationDeCrise"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="VilleNormale">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Ville" />
    <owl:Restriction>
      <owl:onProperty rdf:resource="#aPourEstimation" />
      <owl:hasValue rdf:resource="#normal" />
    </owl:Restriction>
  </owl:intersectionOf>
</owl:Class>
```

OWL – Modularité

- Utilisation d'URIs comme identificateurs
 - pas de collision de noms entre ontologies
 - permet de faire référence à des termes d'une autre ontologie
 - exemple : foaf utilise le vocabulaire de wordnet, wot utilise le vocabulaire de foaf
 - ATTENTION : pas de sémantique définie
 - possibilité d'importer une ontologie
 - inclusion des axiomes de l'ontologie importée

Différentes versions d'OWL

- OWL Lite
 - Hiérarchie de classification
 - Contraintes simples
 - Outils inférentiels plus simples et efficaces
- OWL DL
 - Correspond à certaines logiques de descriptions
 - Complétude
 - Décidabilité
- OWL Full
 - Une classe peut être considérée comme instance d'une classe
 - Complètement compatible avec RDFS
- Les implémentations OWL
 - <http://www.w3.org/2001/sw/WebOnt/impls>

OWL Lite

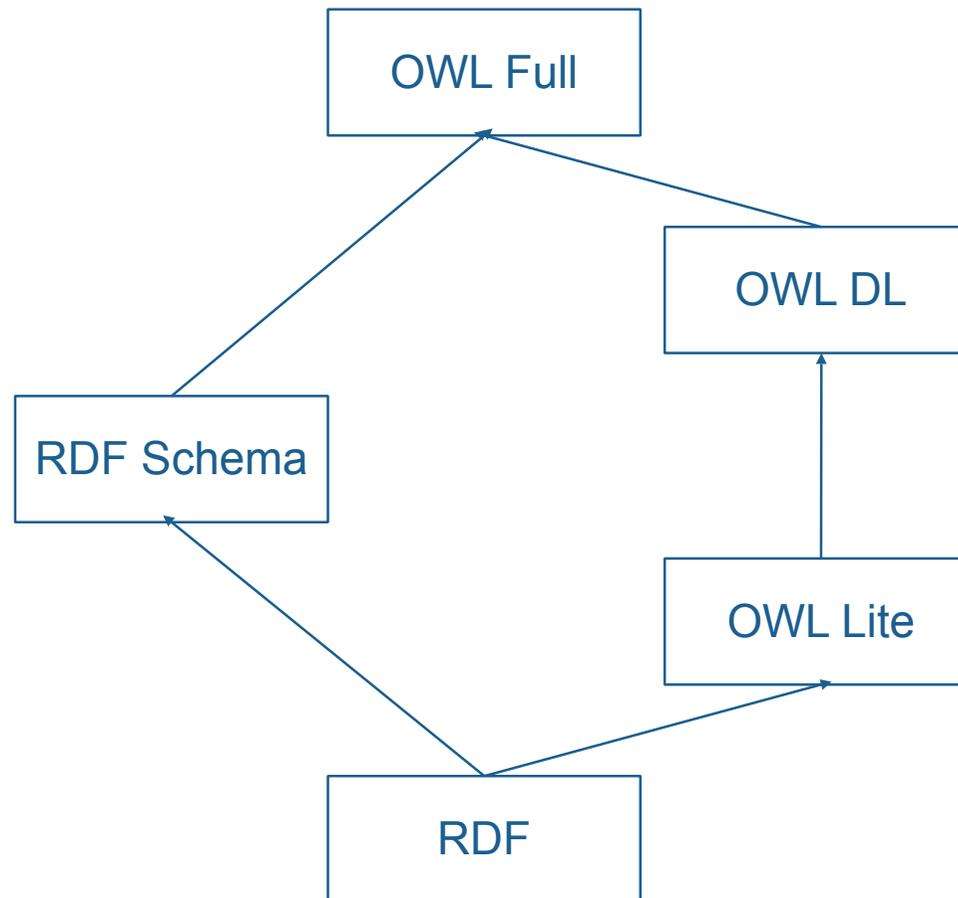
- RDF Schema Features:
 - [Class](#)
 - [rdf:Property](#)
 - [rdfs:subClassOf](#)
 - [rdfs:subPropertyOf](#)
 - [rdfs:domain](#)
 - [rdfs:range](#)
 - [Individual](#)
- (In)Equality:
 - [equivalentClass](#)
 - [equivalentProperty](#)
 - [sameAs](#)
 - [differentFrom](#)
 - [allDifferent](#)
- Property Characteristics:
 - [inverseOf](#)
 - [TransitiveProperty](#)
 - [SymmetricProperty](#)
 - [FunctionalProperty](#)
 - [InverseFunctionalProperty](#)
- Property Type Restrictions:
 - [allValuesFrom](#)
 - [someValuesFrom](#)
- Restricted Cardinality:
 - [minCardinality](#) (only 0 or 1)
