



Institut  
Mines-Télécom

# Robots, Crawler, scraper, spider, ..

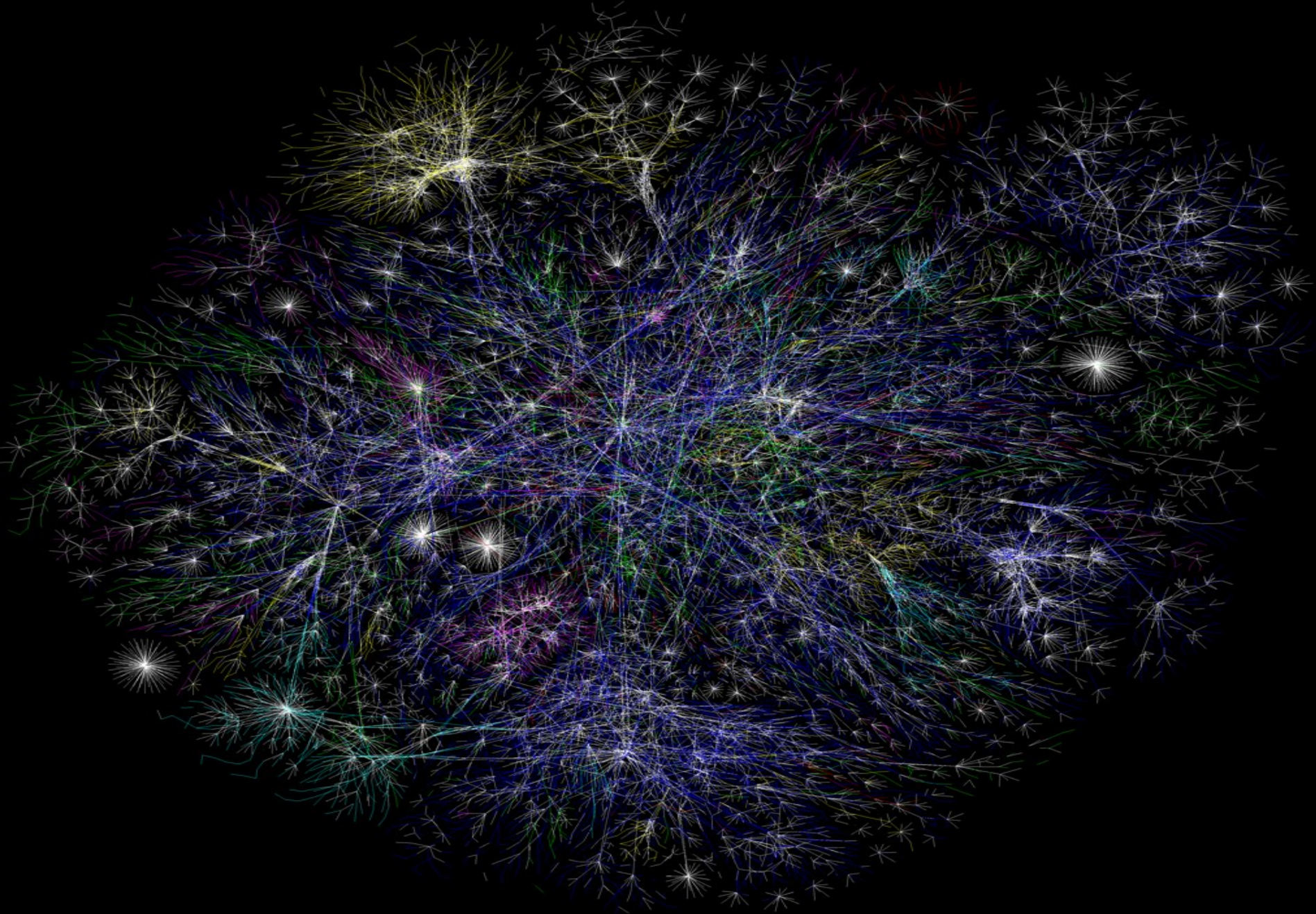
*Ou comment collecter des informations du web*

Télécom ParisTech

Jean-Claude Moissinac – Mai 2018

Avec des éléments de Fabian Suchanek, Pierre Sennellart, Cyril  
Concolato







# Plan

- **Introduction**
- **Sources de données**
- **Principes du crawling**
- **Analyse de pages**
  - Contenu
  - Liens
  - Données structurées
- **Services**
- **Web sémantique**
- **Outils**
- **Conclusion**



# Plan

- **Introduction**
- Sources de données
- Principes du crawling
- Analyse de pages
  - Contenu
  - Liens
  - Données structurées
- Web sémantique
- Outils
- Conclusion



# Objectif

- **Des milliards de pages**
- **Des entrepôts de données**
- **Des services de données**
- **Voir comment obtenir des données du Web**
  - Extraction de données de pages Web
  - Obtention de données disponibles
    - OpenData
    - APIs/services
    - Web sémantique



# Crawler

- **Parcours automatisé du Web**
- **Extraction de données de pages Web**
- **Des actions principales**
  - Choisir des pages à parcourir
  - Parcourir les pages
  - Exploiter les pages obtenues





# Plan

- Introduction
- **Sources de données**
- Principes du crawling
- Analyse de pages
  - Contenu
  - Liens
  - Données structurées
- Web sémantique
- Outils
- Conclusion

# Types de sources de données sur le Web

## Web classique *approfondi dans la section crawler*

- Pages Web HTML
- Pages dynamiques
  - Dont négociation de contenu
- Autres types de contenus
- Pages Web sémantique *abordé plus loin*
- APIs, Web Services
- OpenData



# Web Services: définition

## ■ Un Web Service est

- Un logiciel
- Qui expose des fonctions via un protocole de communication (sur le Web, en général HTTP)
- À l'aide de méthodes d'exploitation standardisées qui en systématisent l'utilisation
  - Indépendance des langages et des systèmes

## ■ Cela permet

- De rendre accessible des services
- De les distribuer
- De composer des services évolués à partir de services élémentaires
- De bénéficier d'une infrastructure réseau bien établie

# Exemples

## REST altitude

■ <http://api.geonames.org/astergdem?lat=45.64&lng=1.85&username=demo>

## REST POI voisins

<http://api.geonames.org/findNearbyPlaceName?lat=45.64&lng=1.85&username=demo&style=full>

<http://api.geonames.org/findNearbyPlaceNameJSON?formatted=true&lat=45.64&lng=1.85&username=demo&style=full>

# REST: Representational state transfer

- **Ni un protocole, ni un format**
- **Un style de mise en œuvre de système distribué**
  - De ce fait, on peut s'inspirer du modèle sans en respecter tous les principes
  - Proposition initiale: thèse de Roy Fielding
- **Principes de base:**
  - Il suffit de connaître l'URI d'un service pour y accéder
  - HTTP fournit toutes les fonctions nécessaires
    - GET, PUT, POST, DELETE
    - Les verbes de HTTP utilisés comme commandes d'actions sur le serveur
  - Fonctionnement sans état
    - Si on est puriste!



# Avantages de REST

## ■ Simplicité de mise en œuvre

- En tout cas pour des développeurs habitués au développement de sites Web dynamiques

## ■ Avantages liés à l'absence d'état

- Moindre charge du serveur => meilleure capacité de réponse
- Facilité de mise au point
- Facilité de répartition de la charge

## ■ Très bonne intégration dans l'univers HTTP

## ■ L'association URI/ressource facilite l'exploitation de caches

# APIs Web

- On parle souvent d'API Web pour les services accessibles sur Internet
- ProgrammableWeb  
<https://www.programmableweb.com/category/all/apis>
- exemple de répertoire d'API
- des centaines d'API recensées: cartographie, réseaux sociaux, traduction...



# Plan

- Introduction
- Sources de données
- **Principes du crawling**
- Analyse de pages
  - Contenu
  - Liens
  - Données structurées
- Web sémantique
- Outils
- Conclusion



## Crawler: rôle

### ■ Crawlers, (Web) spiders, (Web) robots

- Outils qui parcourent automatiquement des pages Web

### ■ Buts

- Sauvegarde/Archives
- Analyse de site
- Consultation offline
- Extraction de données (veille, fusions...)
  - Par analyse au vol
  - Par analyse offline

### ■ Fonctions contraintes

- Limiter aux pages importantes le périmètre du contenu parcouru
- Eviter les pièges à robot
- Faire face à la variabilité des contenus



# Crawler: Parcours

*Voir plus loin 'analyse de page' pour les sources de liens*

## ■ Parcours en profondeur

- URL1->URL2->URL3->URL4...
- On peut se perdre et ne jamais parcourir d'autres parties du web

## ■ Parcours en largeur

- URL1->URL1.1
  - -> URL 1.2
  - -> URL 1.3
  - ...

## ■ Combinaison des deux en mettant une borne sur chaque parcours en profondeur

## ■ Il peut y avoir des boucles

## **Crawler: Procédé général**

- **Sélectionner un ensemble d'URLs à traiter**
- **Récupérer une page en suivant une URL**
  - Maintenir un index des pages visitées
- **Analyser la page**
  - Par exemple avec l'API DOM
- **Sauver le contenu important pour le projet courant**
- **Extraire les URLs de la page et en choisir certaines**
- **Ajouter ces URLs dans la liste des URLs à traiter**
- **... et boucler là-dessus**
  - Soit tant qu'il y a des URLs à traiter
  - Soit que vous ayez récupéré assez de données
  - Soit que vous ayez trouvé l'information que vous cherchez
  - Soit que trop de temps est écoulé
  - ...

