

RDF (Schema)

E.Coquery

`emmanuel.coquery@univ-lyon1.fr`

`http://liris.cnrs.fr/~ecoquery`

→ Enseignement → MIF18 GDW

Ontologies, RDF, SPARQL, RDF-S

- Ontologie
 - ensemble de connaissances
 - formalisées dans un ou plusieurs langages
 - RDF(S), OWL, ...
- [RDF] : format de données
 - graphes
 - annotés par des IRI et des valeurs
- [SPARQL] : langage d'interrogation pour RDF
- RDF *store* : BD native RDF
- [RDF-S] : schéma non restrictif
 - Permet d'enrichir des graphes RDF

Web sémantique et Linked Open Data

[Web sémantique]

- extension du Web
- liens sémantiques entre les ressources
 - relation ayant un sens défini
 - permettant un compréhension par un humain
 - et un traitement par une machine

[Linked Open Data] : partie du Web sémantique

- Ensemble de ressources Web
- librement accessibles
- ayant des URI *déréférencables*
- contenant des liens vers d'autres ressources

[LOD Cloud view]

Linking Open Data cloud diagram 2014, by Max Schmachtenberg, Christian Bizer, Anja Jentzsch and Richard Cyganiak. <http://lod-cloud.net/>

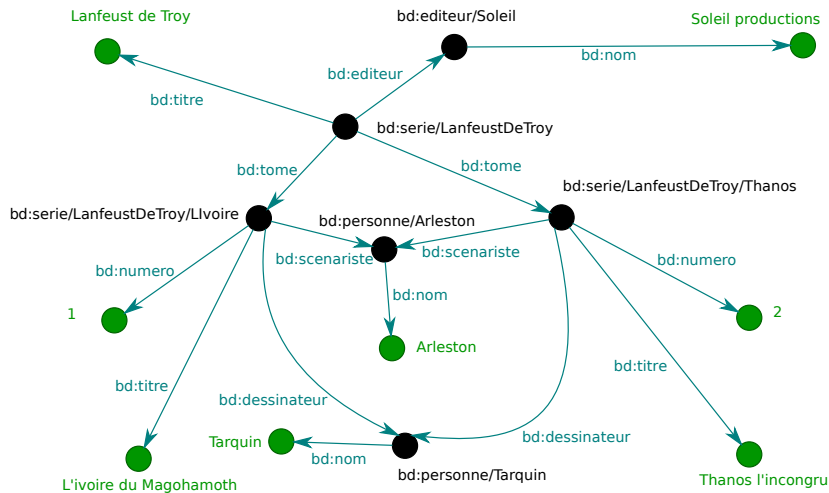
Graphes étiquetés

- Graphes orientés
 - Les sommets et les arêtes sont étiquetés
- Constitue un modèle de données alternatifs aux modèles :
 - Relationnel, semi-structuré, objet
- Permet de représenter aisément des liens entre des choses référencées par un identifiant :
 - Sommet : *chose*
 - Arête : relation entre deux *choses*

RDF : graphes pour le Web sémantique

- Standard du W3C
- Graphes RDF
 - Étiquetés (arêtes et sommets)
 - par des [IRIs] (ressources)
 - par des littéraux (valeurs), uniquement pour les sommets
 - au plus 1 sommet / étiquette
 - pas de max pour le nombre d'arêtes / étiquette
 - l'IRI est symbolique
 - IRIs déréréférencables dans le cadre du *Linked Data*

RDF : exemple



bd :↔http://www.collection.com/bd

Triplet RDF

Description de graphe par des triplets représentant les arêtes

- Sujet
 - Étiquette du sommet de départ
- Prédicat (ou property)
 - Étiquette de l'arête
- Objet
 - Étiquette du sommet d'arrivée

Exemple :

(bd:serie/LanfeustDeTroy, bd:editeur, bd:editeur/Soleil)

Sérialisation : XML

Syntaxe pour représenter des triplets

Élément `rdf:Description`

- Déclaration de triplets ayant pour sujet l'IRI indiquée par l'attribut `rdf:about`
- Attributs/éléments :
 - Espaces de nommage + nom local = IRI du prédicat
- Valeur/attribut `rdf:resource`
 - Objet
 - Littéral/IRI
- Pour les littéraux : `rdf:datatype`

Exemple

```
<?xml version='1.0' encoding='UTF-8'?>
<rdf:RDF
  xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
  xmlns:bd='http://www.collection.com/bd/'>

  <rdf:Description rdf:about='http://www.collection.com/bd/serie/LanfeustDeTroy'>
    <bd:tome rdf:resource='http://www.collection.com/bd/serie/LaufeustDeTroy/LIvoire' />
    <bd:tome rdf:resource='http://www.collection.com/bd/serie/LaufeustDeTroy/Thanos' />
    <rdf:type rdf:resource='http://www.collection.com/bd/serie' />
  </rdf:Description>

  <rdf:Description rdf:about='http://www.collection.com/bd/editeur/Soleil'>
    <bd:nom>Soleil Productions</bd:nom>
  </rdf:Description>

  <rdf:Description
    rdf:about='http://www.collection.com/bd/serie/LaufeustDeTroy/LIvoire'>
    <bd:numero rdf:datatype='http://www.w3.org/2001/XMLSchema#int'>1</bd:numero>
  </rdf:Description>
  ...
</rdf:RDF>
```