 - [maxCardinality](#) (only 0 or 1)
 - [cardinality](#) (only 0 or 1)
- Header Information:
 - [ontology](#)
 - [imports](#)
- Class Intersection:
 - [intersectionOf](#)
- Versioning:
 - [versionInfo](#)
 - [priorVersion](#)
 - [backwardCompatibleWith](#)
 - [incompatibleWith](#)
 - [DeprecatedClass](#)
 - [DeprecatedProperty](#)
- Annotation Properties:
 - [rdfs:label](#)
 - [rdfs:comment](#)
 - [rdfs:seeAlso](#)
 - [rdfs:isDefinedBy](#)
- Datatypes
 - [DatatypeProperty](#)

OWL DL and FULL

- Class Axioms:
 - [oneOf, dataRange](#)
 - [disjointWith](#)
 - [equivalentClass](#)
(applied to class expressions)
 - [rdfs:subClassOf](#)
(applied to class expressions)
- Boolean Combinations of Class Expressions:
 - [unionOf](#)
 - [intersectionOf](#)
 - [complementOf](#)
- Arbitrary Cardinality:
 - [minCardinality](#)
 - [maxCardinality](#)
 - [cardinality](#)
- Filler Information:
 - [hasValue](#)

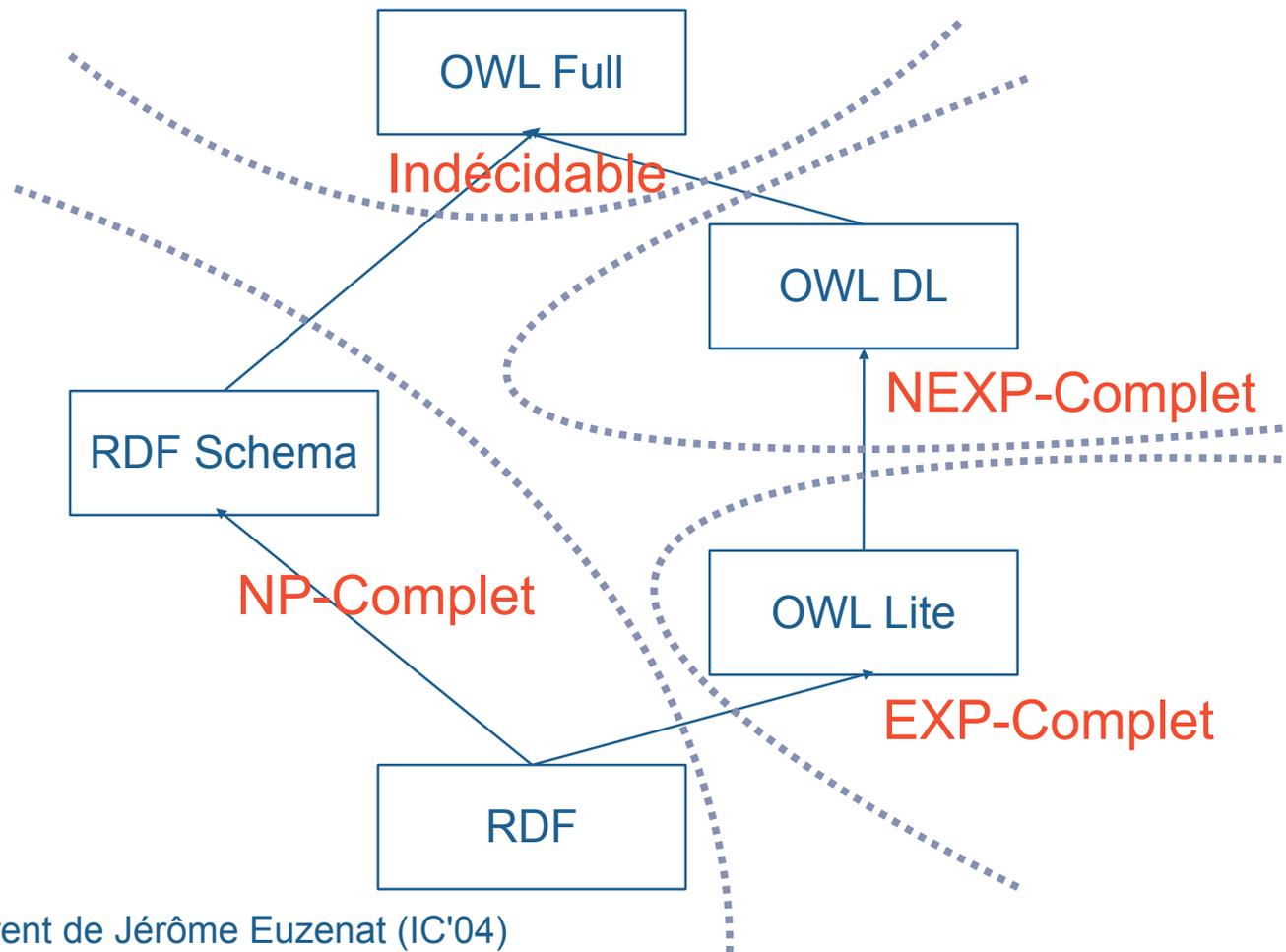
Quelques restrictions pour OWL DL par rapport à OWL FULL

OWL – Modularité (suite)



D'après un transparent de Jérôme Euzenat (IC'04)

OWL – Modularité (3)



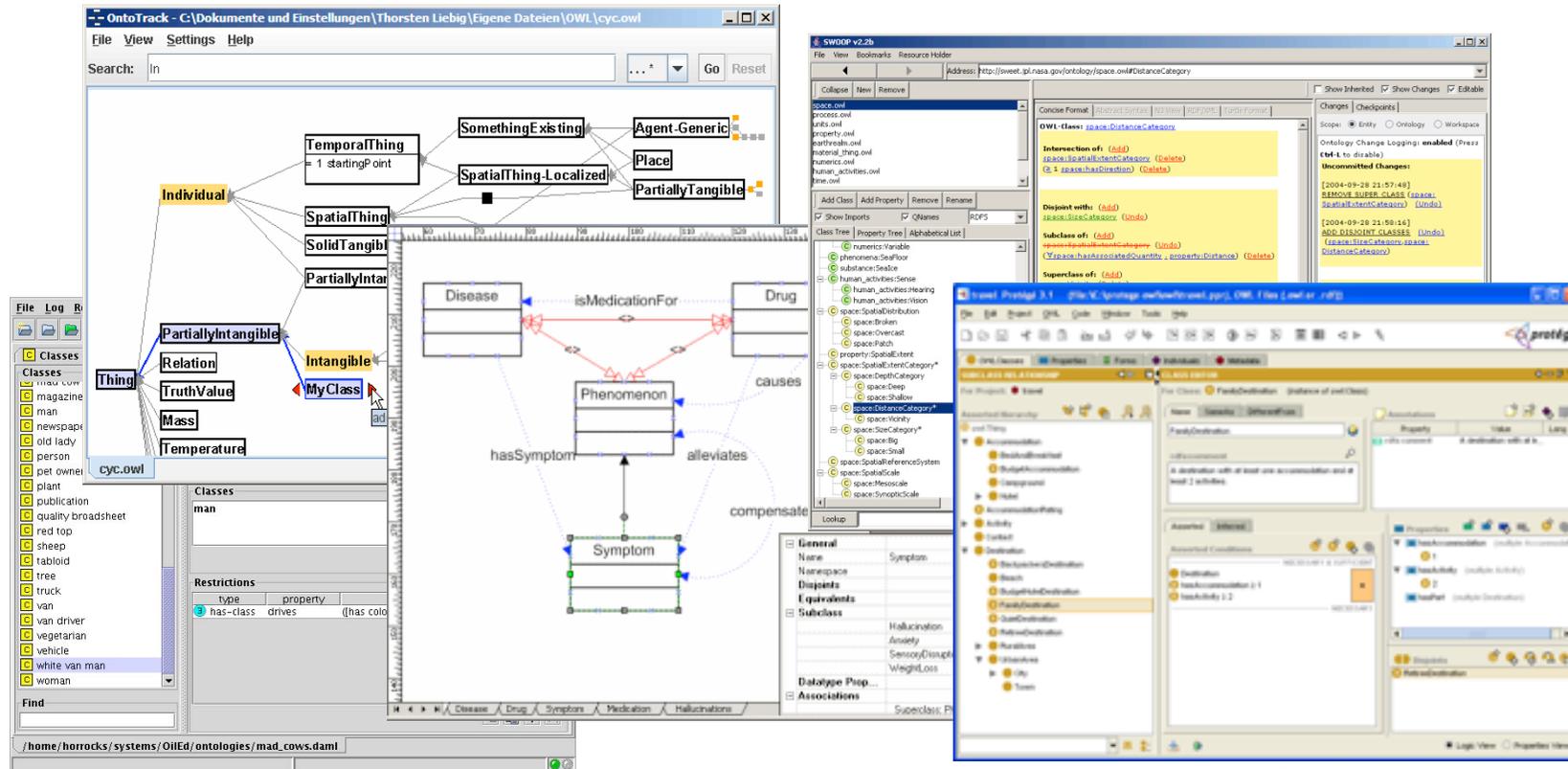
D'après un transparent de Jérôme Euzenat (IC'04)

SPARQL

- Langage de requête pour RDF
- Définit aussi un protocole de communication
- Dépend de la sémantique formelle utilisée
 - conséquence : la même requête SPARQL appliquée au même graphe donnera des résultats différents selon qu'on applique la sémantique de RDF, RDFS, OWL
- Implémentation disponible dans Pellet (utilisant la sémantique de OWL)

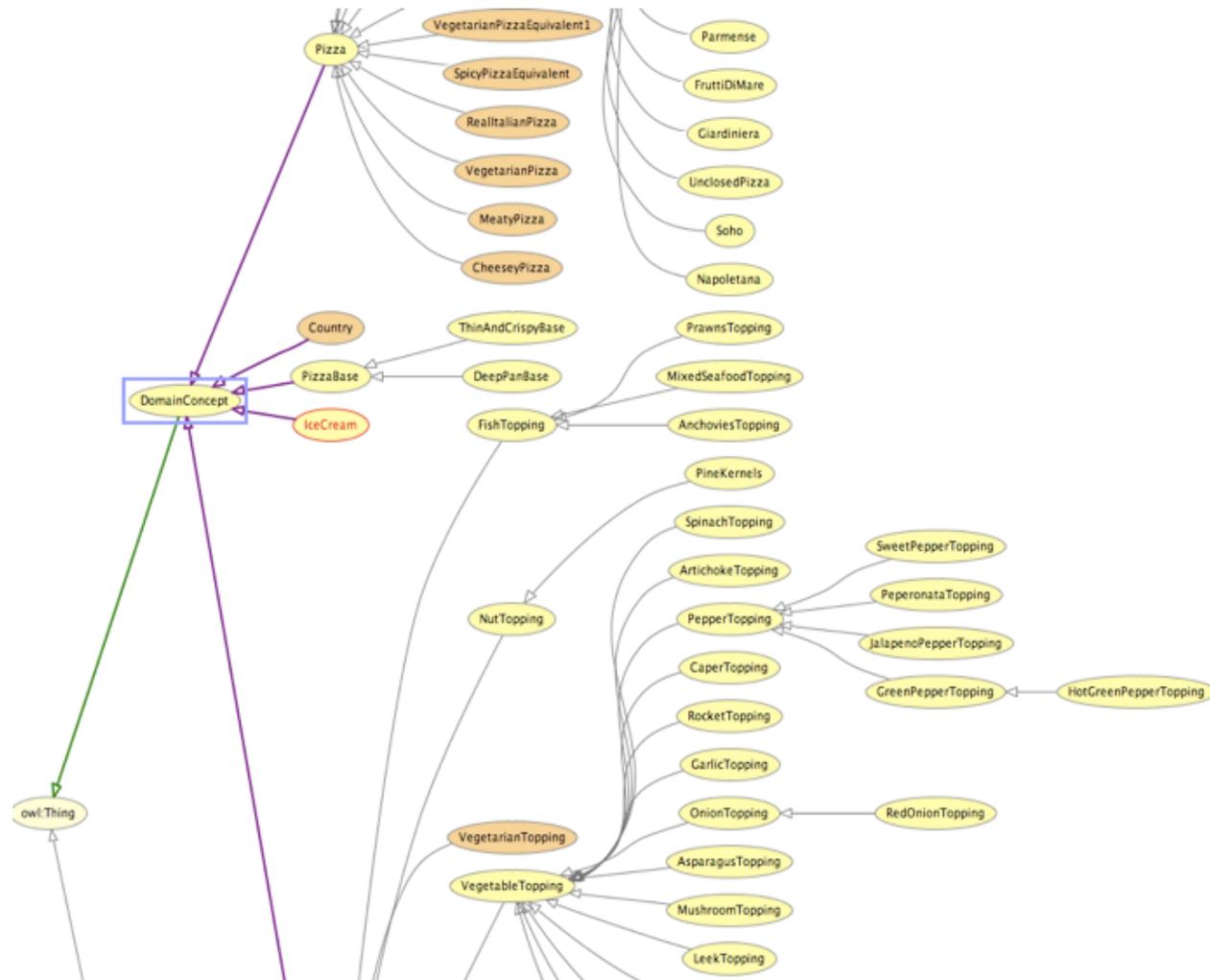
Editeurs OWL

- Editeurs
 - Swoop, Protege, KAON, OntoEdit, Oiled, Doe, Construct, Ontotrack

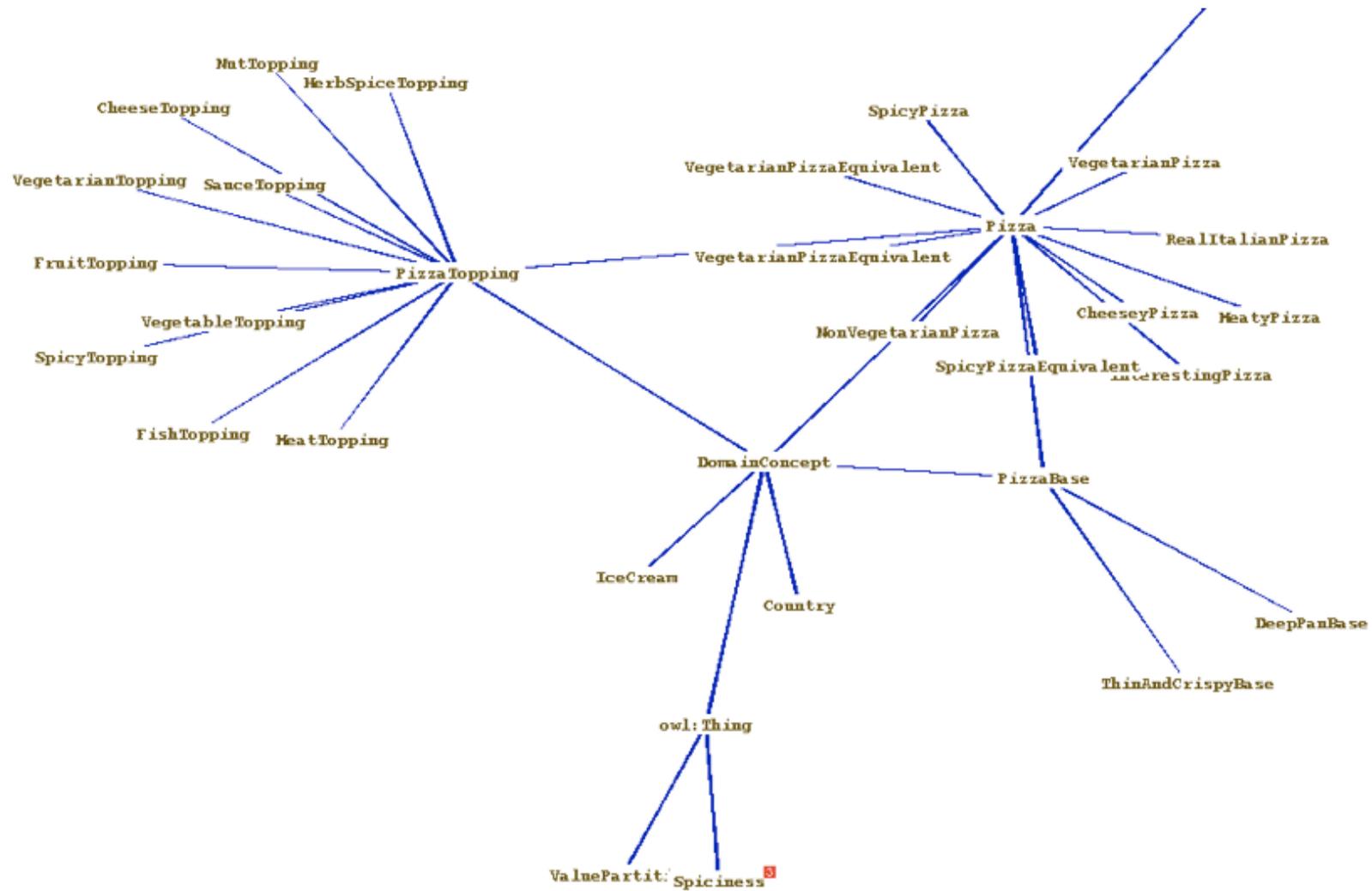


(Horrocks)

