

# Projet REPEC

Christophe WILLAERT    Nahid OULMI

**25 février 2016**



# 1. Introduction

## 2. Outils du web des données

## 3. Produire les données

## 4. Rendre les données accessibles

## 5. Conclusions

- ▶ Créée en 1997
- ▶ Base de données décentralisée
- ▶ 1 800 000 items recensés dont :
  - ▶ 30 000 auteurs
  - ▶ 720 000 articles (publiés)
  - ▶ 480 000 papiers (working papers)
  - ▶ 15 000 chapitres de livre
- ▶ Métadonnées en *Research Document Information Format* (ReDIF)

- ▶ Objectif : *étudier des réseaux de co-auteurs en Sciences-Économiques*
  - ▶ Auteurs des publications présentes dans RePEC
- ▶ Notre mission :
  - ▶ créer une base de données sémantique de ce réseau d'auteurs

- ▶ Accès à l'ensemble des données de RePEc via des accès FTP/HTTP
  - ▶ `repec.org` liste tous les sites (*archives*)
  - ▶ miroir de l'ensemble des sites mis en place pour le projet
  - ▶ accessible sur `http://test.boulgour.com/repec`
- ▶ Une archive particulière
  - ▶ un répertoire du miroir : `remo/per`
  - ▶ contient un fichier ReDIF par auteur présent dans RePEc
  - ▶ chaque fichier contient les informations sur un auteur
    - ▶ nom
    - ▶ liste des documents auxquels il a contribué
- ▶ C'est à partir de ces fichiers que nous avons travaillé

# Base de données sémantiques ?

- ▶ Web sémantique
  - ▶ un projet de fond des fondateurs du web depuis les années 90 (W3C)
- ▶ L'idée
  - ▶ lier l'information
  - ▶ unifier le web
  - ▶ en faire une bibliothèque géante pour mieux diffuser la connaissance
- ▶ On parle désormais de **web des données** (*Linked Data*)

1. Introduction

**2. Outils du web des données**

3. Produire les données

4. Rendre les données accessibles

5. Conclusions

## Resource Description Framework

- ▶ un cadre de travail pour publier/manipuler des données sur le web
- ▶ recouvre à la fois
  - ▶ un modèle liant un *sujet* à un *objet* via un *verbe*
  - ▶ plusieurs syntaxes de représentations
    - ▶ RDF/XML,
    - ▶ Turtle,
    - ▶ N-Triples
- ▶ Notre contexte
  - ▶ produire des fichiers RDF représentant les liens auteurs-publications
  - ▶ utilisation de la syntaxe N-Triples



# URI : l'identifiant des objets sur le web

- ▶ Uniform Ressource Identifier (URI)
  - ▶ un identifiant unique pour une source web
  - ▶ composant important du web sémantique
- ▶ Uniform Ressource Locator (URL)
  - ▶ identifie une source et permet **en plus** d'y accéder directement
- ▶ URI ne permet pas tant de retrouver la source que de la qualifier

# La syntaxe N-triples

- ▶ La structure du RDF est une séquence (**Sujet – Prédicat – Objet**)
- ▶ Utilisation de la syntaxe N-Triple pour produire cette séquence :  
`<Sujet> <Prédicat> <Objet> .`
  - ▶ *Ne pas oublier le point !*
- ▶ Le sujet et l'objet peuvent être
  - ▶ une URI
  - ▶ une URL (mieux)
  - ▶ un littéral (c'est-à-dire une chaîne de caractères non-identifiée)
- ▶ Le prédicat doit obligatoirement être un URI ou une URL

1. Introduction
2. Outils du web des données
- 3. Produire les données**
4. Rendre les données accessibles
5. Conclusions

# Créer un parseur de fichiers ReDif

- ▶ *Parser* un fichier = le parcourir et en extraire les informations utiles
- ▶ Objectif dans notre cas :
  - ▶ Noms
  - ▶ Prénoms
  - ▶ Domaine d'activité
  - ▶ Ensemble des documents auxquels aura participé l'auteur
- ▶ Pour l'ensemble des 30 000 auteurs enregistrés

Problème : comment intégrer le code ?

# Prendre des fichiers en argument

- ▶ 30 000 fichiers ReDIF à traiter
  - ▶ automatisation de la tâche nécessaire
- ▶ **Bash** = langage de programmation des systèmes Unix
- ▶ Avec Bash et Python nous pouvons prendre en argument une infinité de fichiers

# Organiser l'information sous forme de N-Triples

- ▶ Souvenez-vous des N-Triples : **(Sujet – Prédicat – Objet)**
- ▶ Exemple dans notre cas :
  - ▶ Sujet = Auteur
  - ▶ Prédicat = Document
  - ▶ Objet = Co-Auteur.

C'est simple non ?

- ▶ Nom/Prénom
- ▶ Identifiant unique (URL ?)
- ▶ Dernière connexion
- ▶ Problème de la classification NEP/JEL



# Enrichir ces données

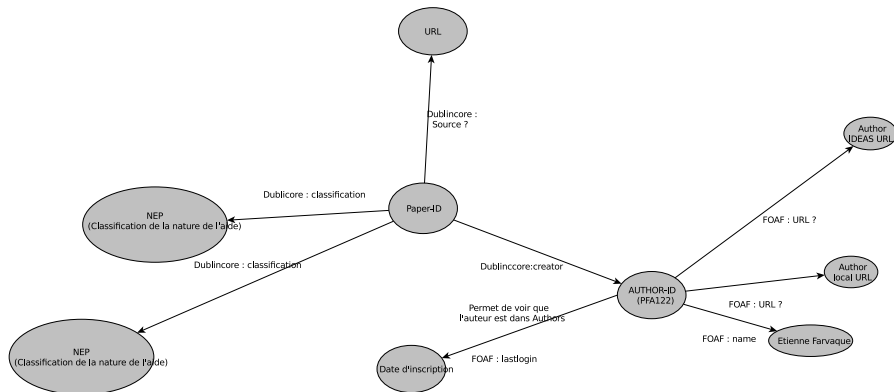


Figure 1:le graphe des liens

1. Introduction
2. Outils du web des données
3. Produire les données
- 4. Rendre les données accessibles**
5. Conclusions

- ▶ Virtuoso est un triplestore
  - ▶ Base conçue pour les données RDF ... mais pas seulement
  - ▶ Complet ... mais complexe
  - ▶ Installé sur le serveur `test.boulgour.com`
- ▶ Objectifs :
  - ▶ Importer nos N-Triples dans la base
  - ▶ Effectuer des requêtes SPARQL

- ▶ Accès :
  - ▶ web via l'outil Conductor (interface graphique via un navigateur)
  - ▶ en ligne de commande via iSQL
- ▶ Syntaxe :
  - ▶ Langage SQL intégré
  - ▶ Langage SPARQL

- ▶ Langage de requêtes pour des données RDF
- ▶ Equivalent au SQL mais pour le web sémantique
- ▶ Standardisé par le W3C depuis 2008
- ▶ Sélectionner le nœuds d'un graphe RDF ainsi que les liens qui les composent.

1. Introduction
2. Outils du web des données
3. Produire les données
4. Rendre les données accessibles
- 5. Conclusions**

# Étapes de travail restantes

- ▶ Etape de visualisation des données
- ▶ Enrichir l'information (JEL/NEP ?)

- ▶ Utilisation des outils du web sémantique
- ▶ Classification JEL/NEP
- ▶ Version de Virtuoso dépassée