Sérialisation : TURTLE

Syntaxe alternative pour RDF

- IRI :
 - `<http://www.collection.com/bd/serie>`
 - `bd:serie`
 - PREFIX `bd: <http://www.collection.com/bd/>`
- Valeur :
 - `'Arleston'`
 - `"2.5"^^xsd:float`
- Triplet :
 - `sujet predicat objet .`
 - `sujet predicat objet ; predicat objet .`
 - `sujet predicat objet, objet ; predicat objet, objet.`

Exemple

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bd: <http://www.collection.com/bd/>

<http://www.collection.com/bd/serie/LanfeustDeTroy> rdf:type
  <http://www.collection.com/bd/serie> .
<http://www.collection.com/bd/serie/LanfeustDeTroy> bd:editeur
  <http://www.collection.com/bd/editeur/Soleil Productions> .

<http://www.collection.com/bd/editeur/Soleil Productions> bd:nom "Soleil Productions" .

<http://www.collection.com/bd/serie/LaufeustDeTroy/LIvoire>
  bd:numero "1"^^xsd:integer ;
  bd:titre "L'ivoire du Magohamoth" ;
  bd:dessinateur <http://www.collection.com/personne/Tarquin> ;
  bd:scenariste <http://www.collection.com/personne/Arleston"> .

<http://www.collection.com/bd/serie/LanfeustDeTroy> bd:tome
  <http://www.collection.com/bd/serie/LaufeustDeTroy/LIvoire> ,
  <http://www.collection.com/bd/serie/LaufeustDeTroy/Thanos 1 incongru> .
```

Sérialisation : JSON-LD

[JSON-LD] : Représentation d'un graphe RDF en JSON

- JSON Object ↔ nœud du graphe RDF
- clés = IRI de prédicat
 - + des clés spéciales (e.g. @id, @type, @context)
- valeurs : objets (= nœuds IRI) ou valeur (= littéraux)

Exemple