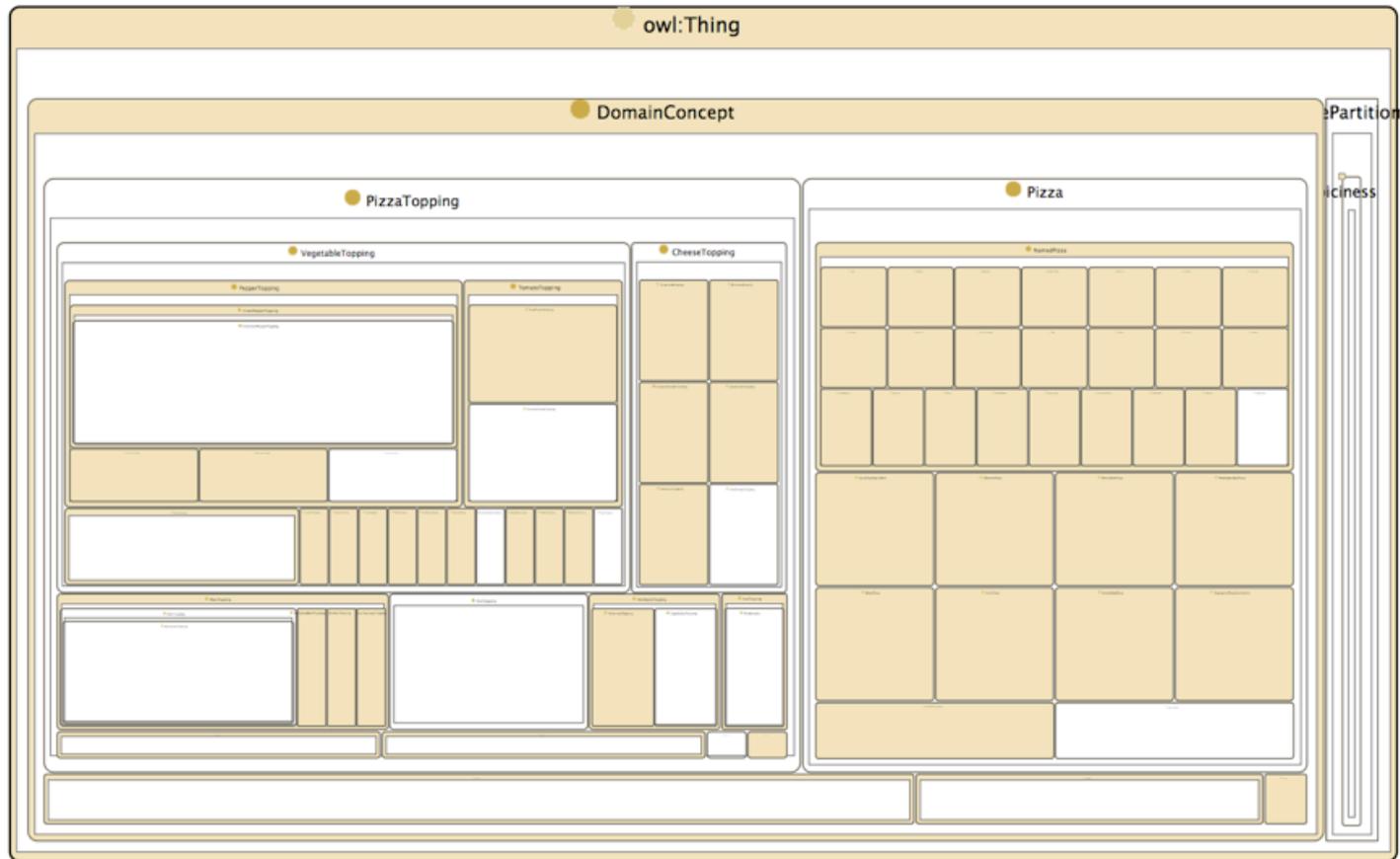
Visualiser et interagir avec des ontologies



Visualiser et interagir avec des ontologies



Visualiser et interagir avec des ontologies



Des ontologies OWL

- Liste sur
 - http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library
- Voir aussi swoogle
 - <http://swoogle.umbc.edu/>

Utilisation dans des projets industriels

- Gestion des connaissances
- Conception
- Héritage culturel
- Semantic desktop
- Voir par exemple "formal ontologies meet industry"
 - <http://www.fomi2008.di.unito.it/>

OWL 2.0

- Restrictions de cardinalité qualifiées
 - $>n$ R.C
- Composition limitée de rôles
 - $r1$ o $r2$
- Assertions de rôles
 - propriétés disjointes, réflexives, asymétriques...
- Chaînes de propriétés
- ...
- Compatibilité ascendante avec OWL
- Disponible dans certains outils
 - Protege
 - Pellet, FACT++

Tendances