# Crawler: Limiter le parcours

## ■ Taille du Web: le Web est infini!

- Pièges à robots
- Pages dynamiquement créées

## ■ Garder un focus sur des pages importantes

- Dans un contexte donné
- Focus sur des domaines DNS
- Focus sur des sujets

## ■ ■ ■ Limiter le parcours (1)

### ■ Focus sur des pages importantes

- Dans un contexte donné
- [Abiteboul et al., 2003]

### ■ Focus sur une liste de domaines DNS

- filtrage simple des URLs

### ■ Focus sur un sujet

- techniques de crawling ciblé [Chakrabarti et al., 1999, Diligenti et al., 2000]
- basé sur la classification de page Web et évaluation/prédiction de l'intérêt d'un lien

## ■ ■ ■ Limiter le parcours (2)

### ■ A l'intérieur d'un domaine

- Limiter la profondeur de visite
- Limiter le nombre pages visitées

### ■ Limiter à une liste de noms de domaine

- Ex: le même que l'URL de départ

### ■ Limiter en définissant une condition d'arrêt

- Ex: termes trouvés dans la page
- Ex: critère de filtrage d'URL

### ■ Limiter à des types de contenus

- Ex: arrêter d'explorer une branche quand on trouve un PDF

# Crawler: Bonnes pratiques

## ■ Eviter le **DOS** (Denial Of Service )

- Attendre de 100ms à plusieurs secondes avant de solliciter à nouveau un domaine déjà sollicité
- Ex: WikiCFP->délai 5s; DBPedia-> délai >10ms

## ■ Respecter les exclusions

- Fichier robots.txt
  - Fichier à la racine d'un serveur qui indique les pages qu'un robot peut parcourir [Koster, 1994]
    - User-agent: \*
    - Allow: /tupeuxyaller
    - Dissallow: /nyvaspas
- Exclusion par meta dans une page

<meta name=« ROBOTS » content=« NOINDEX, NOFOLLOW »>
- Exclusion sur un lien

<a href=« mapagesecrete.html » rel=« nofollow »>...



## Crawler: Traitement parallèle

- **Délais des réponses réseau**
  - => attente des réponses et 'callback'
- **File d'attente par domaine**
  - Et réglages associés: délai, parseur, filtres
- **Traitements parallèles des requêtes**
  - Programmation multi-thread
  - Entrées/sorties asynchrones
- **Utilisation de l'option **keep-alive** (ou HTTP/2) pour diminuer la charge des connexions**
- **Distribution**
  - Map-reduce



# Crawler: contraintes (1)

- **Eviter de revisiter des pages déjà visitées**
  - Si elles n'ont pas été modifiées
  - Peuvent être accédées par des URLs différentes
  - Peuvent conduire à des boucles dans le parcours
- **Prévoir une fréquence de mise à jour**
- **Faire face à la variabilité des contenus Web**
  - Types de ressources Ex type MIME
  - Versions des normes (HTML...) et non respect des normes (TAGSOUP)
- **Définir des méthodes d'extraction d'information**
- **Tenir compte des limites placées par les serveurs**

## Crawler: Contraintes (2)

- **Eviter les pièges à robots**
- **Identifier les pages mises à jour**
- **Une méthode courante:**
  - le hachage d'URL
    - = calcul d'une valeur numérique représentative de l'URL
    - Quand on trouve deux URL associées à la même valeur numérique, on approfondit la comparaison
  - Le hachage de contenu
    - Même principe, mais sur le contenu de la page
    - Pour détecter une page où on arrive par plusieurs URLs
- **Difficulté: pages presque identiques**
  - Ex: à la l'heure près pour une page qui affiche l'heure

*Note: des fonctions de hashage sont disponibles dans la plupart des langages*

# Crawler: éviter les pages visitées

## estampille temporelle

### ■ HTTP Timestamping: 2 mécanismes, potentiellement utilisés avec chaque requête

- **entity tags**
  - identificateur unique du document; change si le document change
  - Peut être utilisé comme sélecteur dans la requête (If-Match)
- **modification dates**
  - Peut être utilisé comme sélecteur dans la requête (If-Modified-Since)

If-Modified-Since: Wed, 15 Oct 2008 19:40:06 GMT

ETag: "497bef-1fcb-47f20645"

Last-Modified: Tue, 01 Apr 2008 09:54:13

Souvent fournis pour les contenus statiques

Rarement fournis pour les contenus dynamiques

# Crawler: HTTP Cache-Proxy

- Deux autres indications de 'fraicheur' d'un contenu, pour les caches et les proxies:

```
Cache-Control: max-age=60, private Expires: Tue, 01 Apr 2008 13:25:55 GMT
```

- max-age: délai maximum en secondes où un document est garanti rester à jour
  - Expires: date à laquelle un document sera considéré comme dépassé
  - Souvent fourni...
  - ... Mais avec 0 ou un délai d'expiration très court.
  - ⇒ information de faible portée
- 
- Données meta et autres dans le contenu des pages

### ■ Des fichiers autres que HTML peuvent avoir une information de version et/ou de date

- PDF, documents Open Office, etc.: des metadonnées contiennent des informations de date de création et de dernière modification.
- RSS feeds: contiennent des estampilles temporelles fiables
- Images, Sons: EXIF metadata (ou similaire). Pas toujours exploitable
- Sitemaps

# Crawler: content type negotiation

- **Navigateur et serveur se comportent différemment en fonction du type de contenu**
  - Mise en page d'un contenu HTML
  - Affichage brut d'une page de texte
  - Affichage d'une image
- **Le client peut indiquer les types qu'il préfère**
- **MIME** est le standard de déclaration de type de contenu
  - Exemple: image/jpeg, text/plain, text/html, application/xhtml+xml, application/pdf
- **Les documents texte et HTML doivent aussi être accompagnés d'une indication sur le jeu de caractère utilisé**

## Exemple

```
HTTP/1.1 200 OK
Content-Type: text/html;
charset=UTF-8
```

# Client and server identification

- Web clients and servers can identify themselves with a character string
- Useful to serve **different content** to different browsers, detect robots. . .
- . . . but any client can say it's any other client!
- Historical confusion on naming: all common browsers identify themselves as Mozilla!

## Example

User-Agent: Mozilla/5.0 (X11; U; Linux x86\_64; fr; rv:1.9.0.3)  
Gecko/2008092510 Ubuntu/8.04 (hardy) Firefox/3.0.3

Server: Apache/2.0.59 (Unix) mod\_ssl/2.0.59 OpenSSL/0.9.8e

PHP/5.2.3

7 October 2013



Institut Mines-Télécom

Pierre Senellart



# Crawler: Doublons

- **Détecter les doublons ou les presque doublons**
  - Pour éviter l'indexation multiple d'un contenu
- **Cas trivial: même ressource issue de même URL**
  - Détecté avec version canonique de l'URL
- **Détection de contenu strictement identique**
  - Comparaison par hachage
- **Contenus presque identiques**
  - Date, Conseil du jour, Publicité...
  - Personnalisation Ex: nom du visiteur connecté
  - => plus compliqué à détecter
  - => on peut essayer détecter et éliminer une partie du contenu variable

# Crawler: Doublons stricts et hachage

- **Détection de contenu strictement identique**
  - Comparaison par hachage
- Une **fonction de hachage** est une fonction mathématique transformant un objet numérique (nombres, chaîne de caractère, binaire,...) en un nombre pseudo-aléatoire de taille fixe
- Par exemple, en Java, pour une chaîne
  - $\sum s[i] * 31^{n-i-1} \bmod 32$

# Crawler: pages très proches

## ■ Distance d'édition (edit distance) Ex Levenshtein

- Compter le nombre de modifications élémentaires – ajout, suppression, échange- pour passer d'une chaîne de caractère à l'autre
- Ne passe pas à l'échelle sur un très grand nombre de documents où il faudrait comparer toutes les paires possibles

## ■ Shingles

- Principe: 2 documents sont similaires si ils partagent un grand nombre de k-grams (suite d'éléments de longueur k)
- Exemple: I like to watch the sun set with my friend.
- My friend and I like to watch the sun set.
- $S = \{i \text{ like}, \text{ like to}, \text{ to watch}, \text{ watch the}, \text{ the sun}, \text{ sun set}, \text{ with my}, \text{ my friend}\}$   $T = \{\text{set with}, \text{ with my}, \text{ friend and}, \text{ and i}\}$



Institut  
Mines-Télécom

# Analyse de page Web



# HTML (HyperText Markup Language) [W3C, 1999]

- normalized by the W3C (World Wide Web Consortium) formed of industrials (Microsoft, Google, Apple. . . ) and academic institutions (ERCIM, MIT, etc.)
- **open** format: possible processing by a wide variety of software and hardware
- **text** files with **tags**
- describes the **structure** and **content** of a document, focus on **accessibility**
- (theoretically) no presentation information (this is the role of CSS)
- no description of dynamic behaviors (this is the role of server-side languages, JavaScript, etc.)



# The HTML language

- HTML is a language alternating text and **tags** ( `<blabla>` or `</blabla>` )
  - Tags allow structuring each part of a document, and are used for instance by a browser to lay out the document.
- HTML files
  - are structured in two main parts: the header `<head> ... </head>` ) and the body `<body> ... </body>` )
- In HTML, blanks (spaces, tabs, carriage returns) are generally equivalent and only serve to delimit words, tags, etc. The number of blanks does not matter.



# Tags

- Syntax: (opening and closing tag)

```
<tag attributes>content</tag>
```

or (element with no content)

```
<tag attributes>
```

**tag** keyword referring to some particular HTML **element** **content** may contain text and other tags

**attributes** represent the various parameters associated with the element, as a list of `name="value"` or `name='value'`, separated by spaces (quotes are not always mandatory, but they become mandatory if value has “exotic” characters)





# Tags

- Names of elements and attributes are usually written in lowercase, but `<head>` and `<HeAd>` are equivalent.
- Tags are opened and closed in the right order ( `<b><i></i></b>` and not `<b><i></b></i>` ).
- Strict rules specify which tags can be used inside which.
- Under some conditions, a tag can be implicitly closed, but these conditions are complex to describe.
- `<!--foobar-->` denotes a comment, which is not to be interpreted by a Web client.

