

20/05/2019

Création de l'ontologie pour le jeu vidéo *Arena of Valor*

Projet Web sémantique

Table de matières

Introduction	2
État de l'art	3
Exemples d'utilisation	3
Questions de compétences	3
Construction de l'ontologie	4
Méthode	4
Création de classes	5
Création d'instances	6
Création de propriétés et d'attributs	7
Data Property	7
Objet Property	8
Vérification de consistance	8
Visualisation de l'ontologie	9
Requêtes SPARQL	10
Point d'accès SPARQL à l'aide Virtuoso	11
Installation	11
Configuration	11
Création du point d'accès	13
Requêtes testées et résultats	14
Conclusion	23
Bibliographie et sitographie	24

1. Introduction

Le Web sémantique, ou le réseau sémantique, est une extension de l'Internet standardisée par le W3C, dans laquelle est encouragée l'utilisation de formats de données normés s'appuyant sur le modèle RDF (Resource Description Framework). L'ontologie, l'une des technologies de base du Web sémantique, consiste à modéliser les données de certains domaines de manière normative. La réalisation d'une ontologie constitue en effet la transformation des connaissances propres à certains groupes en édition électronique facilitant le partage ainsi que la réutilisation de ces connaissances. L'ontologie permet en général une représentation précise d'un certain domaine en formalisant ses propres concepts et les relations qu'ils entretiennent.

Dans notre projet de Web sémantique, nous nous intéressons au domaine d'Esport, lequel désigne l'activité de confrontation intellectuelle basée sur des équipements de haute technologie et sur des plateformes électroniques. Ce dernier, issu des jeux vidéo, s'est développé rapidement depuis son apparition au tout début du XXI^e siècle et est devenu aujourd'hui un phénomène culturel au niveau mondial. L'Esport s'oriente vers une voie de développement professionnel et mérite donc des recherches.

L'Esport peut se dérouler à travers divers genres de jeux, dont le type MOBA (Multiplayer Online Battle Arena) est l'un des plus populaires. Le développement des appareils mobiles en plein essor, de plus en plus de joueurs d'Esport s'affrontent dans les jeux mobiles, parmi lesquels *Arena of Valor* s'avère être le plus prisé. Notre projet vise donc à construire une ontologie formalisant les concepts fondamentaux de ce jeu tels que les personnages, les équipements et les routes de l'arène.

Cette ontologie nous permettrait non seulement de consulter les informations des éléments précités, mais aussi de nous donner des conseils concernant le choix des héros, des équipements ainsi que des routes que les héros doivent emprunter. Elle serait également en mesure de déterminer si certaines connaissances qu'a le joueur sur ce jeu sont correctes ou non. Toutes ses compétences seront testées par l'exécution des requêtes SPARQL, ce qui constitue justement la finalité de notre projet.

2. État de l'art

Bien que le jeu *Arena of Valor* soit très populaire, il n'existe jusqu'à présent ni d'ontologie ni de page wiki en français consacrées à ce jeu. Nous pouvons trouver seulement sur son site officiel un panneau qui présente et classifie les héros selon leur type. En ce qui concerne les équipements, nous ne pouvons pas avoir leurs informations que par le lancement du jeu, ce qui est manifestement loin d'être pratique. Aussi voudrions-nous créer notre ontologie afin de promouvoir son développement, de le faire connaître davantage et d'aider les joueurs à mieux comprendre les fonctionnements du combat.

3. Exemples d'utilisation

Les jeux MOBA nécessitent généralement dix joueurs. Ces derniers forment respectivement deux équipes et ensuite se battent. Dans ces jeux, il vaut mieux que chaque équipe contienne cinq positions, c'est-à-dire cinq héros de types différents, pour que l'équipe soit équilibrée. Les héros se développent en niveaux et sont équipés par divers genres d'équipements selon leurs caractéristiques.

Néanmoins, certains joueurs, surtout ceux auxquels ce genre de jeux sont nouveaux, voire étrangers, ne savent ni quel héros choisir, ni comment l'équiper, ni quelle route emprunter. D'où la nécessité de notre ontologie visant à bien classifier tous ces éléments en vue d'en faire des recommandations raisonnées.

3.1. Questions de compétences

Nous envisageons une dizaine de questions de compétence qui pourraient souvent se poser :

- 1) Qu'est-ce que l'on doit prendre comme équipements si on choisit Murad comme héros ?
- 2) De quel type est le héros Tulen ?
- 3) On voudrait jouer un guerrier de type finisseur. Que propose-t-elle ?