- Ontologies modulaires (cf. composants en OO)
 - réutilisation partielle
 - vérification d'intégration
 - évolution séparée
- Toujours plus d'expressivité
- Meilleur passage à l'échelle

Plan

- Ingénierie des connaissances
- Ontologies
- Logiques de description
- Introduction au Web sémantique
- OWL
- **Conception d'ontologies**

Objectif de la conception

- Formalisation : lever les ambiguïtés, cohérence
- Indépendance (relative) par rapport à une tâche, ou un problème précis, modularité
- Utilisable pour différentes tâches : conception, diagnostic, maintenance, recherche d'information
- Indépendante d'une implémentation : limiter le biais dû à un formalisme de représentation : niveau conceptuel
- Possibilité d'inférences (relations transitives, axiomes, etc.)
- Possibilité d'évolution
- Partage : consensus / standardisation

Les tâches liés à la conception et à la reconception

- Conception d'une ontologie
- Raffinement d'ontologie
- Gestion des erreurs (évolution)
- Intégration d'ontologies
- Réutilisation d'ontologies

Construire une ontologie

- A la main
 - Travail avec experts : entretiens, observation
 - A partir de rien ou en réutilisant
 - ontologies dans des domaines proches
 - *ontologies de haut-niveau* suffisamment génériques
 - SUMO, DOLCE
 - Méthodologies de modélisation conceptuelle
 - e.g. pour spécialiser un concept, pour choisir entre propriété et concept, pour « nettoyer » une ontologie...
 - Outil de validation de la cohérence formelle
 - Bonnes pratiques
 - voir ISWC-Tutorial-Best-practice-2005.ppt (de <http://www.co-ode.org/resources/tutorials/iswc2004/>)

Construire une ontologie

- A partir de corpus documentaire
 - outils de TAL: Traitement Automatique des Langues
 - Extraction de candidats-termes, candidats-relations, etc.