# The different versions of HTML

- HTML 4.01 (1999) strict (as described earlier) and transitional

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
```

```
"http://www.w3.org/TR/html4/strict.dtd">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
```

```
"http://www.w3.org/TR/html4/loose.dtd">
```

- XHTML 1.0 (2000) strict and transitional
- XHTML 1.1 and XHTML 2.0: mostly a failure, unusable and unused in today's Web
- HTML5: future standard, in final development, partly implemented, continuously updated

```
<!DOCTYPE html>
```

# Tag soup

- A lot of HTML documents on the Web date back from before HTML 4.01
- In practice: many Web pages do not respect any standards at all (with or without doctype declarations)  $\Rightarrow$  browsers do not respect these standards  $\Rightarrow$  **tag soup!**
- When dealing with pages from the real Web, necessary to use all sorts of heuristics to interpret a Web page.



# HTML vs XHTML

- XHTML: an XML format
- Tags without content `<img>` , are written `<img />` in XHTML.
- Some elements can be left unclosed in HTML  
( `<ol> <li> one <li>two </ol>` ), but closing is mandatory in XHTML.
- Attribute values can be written without quotes  
( `<img src=toto.png alt=toto>` ) in HTML, quotes are required in XHTML.
- Element and attribute names are not case-sensitive in HTML  
( `<HTML laNg=fr>` ), but are in XHTML (everything must be in lowercase).
- Attributes `xmlns` and `xml:lang` on the `<html>` tag in XHTML. And
- some other small subtleties. . .



# Structure of a document

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML
4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd
```

- The doctype declaration `<!DOCTYPE ...>` specify which HTML version is used.
- The language of the document is specified with the `lang` attribute of the main `<html>` tag.



# Header

- The **header** of a document is delimited by the tags

```
<head> ... </head>.
```

- The header contains **meta-informations** about the document, such as its title, encoding, associated files, etc. The two most important items are:

The character set of the page, usually at the **very beginning** of the header

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

The title of the page (the only required item inside the header). This is the information displayed in the title bar of Web browsers.

```
<title>My great website</title>
```



# The body of a HTML document

- `<body> ... </body>` tags delimit the **body** of a document.
- The body is **structured** into sections, paragraphs, lists, etc.
- 6 tags describe **sections**, by decreasing order of importance:
  - `<h1>`Title of the page`</h1>`
  - `<h2>`Title of a main section`</h2>`
  - `<h3>`Title of a subsection`</h3>`
  - `<h3>`Title of a subsubsection`</h3>`
  - ...
- `<p> ... </p>` tags delimit **paragraphs** of text. All text paragraphs should be delimited thusly.
- Directly inside `<body> ... </body>` can only appear **block** elements: `<p>` , `<h1>` , `<form>` , `<hr>` , `<ul>` , `<table>` ... in addition to the `<div>` tag which denotes a block without precise semantics.

# Character sets

**Unicode: character repertoire**, assigning to each character, whatever its script or language, an integer number.

## Examples

A	→ 65	£	→ 949
é	→ 233	ℵ	→ 1488

**Character set:** concrete method for representing a Unicode character.

## Examples

iso-8859-1	11101001	only for some characters
utf-8	11000011 10101001	
utf-16	11101001 00000000	

**utf-8** has the advantage of being able to represent all Unicode characters, in a way compatible with the legacy **ASCII** encoding.





# Sources de liens dans une page HTML

## ■ Hyperliens

- `<a href=« ... »>`

## ■ Media

- `<img src=« ... »>`
- `<embed src =« ... »>`
- `<object data=« ... »>`

## ■ Frames

- `<frame src=« ... »>`
- `<iframe src=« ... »>`

## ■ Scripts. Exemples:

- Différentes formes d'appels AJAX
- `Window.open(« ... »)`

## ■ Sitemaps (cf.sitemaps.org)

# Liens

- Ce qui différencie les pages Web (pages hypertext) from de simples documents: **les liens!**
- Introduits avec `<a> ... </a>`
- Un lien peut envoyer vers:
  - Un document sur un autre serveur
  - Une autre document sur le même serveur
  - Une autre partie du même document

```
<a href="http://www.cnrs.fr/">  
    
</a>
```

```
<a href="bio/indexbioinfo.html">Bioinformatics</a>
```



# Analyse de page

## ■ Du très empirique...

- Ex: Récupérer le contenu du 1<sup>er</sup> paragraphe du 3eme div
- Pb: fragile dans le temps si la page évolue

## ■ Au très sophistiqué

- Identification de régularités sur des séries de page et définition automatisée de règles d'analyse
- Analyses linguistiques ou sémantiques

## ■ Cas particuliers

- Recherche de suites de mots à partir d'un répertoire de référence
- Recherche d'un 'motif' à partir d'expressions régulières (regular expression)

# Analyse d'une page (exemple)

■ <http://bibliothequenumerique.tv5monde.com/livre/>



TV5MONDE

**L'AVARE** (1668)

Molière

“ Notre phrase préférée :  
Donner est un mot pour qui il a tant d'aversion, qu'il ne dit jamais : « Je vous donne », mais « Je vous prête le bonjour ». ”

Genre : Théâtre

Résumé :

Harpagon n'aime rien plus que l'argent, pas même, Marianne, qu'il projette pourtant d'épouser. Mais il se trouve que son fils, Cléante, ignorant des projets de son père, aime aussi Marianne et est aimé d'elle. Par ailleurs sa fille, Elise, qu'Harpagon destine au seigneur Anselme (parce qu'il la prend sans dot !), aime Valère. Cette seconde intrigue se complique d'une troisième : Valère, qui s'est fait engager par Harpagon comme intendant pour être auprès d'Elise, est accusé par celui-ci de lui avoir volé une cassette contenant une grosse somme d'argent ... Mais, dans les comédies, tout s'arrange à la fin !

Les premiers mots :

« VALERE - Hé quoi ! charmante Elise, vous devenez mélancolique, après les obligeantes assurances que vous avez eu la bonté de me donner de votre foi ? »

VOIR LA FICHE DE L'AUTEUR

TÉLÉCHARGER CE LIVRE

105 liens

<a> 50

<img> 29

<script> 26

Tous n'ont pas besoin  
d'être suivis



Institut  
Mines-Télécom

# Outils



# Outils pour capter des pages

## ■ Outils tout prêt

- WinHtTrack (php)
- Selenium (automatisation du Web)
- Bixo (s'appuie sur Hadoop-Map Reduce)
- Heritrix
- Apache Nutch

## ■ Développement

- Scrapy (python), moteur de crawl
- Crawler4j (java)

# Extracteur de site, crawler

- **Exemple: WinHTTrack**
- **Aspire des pages reliées à une ou plusieurs pages de départ données**
- **Obtient une vision statique du site**
  - État des pages générées à un instant donné
- **Usage**
  - Sauvegarde
  - Analyse de site
  - Consultation offline
  - Extraction de données par analyse offline

## Exemple PHP (principe): récupérer une page

```
<?php
```

```
$ch = curl_init("http://www.example.com/page1");  
$fp = fopen("example_homepage.txt", "w");
```

```
curl_setopt($ch, CURLOPT_FILE, $fp);  
curl_setopt($ch, CURLOPT_HEADER, 0);
```

```
curl_exec($ch);  
curl_close($ch);  
fclose($fp);  
?>
```



# Outils pour analyser les pages

## ■ DOM API

- Tous langages
- Les pages doivent être ‘bien formées’...
- ... sinon utiliser [HTML Tidy](#)

## ■ XSLT

## ■ BeautifulSoup (python)

## ■ [Boilerpipe](#) (java)

## ■ Apache Tika (Java)

## ■ Jtidy (java)

## ■ [Readability](#)

- Cibl  uniquement sur le texte des pages

## Exemple PHP (principe): trouver les <a>

```
<?php
$file = "test.html";
$doc = new DOMDocument();
$doc->loadHTMLFile($file);

$elements = $doc->getElementsByTagName('a');

if (!is_null($elements)) {
    foreach ($elements as $element) {
        // ici trouver l'attribut href et l'ajouter dans une liste...
    }
}
?>
```

# Trouver des informations dans les pages

- **Analyser (parser) la page pour y trouver des motifs**
  - Par exemple avec des expressions régulières
- **Parcourir la page avec l'API DOM**
- **Transformer la page avec XSLT**
  
- **Trouver des informations structurées dans les pages qui en ont**
  - Json-ld, RDFa, microformat
  - *Abordé plus loin*

# XSLT pour analyser les pages

- **XSLT est un langage de transformation de document XML en autre chose**
  - Par exemple, un autre document XML
  - Spécialisé pour cette tâche
  - Basé sur la désignation d'une portion de document et la spécification de la transformation à lui appliquer
- **Xpath pour désigner un « chemin » vers une portion de contenu**
- **Difficultés**
  - Trouver un Xpath robuste à de légères variations dans le document
  - Les pages Web ne sont généralement pas du XML bien formé

## Exemple sur WikiCFP

■ <http://www.wikicfp.com/cfp/servlet/event.showcfp?eventid=63203&copyownerid=98168>

```
...<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
...  version="2.0">
...
  <xsl:template match="/">
    <xsl:apply-templates select="html/body/div/center/table/tr/td/table/tr/td/table/tr/td/table/tr"/>
  </xsl:template>

  <xsl:template match="tr">
    <xsl:variable name="itemWhere"><xsl:value-of select="th"/></xsl:variable>
    <xsl:if test="$itemWhere='Where'">
      <xsl:value-of select="td"/>
    </xsl:if>
    <xsl:apply-templates/>
  </xsl:template>

  <xsl:template match="@*|node()">
  </xsl:template>

</xsl:stylesheet>
```



Institut  
Mines-Télécom

# Web sémantique et Web des données



# Idées du Web Sémantique

## ■ Rendre les données du Web exploitables

- Par les humains
- Par des machines
- *(De préférence par les deux)*

## ■ Pour cela, il faut

- Marquer/typer des données dans les pages du web
- Définir une méthode pour publier des données sur le Web

