

Manuel d'utilisation AlignViz

Haïfa Zargayouna

LRI, Université Paris-Sud, Bât. G, INRIA Saclay - Île-de-France

2-4 rue Jacques Monod, F-91893 Orsay, France

haifa.zargayouna@lri.fr

Table des matières

1	Introduction	1
2	Installation	2
3	Visualisation	2
3.1	Visualiser une ontologie	2
3.2	Visualiser deux ontologies simultanément	4
3.3	Visualiser un alignement	4
4	Validation des correspondances	4
4.1	Affichage et tri	4
4.2	Notion de session	5
4.3	Différents états de validation	5
4.4	Extrait d'une sortie d'alignement évalué (.viz)	6
5	Remerciements	7

1 Introduction

AlignViz est un outil de visualisation d'ontologies et de validation des correspondances retrouvé par un outil d'alignement. Les ontologies doivent être formalisées en OWL. Le fichier d'alignement qui contient les correspondances doit être formalisé en RDF/XML ¹.

Cet outil a pour but de faciliter le travail de l'évaluation d'un alignement par un expert. Il essaie d'apporter les réponses aux besoins suivants :

¹Selon le format décrit dans <http://alignapi.gforge.inria.fr/format.html>

- Un affichage ergonomique d’une ontologie avec diverses fonctionnalités de navigation telles qu’accéder à un type particulier de correspondance ou à un concept particulier.
- Divers niveaux de visualisation, l’évaluateur doit pouvoir avoir une vision d’ensemble de ce qu’il a à évaluer avec la possibilité de zoomer sur les parties qui l’intéressent. Cette fonctionnalité est d’autant plus importante quand les ontologies sont volumineuses.
- Permettre d’évaluer les correspondances en les validant ou en les supprimant ainsi que les sauvegarder. Cette fonctionnalité nécessite de gérer les sessions des utilisateurs. Il est aussi important que les actions de validation soient réversibles, ainsi l’expert peut à tout moment reconsidérer sa validation.

2 Installation

AlignViz requiert l’installation de Graphviz (*Graph Visualization Software*). Graphviz contient un ensemble d’outils qui manipulent des graphes définis à l’aide de scripts suivant le langage DOT. L’outil est accessible à <http://www.graphviz.org/Download.php> dans sa version 2.20 et plus.

Il faut disposer de Java à partir de 1.6.

Il faut copier le fichier .jar dans un répertoire. Pour lancer ALignViz, il faut double-cliquer sur le .jar, ou taper en ligne de commande :

```
java -jar le_fichier.jar
```

3 Visualisation

AlignViz exploite l’outil Graphviz ainsi que les API Protégé-OWL et Jena ontology. Il permet de visualiser une ontologie, ainsi que deux ontologies simultanément afin de pouvoir visualiser les correspondances retrouvées entre concepts par un outil d’alignement. L’onglet *Visualisation* permet de visualiser une ontologie, l’onglet *Alignement* sert à visualiser deux ontologies ainsi que leur alignement.

3.1 Visualiser une ontologie

Il faut cliquer dans le menu *Fichier* sur *Charger un fichier OWL* (voir figure 1).

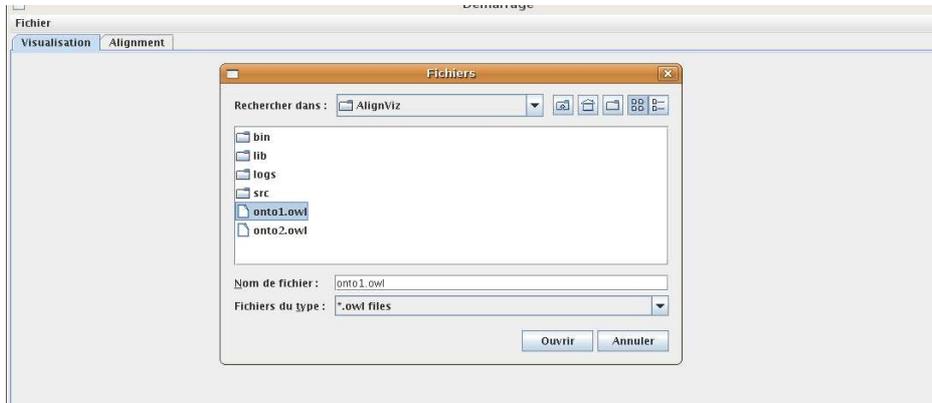


FIG. 1 – Charger une ontologie

Différents boutons permettent de visualiser un concept avec ses ancêtres et fils, de zoomer, etc. Pour visualiser toute l'ontologie, il faut sélectionner le concept Top (*Thing*) et choisir le niveau de granularité voulu. La figure 2, présente les différentes fenêtres.