 - dans des domaines s'y prêtant bien, avec des méthodologies qui demandent encore à être affinées à la main
 - méthodologie du groupe TIA
 - construction de modèles terminologiques à partir de textes
 - qui servent de base à la construction d'ontologies formelles

Méthode simple Stanford

- Natalya F. Noy and Deborah L. McGuinness (2001) *Ontology Development 101: A Guide to Creating Your First Ontology*
- Step 1
 - Determine the domain and scope of the ontology
- Step 2
 - Consider reusing existing ontologies
- Step 3
 - Enumerate important terms in the ontology
- Step 4
 - Define the classes and the class hierarchy
- Step 5
 - Define the properties of classes—slots
- Step 6
 - Define the facets of the slots
- Step 7
 - Create instances

Autres méthodes / outils

- Methontology
 - <http://rhizomik.net/~roberto/thesis/html/Methodology.html>
 - Une description complète du processus de conception d'une ontologie
- Liens outils / méthodologies
 - WebODE (Espagne)
 - KAON / Text2Onto (Allemagne)
 - DOE (France)

DOE

- Differential Ontology Editor
- <http://homepages.cwi.nl/~troncy/DOE/>
- "The specification process is divided in 3 steps.
 - 1st step, the user is invited to build taxonomies of concepts and relations, explicitly justifying the position of each item (notion) in the hierarchy. For each notion, the user builds a definition following 4 principles which come from the Differential Semantics theory. Hence, the user has to explicit why a notion is similar but more specific than its parent (2 principles), and why this notion is similar but different from its siblings (2 others principles). The user can also add synonyms and encyclopedic definition in a few languages for all notions.
 - 2nd step, the two taxonomies are considered from an extensional semantics point of view. The user can augment them with new entities (defined) or add constraints onto the domains of the relations.
 - 3rd step, the ontology can be translated into a knowledge representation language, which allows to use it in an appropriate ontology-based system or to import it into another ontology-building tool to specify it further:

Méthode avec TAL

- Voir par exemple
 - Didier Bourigault et Nathalie Aussenac-Gilles (2003) Construction d'ontologies à partir de textes
 - www.sciences.univ-nantes.fr/info/recherche/taln2003/articles/tut3.pdf
 - Extraits :
- Méthode TERMINAE (Szulman)
 - Partir de textes du domaine comme sources de connaissances : ils constituent un support tangible, rassemblant des connaissances stabilisées qui servent de référence et améliorent la qualité du modèle final ;
 - Enrichir le modèle conceptuel d'une composante linguistique : l'accès aux termes et aux textes qui justifient la définition des concepts garantit une meilleure compréhension du modèle ;
 - Utiliser des techniques et outils de TAL basés sur des travaux linguistiques : ils permettent l'exploitation systématique des textes et leurs résultats facilitent la modélisation ;
 - Construire des ontologies régionales, c'est-à-dire consensuelles dans un domaine et adaptées à une application, mais non universelles ;
 - Appliquer des principes de modélisation systématiques pour assurer une bonne structuration des données et faciliter la maintenance de l'ontologie.

Méthode avec TAL (suite)

- Etape 1
 - Constitution d'un corpus vise à choisir documents techniques, comptes rendus, livres de cours, etc. à partir d'une analyse des besoins de l'application.
- Etape 2
 - L' étude linguistique consiste à identifier des termes et des relations lexicales, en utilisant des outils de traitement de la langue naturelle (SYNTEX comme extracteur de termes, UPPERY comme outil d'analyse distributionnelle, Caméléon pour l'aide au repérage de relations par des patrons linguistiques, YAKWA comme concordancier).
- Etape 3
 - La normalisation sémantique conduit à définir dans un langage formel des concepts et des relations sémantiques que nous appelons terminologiques car provenant des termes et relations précédemment étudiés (Biébow & Szulman, 1999). Leur structuration en réseau s'appuie sur les résultats du dépouillement des textes tout en tenant compte de l'objectif d'utilisation de l'ontologie. Elle nécessite l'ajout de nouveaux concepts et relations dits de structuration.
- Etape 4
 - La formalisation permet de préciser, compléter et valider le modèle construit lors de la normalisation. L'analyste indique si les concepts sont primitifs ou définis, vérifie que les relations sont à la bonne place pour favoriser un héritage maximum, etc.

Remerciements - emprunts

- La partie sur LD / OWL a été réalisé en reprenant nombre de transparents de Pierre-Antoine Champin (Université Lyon 1)
- Quelques transparents sont repris de Philippe Laublet (Université Paris 5), Alain Mille (Université Lyon 1), Chantal Reynaud (Université Paris Sud)
- Site du Gracq