## ■ Résultats attendus

- Faire traiter des données par des machines
- Tisser des liens entre des données dispersées

# Web Sémantique

- Définir une infrastructure qui permet aux machines d'opérer sur les données en les 'comprenant'
- C'est-à-dire:
  - Permettre à des machines d'opérer sur les données d'autres machines
  - Assurer l'interopérabilité
  - Permettre aux données de se décrire elles-mêmes
  - Permettre aux machines de raisonner sur les données
  - Permettre aux machines de fournir des réponses à des requêtes 'sémantiques'
- Méthode: s'appuyer sur le WWW pour rendre les données disponibles d'une façon standard, notamment dans les pages web





# Marquage sémantique

<https://developers.google.com/search/docs/guides/intro-structured-data>

- RDFa
- Microdata
- Json-Ld



# Schema.org

- [Event](#), [Organization](#), [Person](#), [Product](#), [Review](#), [AggregateRating](#), [Offer](#)
- **schema.org**



# Représentation des connaissances

# Granules de connaissances

- Les triplets RDF
- (sujet)(prédicat)(objet)
- **Sujet:** l'entité sur laquelle porte la connaissance
- **Prédicat:** l'affirmation qu'on fait sur le sujet; une propriété applicable au sujet
- **Objet:** valeur qu'on associe au prédicat (valeur de la propriété)

**L'ensemble constitue une connaissance sur le sujet**



# Resource Description Framework

## RDF

### ■ Resource

- Pages, images, vidéo, données...
- Accessibles par une URI (ex: <http://monsite.fr/...>)

### ■ Description

- Propriétés et relations de la ressource

### ■ Framework

- Modèle (simple), langage, syntaxes pour ces descriptions

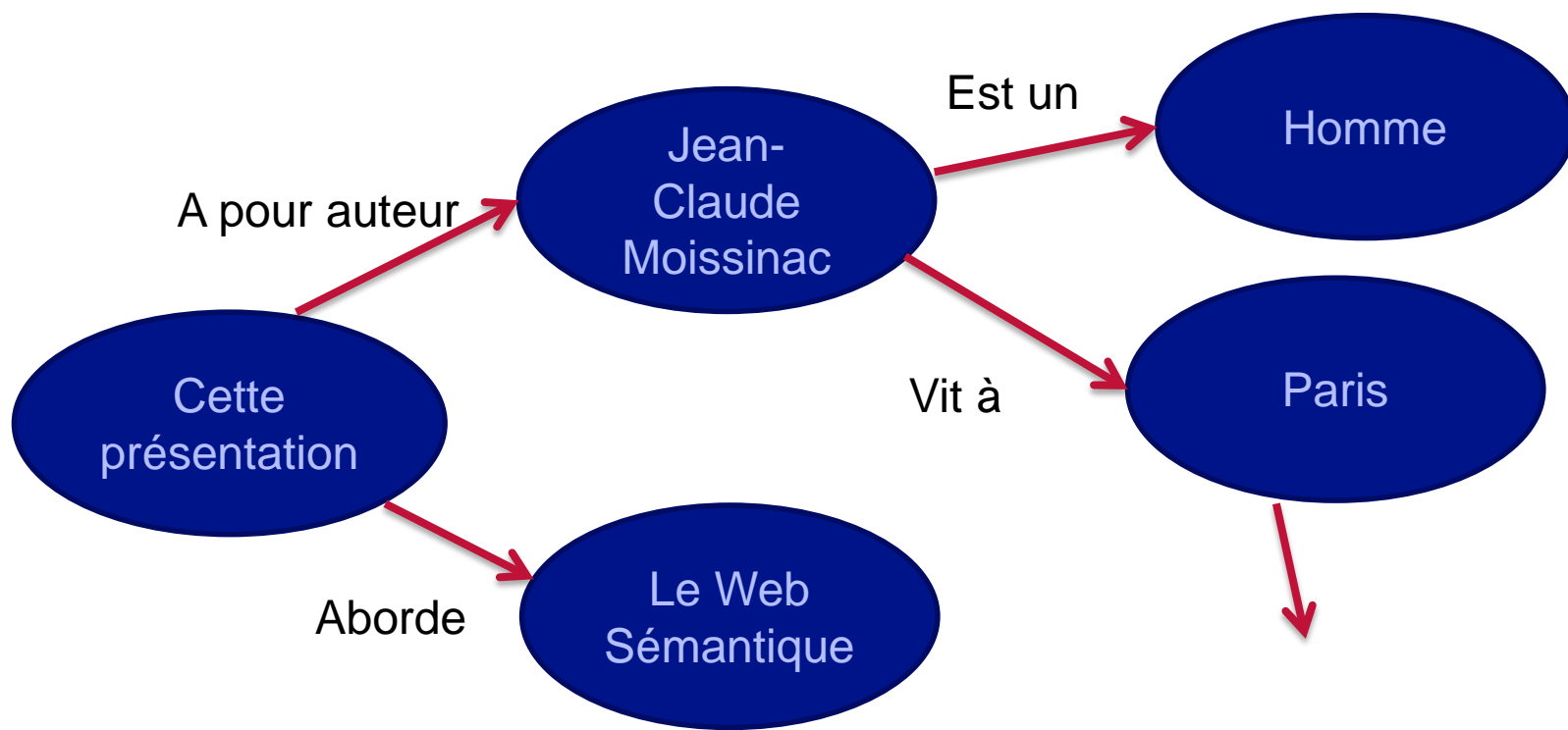


# RDF, le modèle

- Décrire tout ce qu'on peut par des triplets
- (**sujet**, **prédictat**, objet)
- Cette présentation a pour auteur Jean-Claude Moissinac et aborde le Web Sémantique
- (**cette présentation**, **a pour auteur**, Jean-Claude Moissinac)
- (**cette présentation**, **aborde**, le Web Sémantique)

# RDF définit des graphes

- Un ensemble de triplets RDF peut être vu comme un graphe orienté et étiqueté



# Utilisation d'URIs

- Les URIs sont uniques par construction
- Si deux entités (machines, personnes...) utilisent des URIs différentes, il se peut qu'elles traitent de la même chose
- Si deux entités (machines, personnes...) utilisent une même URI, il est sûr qu'elles traitent de la même chose



# Construction d'URLs – modèle des URLs

- Modèle des URLs
- <protocole>:<domaine>/<chemin d'identification>
- En pratique, pour les données liés:
- <http://monsupersite.com/data/geo/Paris>
- Protocole http
- Domaine possédé par un propriétaire de nom
- Chemin désignant de façon unique un concept

*(d'autres modèles d'URLs existent)*

## Construction d'URIs – modèle des URLs (2)

- **Le chemin désignant de façon unique un concept**
  - Peut être totalement abstrait
  - Peut ne pas amener à une page web
- <http://www.geonames.org/2988507/>
- <http://fr.dbpedia.org/resource/Paris>
- <http://dbpedia.org/resource/Paris>
- <http://yago-knowledge.org/resource/Paris>
- **Mais les recommandations (Linked Data)**
  - Indiquent notamment comment ramener à une page web (demo DBPedia)



# RDF dans les pages Web

# Exploiter du RDF dans des pages Web?

## Paris fête le 14 juillet

### SOMMAIRE

BALS DANS LES CASERNES DE  
POMPIERS

DÉFILÉ MILITAIRE SUR L'AVENUE  
DES CHAMPS-ÉLYSÉES

FEU D'ARTIFICE DU 14 JUILLET

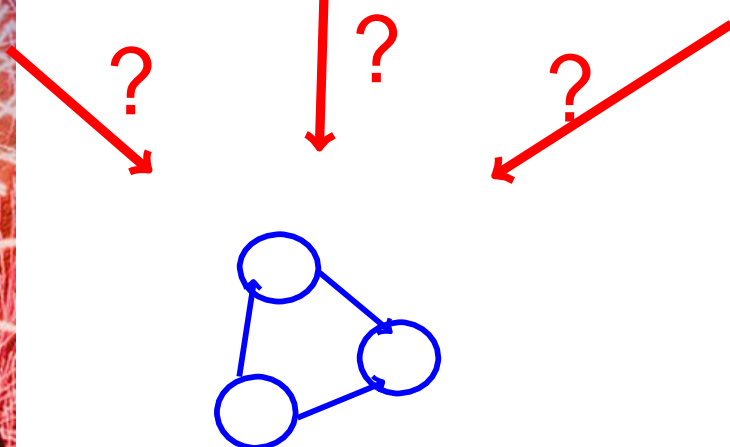
LES FRANCILIENS ACCUEILLEN  
LEURS SOLDATS

LES BONS PLANS DE LA  
JOURNÉE DE FÊTE NATIONALE



### Basic Specifications

|               |  |
|---------------|--|
| Resolution:   | 8.00 Megapixels                          |
| Sensor size:  | 1/2.5"                                   |
| Lens:         | 5.00x zoom<br>(35-175mm eq.)             |
| Viewfinder:   | LCD                                      |
| ISO:          | 80-3200                                  |
| Shutter:      | 2-1/1000                                 |
| Max Aperture: | 3.5                                      |
| Dimensions:   | 3.6 x 2.3 x 0.9 in.<br>(92 x 59 x 22 mm) |
| Weight:       | 6.1 oz (172 g)<br>includes batteries     |
| MSRP:         | \$400                                    |
| Availability: | 03/2007                                  |



### Homepage



### Gerhard Weikum

[Max-Planck-Institut für Informatik](#)  
[Department 5: Databases and Information Systems](#)  
[Building E1.4, Room 402](#)  
[Campus E1.4](#)  
[66123 Saarbrücken](#)  
[Germany](#)