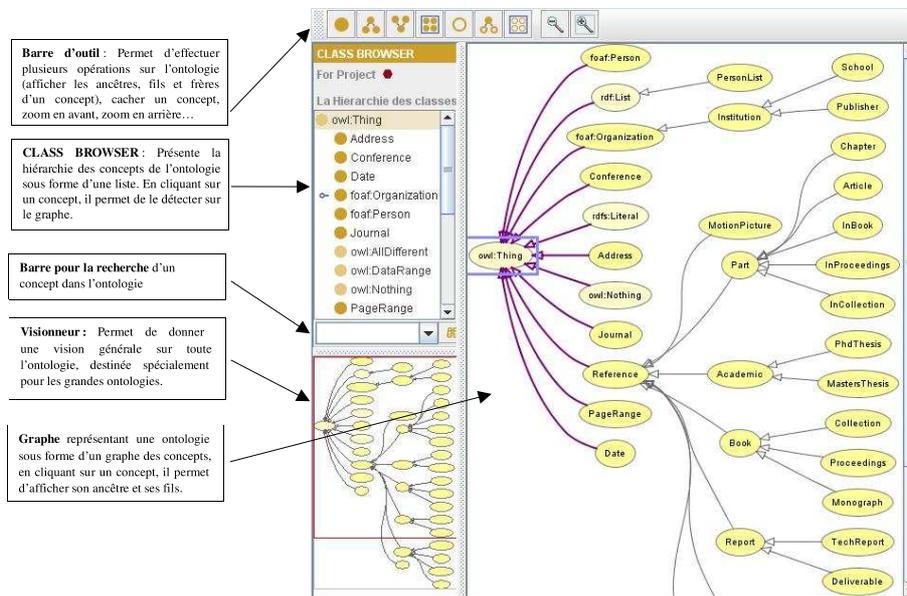


FIG. 2 – Visualisation d'une ontologie

3.2 Visualiser deux ontologies simultanément

L'onglet *Alignement* permet de charger deux ontologies. Une fois les ontologies chargées, le bouton *Afficher* permet d'afficher simultanément les deux ontologies, avec les mêmes fonctionnalités que celle décrites dans la section précédente (voir figure 3).

