

# MAS-WES | Mini projets à choix

Vous devez proposer une version du cahier des charges lors de la première semaine. Après validation du cahier des charges par Julien Tscherrig, Omar Abou Khaled et Maria Sokhn vous pouvez commencer à réaliser le travail.

Veuillez indiquer votre nom dans la case correspondante au travail que vous souhaitez effectuer. Le détail pour chaque projet est détailler plus bas. Vous pouvez également proposer un travail traitant du Web 3.0 :

<b>A.1 - 1 pers.</b>	Ambord, Sehgelmeble, Hauenstein		
<b>A.2 - 1 pers.</b>	Blanc Grégoire		
<b>A.3 - 1 pers.</b>	Kouch		
<b>B.1 - 2 pers.</b>	Chassot , Yazid Y.	Stauffer	
<b>B.2 - 3 pers.</b>	Gomez	Simon Byrne	
<b>B.3 - 3 pers.</b>	Peter	Chèvre Sébastien	Muller
<b>C.1 - 2 pers.</b>			
<b>C.2 - 2 pers.</b>			

## A. Extraction et Conversion

L'objectif est de récupérer de manière automatique les données se trouvant sous forme de documents ou de pages / sites Internet, ensuite modéliser une ontologie, puis convertir les données extraites au format RDF(S).

### Critères pris en compte :

1. La qualité de l'ontologie créée
2. La qualité / complexité des requêtes SPARQL réalisées
3. Rapport

### Contraintes :

- Format du programme attendu : application Web Java
- Travail avec la librairie OpenRDF vue au cours (<http://www.rdf4j.org>)

## A.1 Extraction de données provenant de blogs (TDG)

### Détail

Extraction et conversion d'informations provenant de la tribune de Genève

<http://blog.tdg.ch>

Nombre de personnes

1

## A.2 Extraction de données provenant de blogs (NZZ)

### Détail

Extraction et conversion d'informations provenant de la NZZ Online

<http://blog.nzz.ch>

Nombre de personnes

1

## A.3 Extraction de données provenant de blogs (Tages Anzeiger)

### Détail

Extraction et conversion d'informations provenant du Tages Anzeiger

<http://www.tagesanzeiger.ch/blogs/>

Nombre de personnes

1

## B. Requêtes et Visualisations

L'objectif est d'exploiter des informations mises à disposition au format RDF afin de pouvoir visualiser au mieux des statistiques / relations intéressantes via des tableaux, graphiques, etc. Le travail devra au minimum contenir 2 visualisations distinctes utilisant ou non des graphiques (tableaux ou autres représentations).

Critères seront pris en compte :

4. La créativité et pertinence des informations présentées
5. La qualité / complexité des requêtes SPARQL réalisées
6. La qualité de la présentation des informations
7. Rapport

Contraintes:

- Format du programme attendu : application web Java
- Librairie graphique : D3JS (<http://www.d3js.org>), uniquement si graphique nécessaire
- Données mise à disposition au format RDF pouvant-être chargées dans une base de données sémantique (<http://www.rdf4j.org>)

## B.1 20 Minutes

### Détail

Une extraction d'environ 50'000 articles provenant du 20 minutes (FR+DE) sera mise à disposition. L'extraction comporte le nom de l'article, le sous-titre, la date de parution, le contenu ainsi que la liste des parlementaires cités dans l'article.

### Remarque

*vous pouvez travailler avec un nombre restreint d'articles*

### Nombre de personnes

2

## B.2 Twitter et/ou Facebook

### Détail

Une extraction d'informations provenant de Twitter / Facebook via leur API officielle concernant des profils ciblés (récupération complète) et non-ciblés (récupération de base) au format RDF sera mise à votre disposition. Environ 100 profils complets (relations + tweets / statuts) et 10'000 profils de bases

### Remarque

*vous pouvez travailler avec un nombre restreint de profils*

### Nombre de personnes

3

## B.3 DBPedia

### Détail

Toutes les données provenant de DBPedia via son portail (<http://dbpedia.org>)

### Nombre de personnes

3

## C. Utilisation de Moteurs d'Inférences

Durant les cours nous avons parlé à plusieurs reprises des inférences possibles grâce aux langages RDF(S) + OWL. Il s'agit de démontrer l'avantage d'utiliser un moteur d'inférence.

Il s'agira de créer une petite ontologie et d'y ajouter plusieurs règles d'inférences pertinentes pour juger de leur fonctionnement (résultats avec et sans inférences). Une application web devra permettre de tester et visualiser le fonctionnement (avec / sans inférences).

Une courte analyse statistique des temps d'exécutions avec ou sans inférences est également attendue (ajout / suppression / lecture).

### Critères seront pris en compte :

1. La créativité et modélisation de l'ontologie
2. L'intégration correcte des moteurs d'inférence

3. La qualité de la présentation des informations
4. Rapport

Contraintes:

- Format du programme attendu : application web Java
- Librairie graphique : D3JS (<http://www.d3js.org>), uniquement si graphique nécessaire

## C.1 Jena

Détail

Nous avons travaillé durant les cours avec la Librairie OpenRDF. Jena est une autre librairie libre - fournie par Apache (<http://jena.apache.org>) - permettant de réaliser les mêmes opérations.

Nombre de personnes

2

## C.2 OpenRDF

Détail

Utilisation de la librairie OpenRDF (vue au cours) pour démontrer et comprendre les inférences (<http://rdf4j.org>).

Nombre de personnes

2