**Email:** [weikum@mpi-inf.mpg.de](mailto:weikum@mpi-inf.mpg.de)  
**Phone:** +49 681 9325 500  
**Fax:** +49 681 9325 599

# Programmes scolaires

- <http://127.0.0.1/givingsense.eu/programmes/HistoireCollegeAout2008/Rich/>
- <http://givingsense.eu/programmes/>

Programmes scolaires avec étiquetage sémantique -

Source:

<http://givingsense.eu/programmes/>

Système d'annotation créé par:

<http://moissinac.wp.mines-telecom.fr/>

Programmes d'histoire-géographie-éducation civique du collège:

| Pdf                  | HTML                 | Annot.ontologie...   | ...Turtle   | Annot.par SpotLight  | Enrich. |
|----------------------|----------------------|----------------------|---|----------------------|---------|
| <a href="#">6eme</a> | <a href="#">6eme</a> | <a href="#">6eme</a> |    | <a href="#">6eme</a> |         |
| <a href="#">5eme</a> | <a href="#">5eme</a> | <a href="#">5eme</a> |  | <a href="#">5eme</a> |         |
| <a href="#">4eme</a> | <a href="#">4eme</a> | <a href="#">4eme</a> |  | <a href="#">4eme</a> |         |
| <a href="#">3eme</a> | <a href="#">3eme</a> | <a href="#">3eme</a> |  | <a href="#">3eme</a> |         |

RDFa est une syntaxe pour annoter des pages HTML avec du RDF

<https://rdfa.info/>

**<div>Jean Mois<br>**

**Chercheur en dessin animé 1957-<br>**

**Roubaix, Nord**

**</div>**

[RDFa Lite](#)

## Définir le vocabulaire

Localement, tous les termes associés à un noeud HTML vont venir du vocabulaire défini dans 'vocab'.

```
<div vocab="http://schema.org/">
```

Jean Mois<br>

Chercheur en dessin animé 1957-<br>

Roubaix, Nord

```
</div>
```

## Définir le sujet

Toutes les propriétés associées au nœud HTML ont pour sujet l'entité désignée dans 'resource'.

```
<div vocab="http://schema.org/"  
resource="http://moissinac.wp.mines-telecom.fr/">
```

JC. Moissinac<br>

Chercheur en dessin animé 1957-<br>

Roubaix, Nord

</div>



## Définir un type

Le type du sujet est donné par 'typeOf'.

```
<div vocab="http://schema.org/"  
resource="http://moi..." typeOf="Person">
```

Jean Mois<br>

Chercheur en dessin animé 1957-<br>

Roubaix, Nord

```
</div>
```

*Triplet*

```
<http://moissinac...> rdf:type <http://schema.org/Person> .
```

# Definir un fait avec une valeur

Un tag avec 'property' defini un fait sur le sujet courant; la valeur associée est celle du atg

```
<div vocab="http://schema.org/"  
  resource="http://moi..." typeOf="Person">  
  <span property="name">Jean Mois</span><br>
```

Chercheur en dessin aimé 1957-<br>

Roubaix, Nord

</div>

*Triplet*

<<http://moi...>> <<http://schema.org/name>> "Moissinac".

### Standards similaires à RDFa:

- Microdata
- Json-Ld
  - Désormais recommandé par Google

# RDFa Exemple

## Kontakt

[Fabian M. Suchanek](#)

[Max-Planck Institut für Informatik](#)

Otto Hahn Research Group "[Ontologies](#)", office 414

Campus E1.4

66123 Saarbrücken

Germany

E-Mail: [Vorname@Nachname.name](mailto:Vorname@Nachname.name)

URL: <http://suchanek.name>



## RDFa Validator

Donne



@prefix og: <<http://ogp.me/ns#>> .

@prefix rdfa: <<http://www.w3.org/ns/rdfa#>> .

@prefix schema: <<http://schema.org/>> .

<[http://suchanek.name/about/index\\_e.php](http://suchanek.name/about/index_e.php)> rdfa:usesVocabulary schema: .

<<http://suchanek.name/fabian>> a schema:Person; og:description "leader of the Otto Hahn Research Group"; og:image <<http://suchanek.name/about/fabian.jpg>>; og:title "Fabian M. Suchanek";

schema:address [ a schema:PostalAddress;

schema:addressCountry <<http://yago-knowledge.org/resource/Germany>>;

schema:addressLocality "Saarbrücken";

schema:postalCode "66123"; schema:streetAddress "Campus E1.4" ];

schema:image <<http://suchanek.name/about/fabian.jpg>>;

schema:jobTitle "leader of the Otto Hahn Research Group"; schema:name "Fabian M. Suchanek";

schema:url <<http://suchanek.name>>;

schema:worksFor <<http://mpii.de>> .



# Demo

- <http://www.w3.org/2012/pyRdfa/Validator.html>
- <https://rdfa.info/play>
  - <http://givingsense.eu/foaf/moissinacRdfa.htm>
- <https://developers.google.com/structured-data/testing-tool/>

# Marquage utilisé par les moteurs de recherche

[Sony Cyber-shot DSC-T100 review - Digital Camera - Trusted ...](#)

[www.trustedreviews.com](#) > [Cameras](#) > [Digital Camera](#) ▼

★★★★★ Rating: 8/10 - Review by Cliff Smith

Feb 5, 2011 - Sony Cyber-shot **DSC-T100** Digital Camera review: Is Sony's flagship compact camera worth the asking price?

Demo

Aller sur <http://www.trustedreviews.com/lenovo-p2-review>

Copier ce lien dans <https://search.google.com/structured-data/testing-tool>

# JSON-LD: exemple

```
<script type="application/ld+json">
{
  "@context": "http://schema.org",
  "@type": "Organization",
  "url": "http://www.your-company-site.com",
  "contactPoint": [{
    "@type": "ContactPoint",
    "telephone": "+1-401-555-1212",
    "contactType": "customer service"
  }]
}
</script>
```

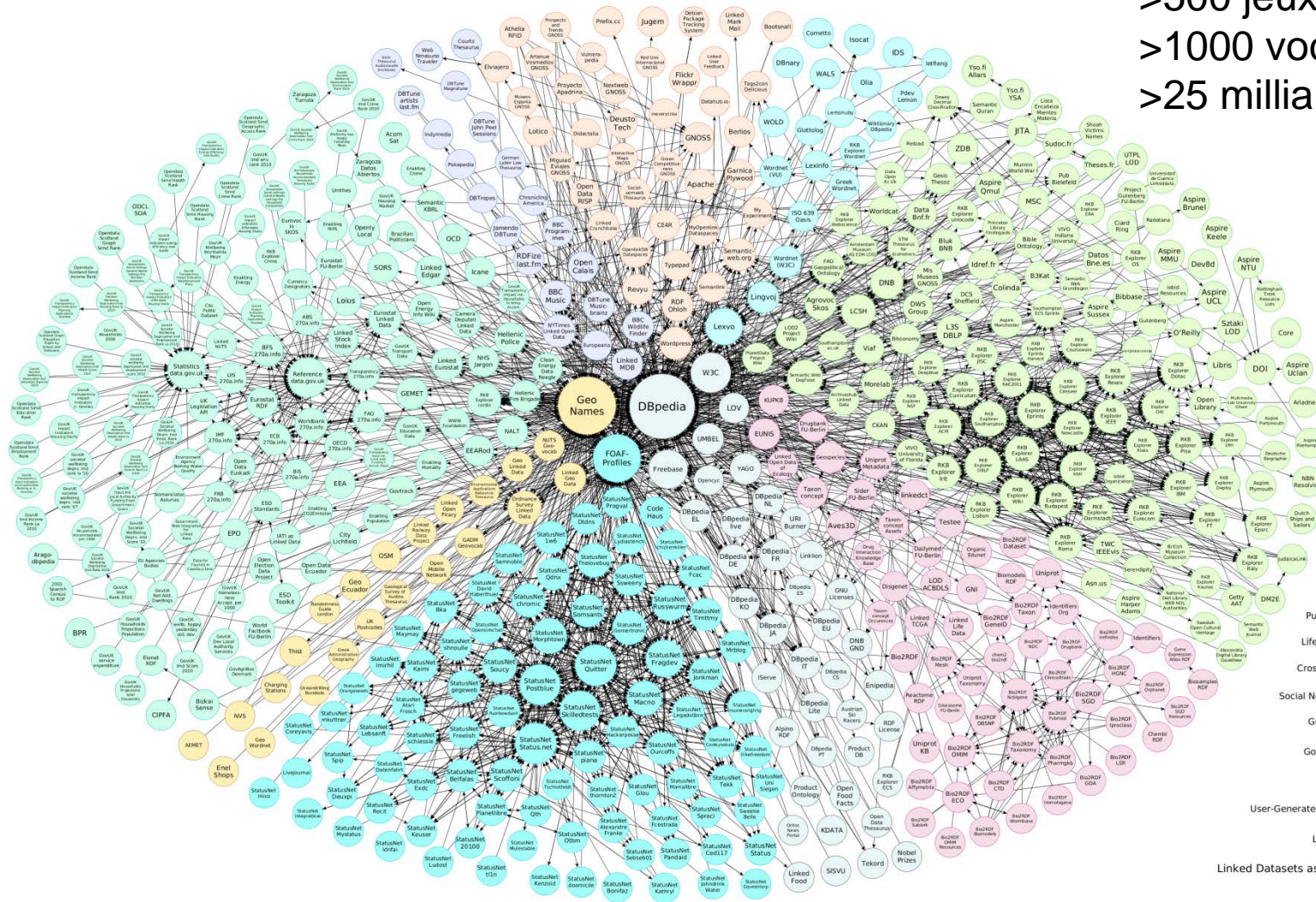


# Web des données



## Linked Open data

- >500 jeux de données
- >1000 vocabulaires
- >25 milliards de triplets



Linked Datasets as of April 2014



# Bases de connaissances extraites du Web

- **DBPedia**
- **Yago**
- **Wikidata**

- Initiative pour tirer une représentation sémantique du contenu de Wikipedia
- Défini des gabarits d'extraction de faits (triplets sujet, prédicat, objet) à partir de portions de page de DBPedia
- L'extraction est automatique sur la base de ces gabarits
- Demo