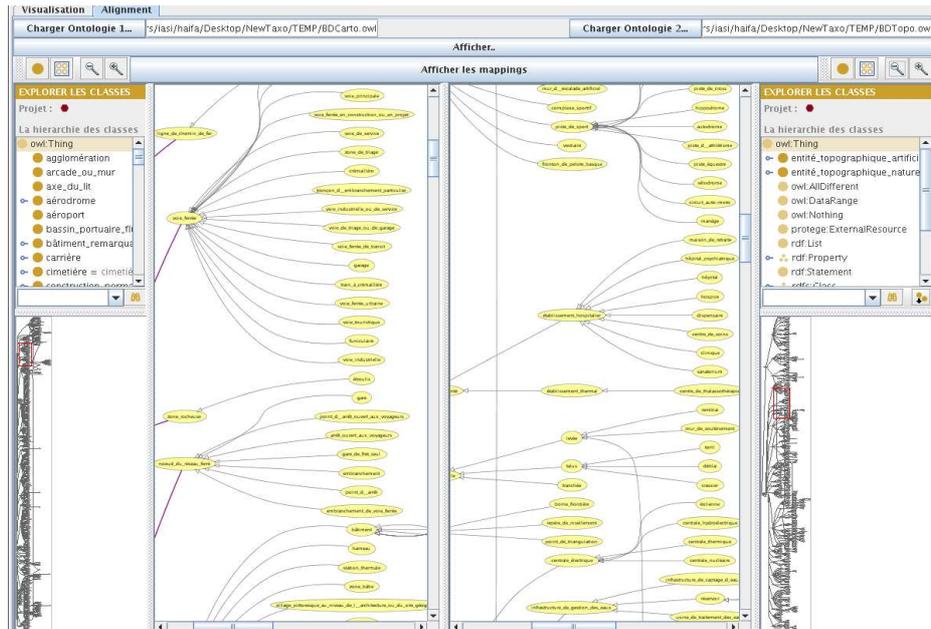


FIG. 3 – Visualisation de deux ontologies

3.3 Visualiser un alignement

Le bouton *Afficher les mappings* permet de charger le fichier d'alignement des deux ontologies. Les correspondances sont affichées dans un tableau qui comprend l'URI des concepts, les types des relations de correspondance ainsi que leur état de validation. Ces aspects sont détaillés dans la section qui suit.

4 Validation des correspondances

4.1 Affichage et tri

Les correspondances peuvent être triées, en cliquant sur l'onglet du champs qu'on veut trier. Ainsi nous pouvons visualiser les correspondances par type de relation. En cliquant deux fois sur une ligne du tableau, les concepts

concernés par la correspondance sont marqués par un cadre dans leurs ontologies respectives (voir figure 4).

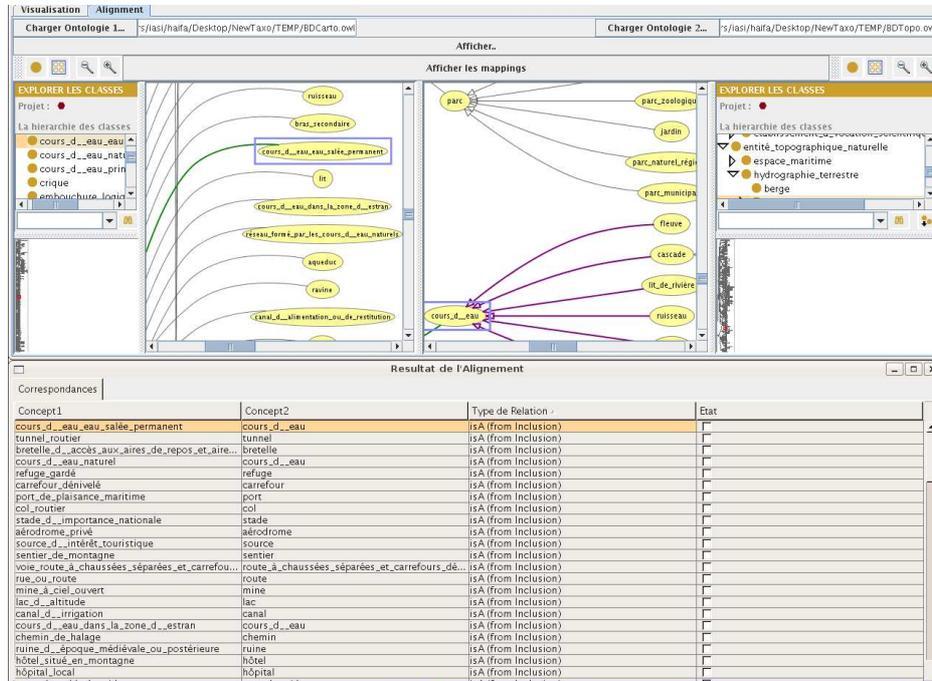


FIG. 4 – Visualisation de correspondances et ancrage dans les deux ontologies

4.2 Notion de session

Lors d'un premier téléchargement, les correspondances n'ont pas encore été validées, la case *Etat* n'est donc pas cochée. Elle le sera au fur et à mesure de la validation (voir section suivante). Les différentes validations peuvent être enregistrées à tout moment (en cliquant sur le bouton *Enregistrer*), un fichier *.viz* est donc créé et il est possible alors de reprendre la validation à l'état où on l'a laissée en téléchargeant le *.viz* à la place du *.rdf*. L'expert peut modifier son évaluation et re-enregistrer. Un autre expert peut valider le même alignement en repartant du *.rdf*.

4.3 Différents états de validation

Un clic droit sur une ligne du tableau permet de valider la correspondance, les différents états sont :

- Valide : Quand la correspondance est correcte.