```
{
  "@context": {
    "numero": {
      "@id": "http://www.collection.com/bd/numero",
      "@type": "http://www.w3.org/2001/XMLSchema/integer"
    }
  },
  "@id": "http://www.collection.com/bd/serie/LaufeustDeTroy/LIvoire",
  "numero": "1",
  "http://www.collection.com/bd/titre": "L'ivoire du Magohamoth",
  "http://www.collection.com/bd/dessinateur": {
    "@id": "http://www.collection.com/personne/Tarquin"
  },
  "http://www.collection.com/bd/scenariste": {
    "@id": "http://www.collection.com/personne/Arleston"
  }
}
```

Nœuds anonymes (blank nodes)

- Pas des IRIs, ni des littéraux
- Peuvent être utilisés comme des nœuds IRI
- *Intuitivement*, deux nœuds anonymes peuvent être remplacés par un même nœud

Types de données

- Un type T :
 - Espace lexical EL : ensemble de chaînes de caractères
 - c.f. types simples XML
 - Espace (ensemble) de valeurs EV
 - Fonction $L2V(T) : EL \rightarrow EV$

- Exemple : $T =$ Entiers XML Schema
 - EL : chaînes reconnues par $-?[0-9]^+$
 - EV : \mathcal{Z}
 - $L2V(T)$: parsing des entiers

Vocabulaires RDF

- Ensemble d'IRI
- Pouvant être utilisés par une application
- Ayant souvent une définition à minima informelle
- Souvent rattaché à un espace de nommage (en quelque sorte)

Exemple [FOAF] :

- `foaf:firstName` (prénom)
- `foaf:knows` (connaît)

Un vocabulaire particulier : RDF

Préfixe rdf :

`http://www.w3.org/1999/02/22-rdf-syntax-ns#`

- Typage :
`rdf:type rdf:langString rdf:Property`
- Réification :
`rdf:subject rdf:predicate rdf:object`
- Listes :
`rdf:first rdf:rest rdf:value rdf:nil rdf:List`
- Conteneurs : `rdf:_1 rdf:_2`

RDFS, OWL, ...

Règles/axiomes logiques permettant :

- De déduire des triplets additionnels
- D'ajouter des contraintes d'intégrité
 - Seulement sur les types de données en RDFS

Exemple : tous les tomes de série sont des livres

RDFS, OWL : vus comme des schémas

RDF Schema

- Classification des ressources
- Contraintes d'intégrité simples
 - Sur les types primitifs

OWL

- Logique plus riche
- Contraintes d'intégrité plus complexes

RDF Schema (RDFS)

- Système de classes de ressources
 - Avec système de sous-classes
- Description des prédicats
 - Quel sujet, quel objet ?
- Interprétation spécifiques → système d'inférence
 - Déduction de nouveaux faits
Comment les prendre en compte ?

`rdfs:` → `http://www.w3.org/2000/01/rdf-schema#`

`rdf:` → `http://www.w3.org/1999/02/22-rdf-syntax-ns#`

Vocabulaire RDFS

- Typage des propriétés :
 - `rdfs:domain` `rdfs:range`
- Types de base
 - `rdfs:Resource` `rdfs:Literal` `rdfs:Datatype`
`rdfs:Class`
- Relations entre types et propriétés
 - `rdfs:subClassOf` `rdfs:subPropertyOf`
- Conteneurs
 - `rdfs:member` `rdfs:Container`
`rdfs:ContainerMembershipProperty`
- Divers
 - `rdfs:comment` `rdfs:seeAlso` `rdfs:isDefinedBy`
`rdfs:label`

Inférence RDFS

Certains triplets peuvent être considérés comme existants *implicitement* :

- si A est sous-classe de B
- et B est sous-classe de C
- on considère *implicitement* que A est sous-classe de C

Inférence : rendre explicites les triplets implicites

Graphe **saturé** : on ajoute toutes les inférences possibles

- par exemple en itérant les déductions

Classes et instances

- Classe \leftrightarrow ensemble de ressources
- Ressource $R \in$ classe C :
 $R \text{ rdf:type } C$
- Une ressource peut appartenir à plusieurs classes
- Tout ce qui contient une ressource est une classe
on a la règle d'inférence suivante :

$$\frac{R \text{ rdf:type } C}{C \text{ rdf:type rdfs:Class}}$$

Type de données des littéraux

- Les types de littéraux L sont de type `rdfs:Datatype`
- Les types de littéraux L sont des sous-classes de `rdfs:Literal`
- Idéalement, les types des littéraux doivent être respectés par leur interprétation
 - Seule vérification de type de RDFS
 - Une date ne peut pas être un entier
 - Norme un peu plus subtile

rdfs:domain

- Domaine au sens domaine d'une fonction
- Fixe le type T des sujets d'un prédicat P
 P rdfs:domain T
- Inférence

$$\frac{P \text{ rdfs:domain } T \quad \text{et} \quad S \text{ } P \text{ } O}{S \text{ rdf:type } T}$$

rdfs:range

- Dual de rdfs:domain
- Fixe le type T des objets pour une propriété P :
 P rdfs:range T
- Inférence

$$\frac{P \text{ rdfs:range } T \quad \text{et} \quad S \text{ } P \text{ } O}{O \text{ rdf:type } T}$$

Sous-classes

- Classe $C \subseteq$ une classe D :
`C rdfs:subClassOf D`
- Exemple :
`univ:admin rdfs:subClassOf univ:user`
- Inférence :

$$\frac{C \text{ rdfs:subClassOf } D \quad \text{et} \quad E \text{ rdf:type } C}{E \text{ rdf:type } D}$$

Sous-prédicats

- Prédicats P cas particulier d'un prédicat Q :
 P `rdfs:subPropertyOf` Q
- Exemple :
`univ:enseigne rdfs:subPropertyOf univ:participeA`
`univ:inscritA rdfs:subPropertyOf univ:participeA`
- Inférence :

$$\frac{P \text{ rdfs:subPropertyOf } Q \quad \text{et} \quad S \ P \ O}{S \ Q \ O}$$

Transitivité de subClassOf et subPropertyOf

Pour `rdfs:subClassOf` :

$$\frac{C \text{ rdfs:subClassOf } D \quad \text{et} \quad D \text{ rdfs:subClassOf } E}{C \text{ rdfs:subClassOf } E}$$

Pour `rdfs:subPropertyOf` :

$$\frac{P \text{ rdfs:subPropertyOf } Q \quad \text{et} \quad Q \text{ rdfs:subPropertyOf } R}{P \text{ rdfs:subPropertyOf } R}$$

Transitivité de subClassOf et subPropertyOf

Pour `rdfs:subClassOf` :

$$\frac{C \text{ rdfs:subClassOf } D \quad \text{et} \quad D \text{ rdfs:subClassOf } E}{C \text{ rdfs:subClassOf } E}$$

Pour `rdfs:subPropertyOf` :

$$\frac{P \text{ rdfs:subPropertyOf } Q \quad \text{et} \quad Q \text{ rdfs:subPropertyOf } R}{P \text{ rdfs:subPropertyOf } R}$$

RDFS : autres règles

Il existe d'autres règles RDFS, voir :

<http://www.w3.org/TR/rdf11-mt/#entailment-rules-informative>