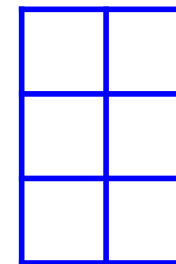
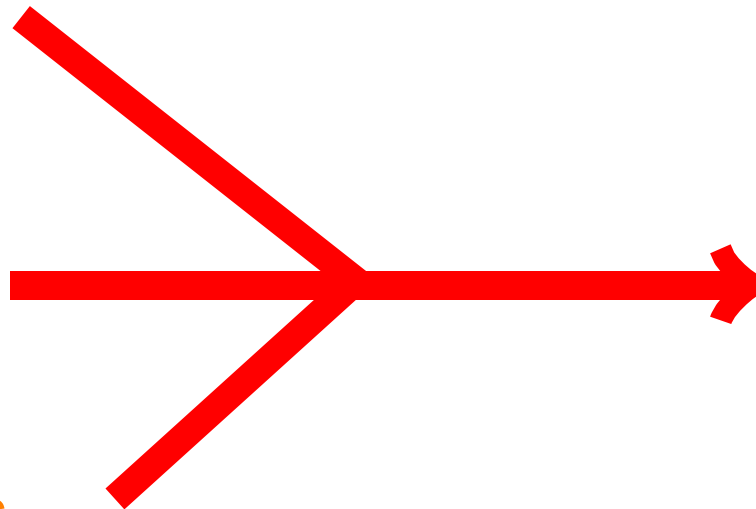
# YAGO

Le projet YAGO project extrait des informations de Wikipedia et d'autres sources.



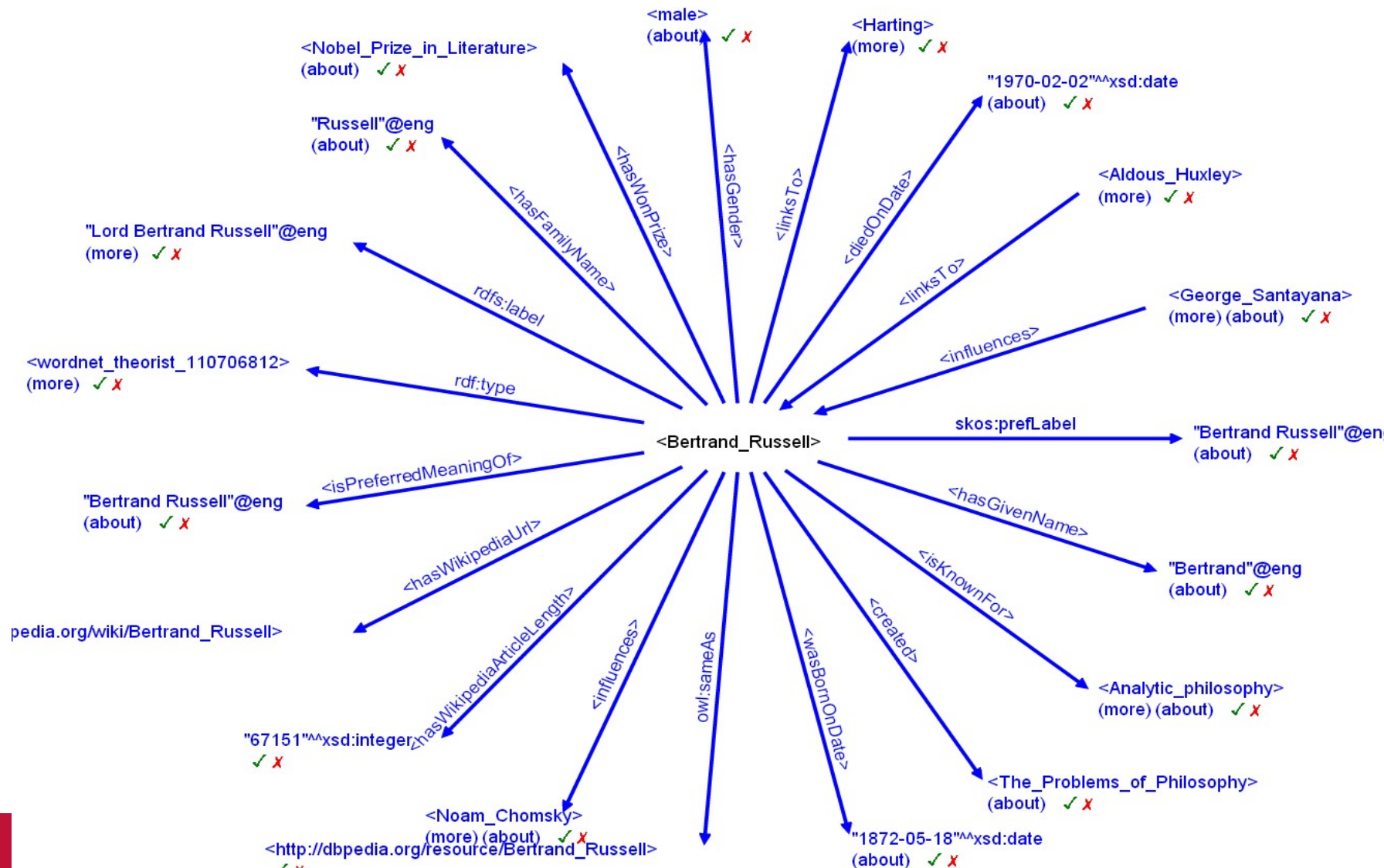
WordNet

GeoNames





# YAGO



# Vocabulaires généraux

## ■ Rdf

- Rdf:type

## ■ Rdfs

- *rdfs:subClassOf, rdfs:property, rdfs:domain, rdfs:range*

## ■ Dublin Core

- xmlns:dc=<http://purl.org/dc/elements/1.1/>
- **dc:title ... description de documents**

## ■ (Dolce)

## ■ Geo84

- Geo:lat, geo:lon

## ■ Foaf

- **foaf:Person -> foaf:name**

## ■ ...

## Autres vocabulaires

- Schema.org (for Web content)

<http://schema.org>

- Creative Commons (types of licences)

<http://creativecommons.org/ns#>

- Facebook Open Graph (for Web content)

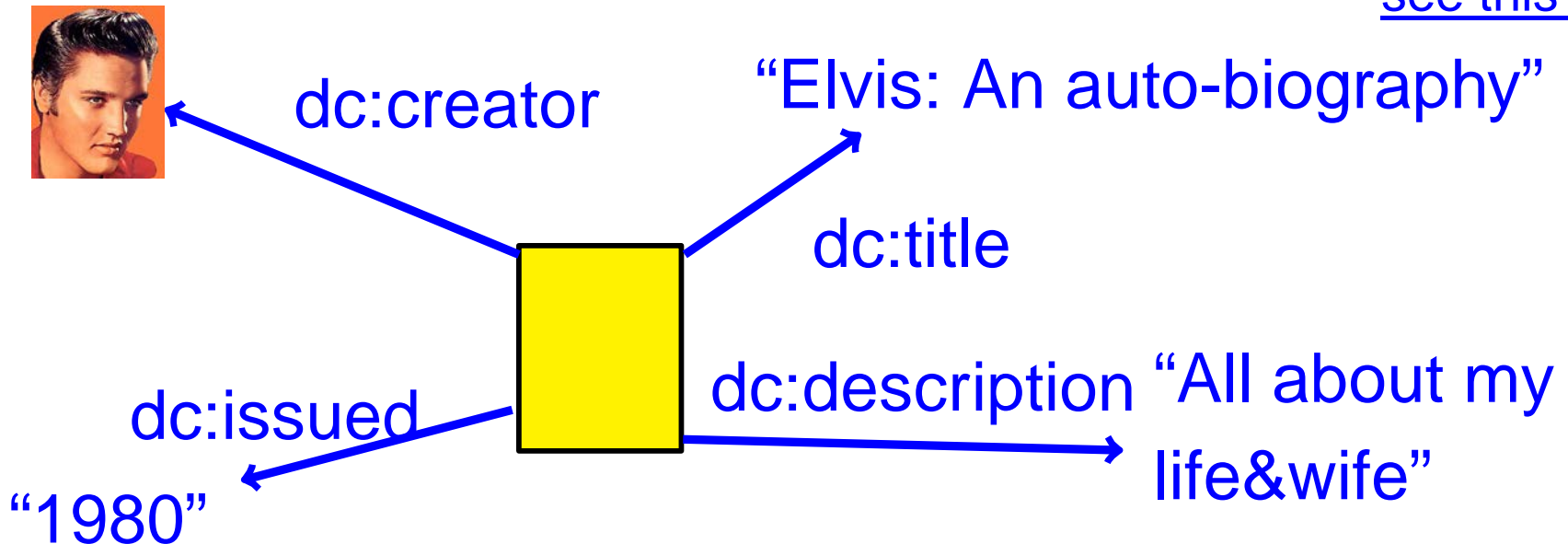
<http://ogp.me/>

## Exemple: Dublin Core

Dublin Core pour décrire des documents.

dc:creator, dc:title, dc:format, dc:MediaType,  
dc:language...