- Invalide : Quand la correspondance est fausse.
- Neutre : Quand l'expert n'arrive pas à se faire un avis. Il est toujours possible de revenir sur ces correspondances et modifier leur état.

Une fois évalué la case *Etat* de la correspondance est cochée automatiquement. Les correspondances qui n'ont pas été évaluées sont marquées comme non analysées dans la sortie .viz (voir annexe).

Il est également possible d'ajouter un commentaire pour motiver l'évaluation de la correspondance et qui peut servir de base pour une discussion entre experts (voir figure 5).

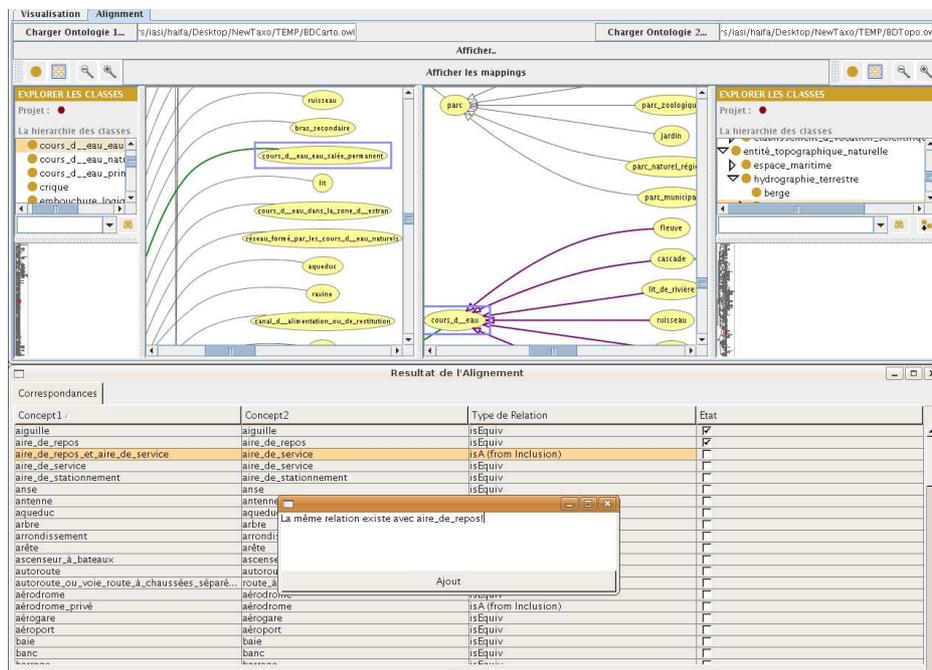


FIG. 5 – Ajout d'un commentaire

Annexe

4.4 Extrait d'une sortie d'alignement évalué (.viz)

```

...
<map>
  <Cell>
    <entity1 rdf:resource="file:/E:/BDCarto/BDCarto.owl#colline"/>
    <entity2 rdf:resource="file:/E:/BDTopo/BDTopo.owl#colline"/>

```

```

    <measure rdf:datatype="float">1.0</measure>
    <relation>isEquiv</relation>
    <analyse>Valide</analyse>
    <comment/>
  </Cell>
</map>
<map>
  <Cell>
    <entity1 rdf:resource="file:/E:/BDCarto/BDCarto.owl#aire_de_repos_et_aire_de_servi
    <entity2 rdf:resource="file:/E:/BDTopo/BDTopo.owl#aire_de_service"/>
    <measure rdf:datatype="float">0.7808157090468507</measure>
    <relation>isA (from Inclusion)</relation>
    <analyse>Valide</analyse>
    <comment>La même relation existe avec aire_de_repos!</comment>
  </Cell>
</map>
<map>
  <Cell>
    <entity1 rdf:resource="file:/E:/BDCarto/BDCarto.owl#hameau"/>
    <entity2 rdf:file:/E:/BDTopo/BDTopo.owl#hameau"/>
    <measure rdf:datatype="float">1.0</measure>
    <relation>isEquiv</relation>
    <analyse>notAnalysed</analyse>
    <comment/>
  </Cell>
</map>
...

```

5 Remerciements

Je remercie Jamal Ait Alla qui a participé à l'implémentation de l'aspect visualisation de l'outil.