- 4) On voudrait choisir un héros de type mage qui ne soit pas si fragile (ne pas avoir de « Poise¹ » faible), est-ce qu'il existe ?
- 5) On veut choisir une héroïne de type mage. Que propose-t-elle ?
- 6) On veut choisir un héros qui soit facile à manipuler. Que propose-t-elle ?
- 7) Est-ce que le héros Raz emprunte la voie de dragon ?
- 8) Alice prend les bottes de paladin comme équipement. Est-ce que c'est correct ?
- 9) Le HP d'Arthur est-il supérieur à celui d'Annette ?
- 10) Est-ce que tous les mages sont féminins ?
- 11) Existe-il un héros de type tank qui a une très forte « ability effects » ?
- 12) Je voudrais savoir tous les héros de type Assassin dont le base HP dépasse 3300 et leurs niveau de difficulté.
- 13) On voudrait choisir un héros de type tireur qui est facile à manipuler en regardant une liste montrant leur valeur de « BaseAttack » par ordre croissant.

4. Construction de l'ontologie

Une fois l'objectif en exergue, nous nous prenons à construire l'ontologie de manière à répondre aux questions susmentionnées en utilisant *Protégé*, système auteur destiné à la création d'ontologies.

4.1. Méthode

Notre méthode employée paraît assez systématique eu égard au domaine à caractère très précis. Autrement dit, à la différence des sujets classiques tels que le vin ou le fromage, lesquels exigent parfois des informations de grande envergure, notre projet ne nécessite que des données très spécifiques limitées propres à ce seul jeu vidéo.

Comme précité, les trois éléments fondamentaux de notre ontologie sont les héros, les équipements et les routes. Nous nous référons tout d'abord au panneau des héros à son site officiel, les classifions selon leurs types et en créons des classes. Ensuite, nous ajoutons à chaque héros ses propriétés et attributs. Pour ce qui est des équipements, nous entrons

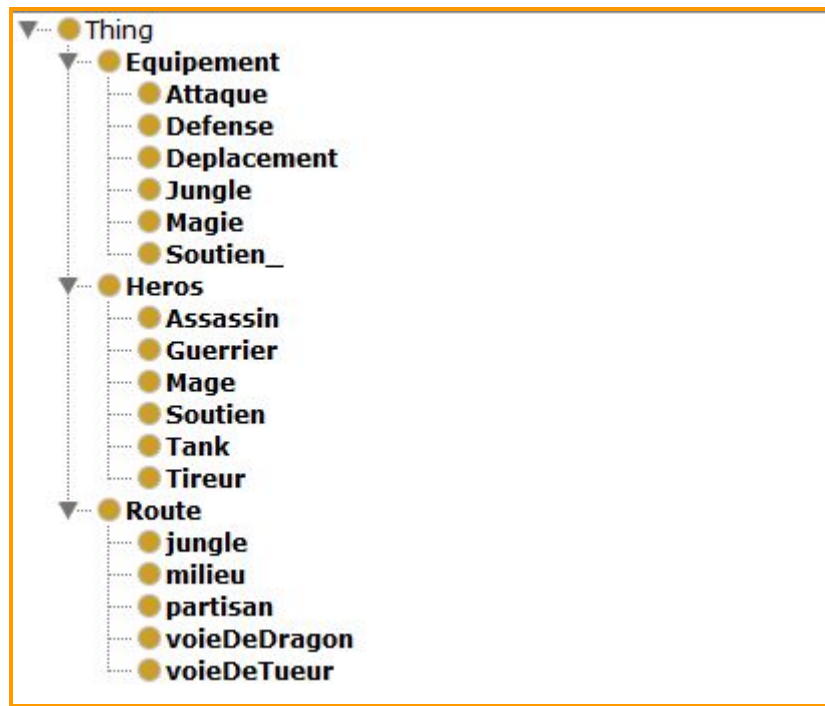
¹ Poise : la force ou les points de santé d'un héros

directement au jeu et consultons leurs informations. Plusieurs guides du jeu sous forme de texte ou de vidéo trouvés sur Internet nous permettent alors de connaître la bonne façon de jouer et ensuite de créer des liens entre ces trois éléments conformément aux conseils donnés par ces guides.

4.2. Création de classes

La création des classes est l'une des étapes les plus importantes dans le développement de notre ontologie. Parmi les approches proposées pour le développement de la hiérarchie des classes de l'ontologie, l'approche « du haut en bas » est employée. C'est-à-dire que l'on présente d'abord les concepts les plus généraux et ensuite on les spécifie. Dans notre cas, nous créons trois classes générales : **Heros**, **Route** et **Equipement**. Ensuite, nous précisons ces trois classes en créant des sous-classes.