[see this KB](#)





## Exemple FOAF

### ■ Vocabulaire pour décrire une personne et ses relations avec d'autres; en format RDF/XML

#### ■ <rdf:RDF ...>

- <foaf:Person> <foaf:name>Jimmy Wales</foaf:name> <foaf:title>Mr.</foaf:title>  
<foaf:givenName>Jimmy</foaf:givenName>
  - <foaf:familyName>Wales</foaf:familyName>
  - <foaf:mbox rdf:resource="mailto:jwales@bomis.com"/>
  - <foaf:homepage rdf:resource="http://www.jimmywales.com/"> <foaf:nick>Jimbo</foaf:nick>  
<foaf:depiction rdf:resource="http://www.jimmywales.com/aus\_img\_small.jpg"/>
  - <foaf:interest> <rdf:Description rdf:about="http://www.wikimedia.org" rdfs:label="Wikipedia"/>  
</foaf:interest>
  - <foaf:publications rdf:resource="http://www.jimmywales.com/pubs/publications.rdf"/> ...
  - <foaf:knows>
    - <foaf:Person> <foaf:name>Angela Beesley</foaf:name></foaf:Person>
  - </foaf:knows>
  - <foaf:knows>
    - <foaf:Person rdf:about="http://jimmycricket.com/me"> <foaf:name>Jimmy Cricket</foaf:name> </foaf:Person>  
</foaf:knows>
  - </foaf:Person>
- </rdf:RDF>



# Trouver un vocabulaire

- **Lov**
- <http://lov.okfn.org/dataset/lov/>
- **Demo**



# Des outils pour trouver des vocabulaires et ensembles de données

- <http://datahub.io/>
- <http://lov.okfn.org/dataset/lov/>
- <http://prefix.cc/>
- <http://data.gouv.fr>

# Des jeux de données de référence

Généralement associés à un vocabulaire, éventuellement défini par une ontologie

- Dbpedia
- Geonames
- Bnf
- Europeana
- DBLP
- BBC
- British Museum
- Library of Congress
- Fondation Getty
- ...



# Open Data

# Rapports avec l'OpenData

## ■ Open Data

- Mouvement international qui tend à rendre disponibles publiquement les données produites sur fonds publics
- S'étend à une tendance à rendre des données utilisables publiquement sur le Web

## ■ Open Data n'implique pas Web Sémantique et Données Liées (Linked Data), mais le permet

- Ex: publications de données au format CSV



## Linked Open Data Project

- US census data
- BBC music database
- Gene ontologies
- DBpedia general knowledge, + YAGO, + Cyc etc.
- UK government data
- geographical data in abundance
- national library catalogs (USA, Germany etc.)
- publications (DBLP)
- ...and many more



## **[Data.gouv.fr](https://data.gouv.fr)**

- **Presque toutes les réutilisations mises en avant sont des cartes**
- **Exemples de données**
- **Exemples de cartes: cf OpenGeoData.fr**



# Ensemble des gares voyageurs du territoire métropolitain.

Ajouter ▾

Fond de carte

Imprimer

Mesurer

Géosig

res voyageurs  
ropolitain.

iment les gares  
d'une commune

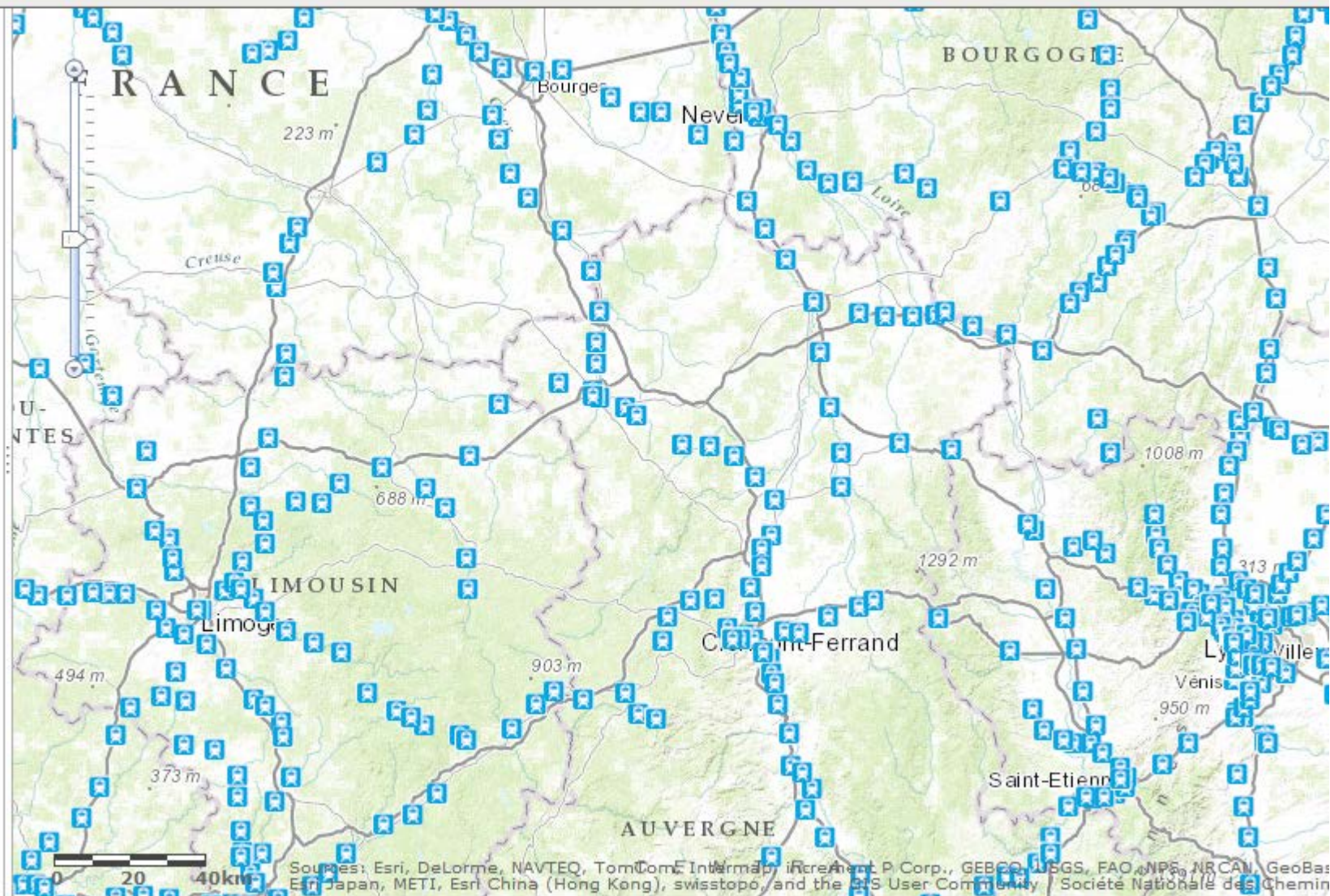
ta  
odification : 24

aluations, 0  
(8 vues)

lémentaires...

e dans :  
nline

te



Sources: Esri, DeLorme, NAVTEQ, TomTom, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, Esri Japan, METI, Esri China (Hong Kong), swisstopo, and the GIS User Community / Société Nationale des Chemins

# France 2002 : premier tour des élections présidentielles

Si votre souris survole une commune, son nom et ses résultats sont affichés. Zooms et déplacements possibles! (avantages du SVG).

[Retour à la documentation](#)

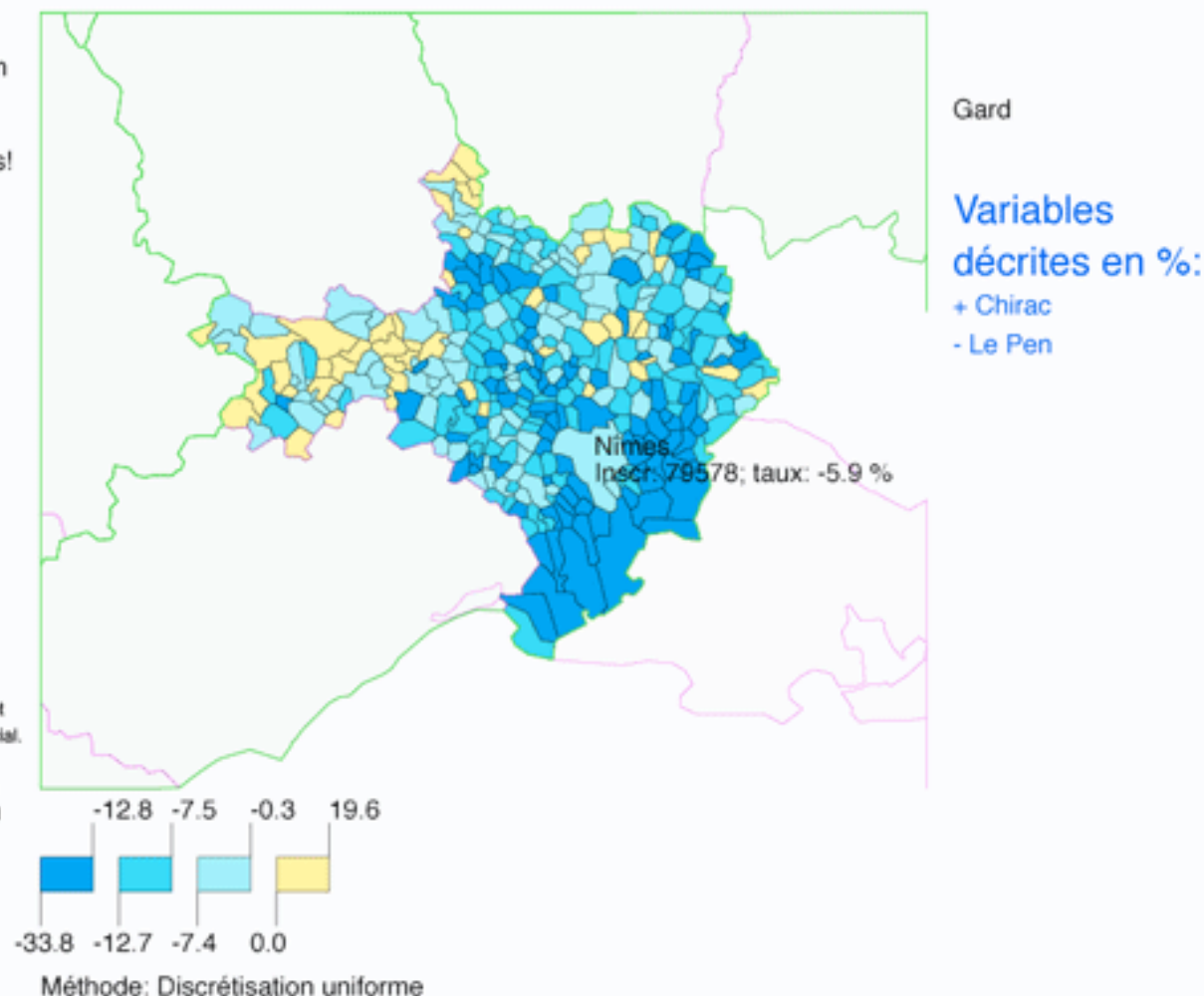
Éric Guichard

ENSSIB et ENS

Usage libre des cartes, données et fond sauf pour un usage commercial. Mention de l'auteur obligatoire.

Any human or machine accessing this document is supposed to read French and to have accepted the legal contract written above in this language.

[Lire la licence](#)



Auteur: Éric Guichard. Logiciel Ératostène, écrit en Perl (2000 pour la version ps, 2004 pour la version svg).

Sources: Ministère de l'Intérieur, CD-Atlas de France (revu et largement corrigé), Éric Guichard.

Remerciements: J. Beigbeder, A. Danzart, J.-C. Moissinac, C. Potier, H. Théry.



# Les contours des communes

Découpage administratif

https://www.data.gouv.fr/fr/dataset/decoupage-administratif-communal-francais-issu-d-openstreetmap

Applications Free - Envoyez vos d... STEAMER - Spatio-T... XCWeather - Foreca... Etalab, mission gou... Open Knowledge Pa... Save to Mendeley www.stat.berkeley.e... Autres favoris

data.gouv.fr Plateforme ouverte des données publiques françaises

Comment ça marche ? Producteurs Licence Ouverte Métriques Etalab Jean-Claude Moissinac

Rechercher Où Thématiques PUBLIEZ UN JEU DE DONNÉES !

## Découpage administratif communal français issu d'OpenStreetMap

Ce jeu de données a été publié le 17 novembre 2013 à l'initiative et sous la responsabilité de **OpenStreetMap France** **NON CERTIFIÉ**

Exports du découpage administratif français au niveau communal (contours des communes) issu d'OpenStreetMap produit dans sa grande majorité à partir du cadastre.

Ces données sont issues du crowdsourcing effectué par les contributeurs au projet OpenStreetMap et sont sous licence ODbL qui impose un partage à l'identique et la mention obligatoire d'attribution doit être "© les contributeurs d'OpenStreetMap sous licence ODbL" conformément à <http://osm.org/copyright>


Un export automatique quotidien au format shapefile est disponible, ainsi qu'un second export avec des géométries allégées et vérifiées topologiquement (pas de chevauchement).

### Descriptif du contenu des fichiers "communes"

#### Origine

Les données proviennent de la base de données cartographiques OpenStreetMap. Celles-ci ont été constituées à partir du cadastre mis à disposition par la DGFIP sur [cadastre.gouv.fr](http://cadastre.gouv.fr). En complément sur Mayotte où le cadastre n'est pas disponible sur

**Producteur**



Le wiki cartographique mondial qui crée et fournit des données géographiques sous licence libre ODbL. OSM est représenté en France par OpenStreetMap France, association régie...

S'ABONNER

Informations

# Les résultats

The screenshot shows a web browser window with the URL <https://www.data.gouv.fr/fr/dataset/election-presidentielle-2012-resultats-572124>. The page header includes the data.gouv.fr logo and the tagline 'Plateforme ouverte des données publiques françaises'. The navigation bar contains links like 'Comment ça marche?', 'Producteurs', 'Licence Ouverte', 'Métriques', and 'Etalab'. The main content area is titled 'Election présidentielle 2012 - Résultats' and includes a description: 'Ce jeu de données provient d'un service public certifié' and 'Publié le 14 septembre 2013 par Etalab Bot'. It also mentions 'Résultats de l'élection présidentielle 2012, tours 1 et 2, par régions, départements, circonscriptions législatives, cantons'. A 'Ressources' section lists an 'XLS' file named 'Ressource sans nom'. On the right, a 'Producteur' section features the logo of the 'MINISTÈRE DE L'INTÉRIEUR' and a description of its role. A 'S'ABONNER' button is also visible.

Election présidentielle 2012 - Résultats

Ce jeu de données provient d'un service public certifié  
Publié le 14 septembre 2013 par Etalab Bot

Résultats de l'élection présidentielle 2012, tours 1 et 2, par régions, départements, circonscriptions législatives, cantons

Ressources

XLS Ressource sans nom

☆ UTILE (1) ⚠

Producteur

Liberté • Égalité • Fraternité  
RÉPUBLIQUE FRANÇAISE

MINISTÈRE DE L'INTÉRIEUR

Placé au cœur de l'Etat, le ministère de l'intérieur assure, en premier lieu, la permanence et la continuité de l'Etat. Cette fonction régalienne se concrétise par le rôle...

S'ABONNER

# Conclusion

## Quelques points importants

- **Grandes variétés de protocoles, langages, technologies utilisées sur le Web**
- **Crawler, c'est parcourir un graphe**
- **Construire un crawler est une tâche non triviale d'ingénierie**
  - en particulier si on veut crawler à grande échelle

7 October 2013



Institut Mines-Télécom

Licence de droits d'usage

# References

## Software

Wget, a simple yet effective Web spider (free software)

Heritrix, a Web-scale highly configurable Web crawler, used by the Internet Archive (free software)

HTML Parser, TagSoup: Java libraries for parsing real-world Web pages

## To go further

A good textbook [Chakrabarti, 2003] Main references:

HTML 4.01 recommendation [W3C, 1999] HTTP/1.1 RFC [IETF, 1999b]

-

-



# Bibliography I

Serge Abiteboul, Grégory Cobena, Julien Masanès, and Gerald Sedrati. A first experience in archiving the French Web. In *Proc. ECDL*, Roma, Italie, September 2002.

Serge Abiteboul, Mihai Preda, and Gregory Cobena. Adaptive on-line page importance computation. In *Proc. WWW*, May 2003.

BBC. Fifteen years of the web.

<http://news.bbc.co.uk/2/hi/technology/5243862.stm>, 2006.

Accessed March 2009.

Andrei Z. Broder, Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. Syntactic clustering of the Web. *Computer Networks*, 29(8-13):1157–1166, 1997.

Soumen Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, San Fransisco, USA, 2003.

Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused crawling: A new approach to topic-specific Web resource discovery. *Computer Networks*, 31(11–16):1623–1640, 1999.

## Bibliography II

Michelangelo Diligenti, Frans Coetzee, Steve Lawrence, C. Lee Giles, and Marco Gori. Focused crawling using context graphs. In *Proc. VLDB*, Cairo, Egypt, September 2000.

Electronic Software Publishing Corporation. Internet & World Wide Web history. [http://www.elsop.com/wrc/h\\_web.htm](http://www.elsop.com/wrc/h_web.htm), 2008.  
Accessed March 2009.

IETF. Request For Comments 791. Internet Protocol.  
<http://www.ietf.org/rfc/rfc0791.txt>, September 1981a.

IETF. Request For Comments 793. Transmission Control Protocol.  
<http://www.ietf.org/rfc/rfc0793.txt>, September 1981b.

IETF. Request For Comments 1738. Uniform Resource Locators (URLs). <http://www.ietf.org/rfc/rfc1738.txt>, December 1994.

IETF. Request For Comments 1034. Domain names—concepts and facilities. <http://www.ietf.org/rfc/rfc1034.txt>, June 1999a.



## Bibliography III

IETF. Request For Comments 2616. Hypertext transfer protocol—HTTP/1.1. <http://www.ietf.org/rfc/rfc2616.txt>, June 1999b.

IETF. Request For Comments 2965. HTTP state management mechanism. <http://www.ietf.org/rfc/rfc2965.txt>, October 2000.

Martijn Koster. A standard for robot exclusion. <http://www.robotstxt.org/orig.html>, June 1994.

Pierre Senellart. Identifying Websites with flow simulation. In *Proc. ICWE*, pages 124–129, Sydney, Australia, July 2005.

sitemaps.org. Sitemaps XML format. <http://www.sitemaps.org/protocol.php>, February 2008.

W3C. HTML 4.01 specification, September 1999. <http://www.w3.org/TR/REC-html40/>.

# Licence de droits d'usage



Contexte public } avec modifications

***Par le téléchargement ou la consultation de ce document, l'utilisateur accepte la licence d'utilisation qui y est attachée, telle que détaillée dans les dispositions suivantes, et s'engage à la respecter intégralement.***

La licence confère à l'utilisateur un droit d'usage sur le document consulté ou téléchargé, totalement ou en partie, dans les conditions définies ci-après et à l'exclusion expresse de toute utilisation commerciale.

Le droit d'usage défini par la licence autorise un usage à destination de tout public qui comprend :

- le droit de reproduire tout ou partie du document sur support informatique ou papier,
- le droit de diffuser tout ou partie du document au public sur support papier ou informatique, y compris par la mise à la disposition du public sur un réseau numérique,
- le droit de modifier la forme ou la présentation du document,
- le droit d'intégrer tout ou partie du document dans un document composite et de le diffuser dans ce nouveau document, à condition que :
  - L'auteur soit informé.

Les mentions relatives à la source du document et/ou à son auteur doivent être conservées dans leur intégralité. Le droit d'usage défini par la licence est personnel et non exclusif.

Tout autre usage que ceux prévus par la licence est soumis à autorisation préalable et expresse de l'auteur : [sitopedago@telecom-paristech.fr](mailto:sitopedago@telecom-paristech.fr)

7 October 2013