Pour la classe **Heros**, six sous-classes y sont attribuées : **Assassin**, **Guerrier**, **Mage**, **Soutien**, **Tank**, **Tireur**, lesquelles désignent les six types de héros. Les sous-classes de la classe **Equipement** correspondent à tout ce qui est nécessaire pour un héros de combattre : **Attaque**, **Defense**, **Deplacement**, **Jungle**, **Magie** et **Soutien**. La dernière classe **Route** est composée de cinq sous-classes désignant les différents chemins qu'un héros doit emprunter en fonction de ses caractéristiques : **jungle**, **milieu**, **partisan**, **voieDeDragon**, **voieDeTureur**.



4.3. Création d'instances

Dans ce jeu, chaque héro et chaque équipement possède un nom spécial et unique. Ils constituent les instances des classes. Voici ci-dessous deux tableaux qui les présente explicitement :

Classe « Heros »	
Assassin	Airi, Batman, Butterfly, Kriknak, Murad, Nakroth, Quillen, Wukong
Tireur	Elsu, Fennik, Lindis, Moren, Slimz, Tel'Annas, The_Joker, Valhein, Violent, Wisp, Yorn
Mage	Aleister, Alice, Azzen'Ka, Diaochan, Gildur, Ignis, Ilumia, Jinnar, Kahlia, Krixi, Lauriel, Liliana, Marja, Mganga, Natalya, Preyta, Raz, Serphera, The_Flash, Tulen, Veera, Zill
Soutien	Alice, Annette, Chaugnar, Grakk, Peura, TeeMee
Tank	Arum, Baldum, Cresht, Lumburr, Mina, Omega, Roxie, Taara, Thane, Toro, Xeniel, Y'bneth

Guerrier	Amily, Arduin, Arthur, Astride, Florentino, Kil’Groth, Lu_Bu, Maloch, Max, Omen, Ormarr, Riktor, Rourke, Ryoma, Skud, Superman, Wonder_Woman, Zanis, Zephys, Zuka
-----------------	---

Classe « Equipement »	
Attaque	Armes_omni, Claves_Sancti, Croc_de_Fenrir, Destructeur_de_formation, Griff_de_Fafnir, La_Bete, Lame_de_foudre, Muramasa, poigne_du_diable
Defence	Amulette_de_longevite, Armure_de_Ra, Bouclier_de_roche, Bouclier_des_damnes, Cape_de_givre, Egide, Hache_de_Hyoga, Lame_de_l_eternite, Medaillon_de_Troie, Plastron_de_Gaia
Deplacement	Bottes_d_Hermes, Bottes_de_chevalier, Bottes_de_paladin, Bottes_de_protection, Bottes_enchantees, Bottes_soniques
Jungle	Brise_ardente, Faucheuse_d_ames, Leviathan, Malediction_de_Loki
Magie	Agonie_de_Berith, Apocalypse, Aura_intense, Benediction_de_Rhea, Diademe_d_Hecate, Espadon, Orbe_arctique, Orbre_d_energie, Sceptre_explosif, Tome_sacre, Vengeance_glaciale
Soutien_	Anneau_demoniaque, Bracelet_purificateur, Essence_du_vnt et Marque_du_seigneur_des_mers

Il est à noter que la classe **Route** ne possède pas d’instances.

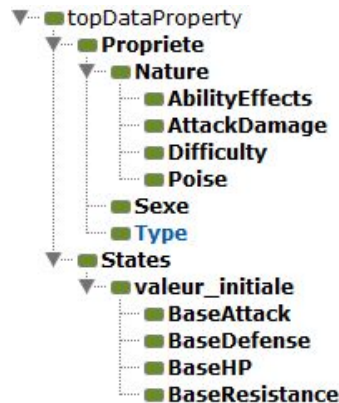
4.4. Création de propriétés et d’attributs

Dans notre ontologie, nous définissons les caractéristiques des héros comme « Data property », et les relations entre la classe **Heros** et la classe **Equipement** ainsi que celles entre la classe **Heros** et la classe **Route** comme « Object property ».

■ Data Property

Pour ce qui est de « Data Property », nous définissons d'abord la « **Propriete** » qui contient les traits principaux des héros : « **Natrue** », « **Sexe** » et « **Type** ». Sous la « **Nature** » sont inclus « **AbilityEffects** », « **AttackDamage** », « **Difficulty** » et « **Poise** ».

Ensuite, « **States** » est créé pour préciser les valeurs initiales des héros : « **HP** » (points de santé), « **BaseAttack** », « **BaseDefense** » et « **BaseResistance** ».



■ Objet Property

En ce qui concerne « Objet Property », nous en avons créé trois.



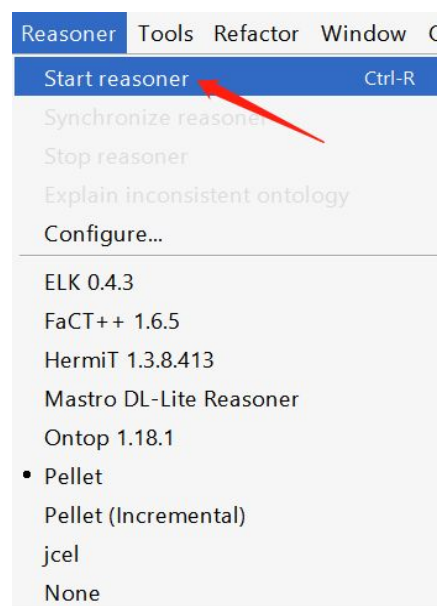
Nous mettons en relation les classes par l'intermédiaire de « Objet Property ». La propriété d'objet « *emprunte* » a pour « Ranges » la classe « **Route** », et pour « Domains », la classe « **Heros** », ce qui nous permet de savoir que le héros choisi doit emprunter quelle route dans un combat. La propriété d'objet « *utilise* » relie la classe « **Equipement** » et la classe « **Heros** » et est conçu pour la recommandation des équipements. Finalement, une autre propriété « *utilisePar* » est à l'inverse de la propriété « *utilise* ».

Il est à noter que ces trois propriétés ont toutes une caractéristique « Asymmetric ». Les relations entre la domaine et la range ne peuvent pas être inversées et elles demeurent toujours dans l'ordre que nous avons défini.

5. Vérification de consistance

Une fois l'ontologie construite, il est impératif de vérifier sa consistance pour qu'elle ne génère pas de contradictions, ce qui constitue une étape importante du maintien de l'ontologie. Nous utilisons ainsi « Pellet », raisonneur (reasoner) qui nous permet la vérification de consistance.

L'opération effectuée comme illustré ci-après, aucun message d'erreur ne s'affiche. Notre ontologie s'avère être bien consistante.



6. Visualisation de l'ontologie

Après avoir vérifié sa consistance, nous aimerions en outre visualiser notre ontologie afin d'en avoir une vue panoramique, ce qui nous amène à bien comprendre la structure des éléments fondamentaux du jeu Arena of Valor.

Pour ce faire, nous utilisons un plugin nommé « VOWL² » de *Protégé*. Il faut le télécharger et ensuite le placer dans le dossier « plugins » du répertoire d'installation de *Protégé*. Sur l'interface graphique du logiciel, on clique comme montré ci-dessous pour afficher l'onglet « VOWL ».

² Disponible sur : <http://vowl.visualdataweb.org/protegevowl.html>

Ce langage conçu pour requêter des données RDF, nous permet donc de requêter sur toutes les ontologies puisque OWL est basé sur RDF.

7.1. Point d'accès SPARQL à l'aide Virtuoso

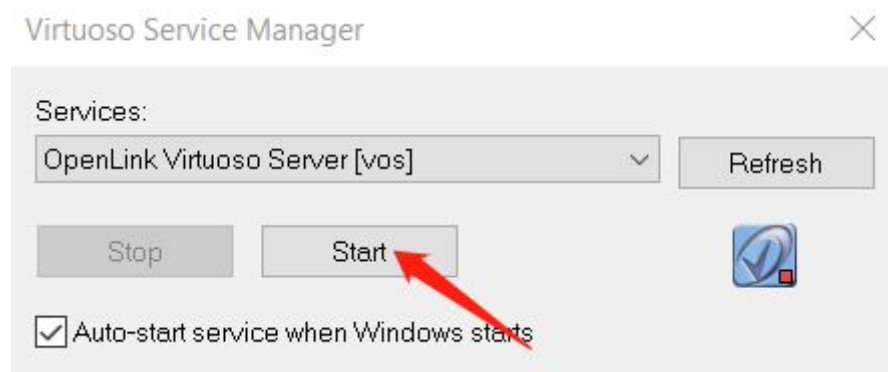
Pour tester la portée de notre ontologie, nous utilisons l'édition open-source de l'OpenLink Virtuoso, qui est un serveur multiplateforme évolutif qui sert à générer des bases de données. Ce dernier, implémenté avec le langage C, nous permet de créer le point d'accès.

■ Installation

Au lieu de nous référer sur Github et de recourir à l'invite de commandes selon les instructions que nous avons vus en cours, ce qui nous est paru compliqué, nous avons trouvé directement le fichier exécutable d'installation disponible sur le site *SourceForge*³.

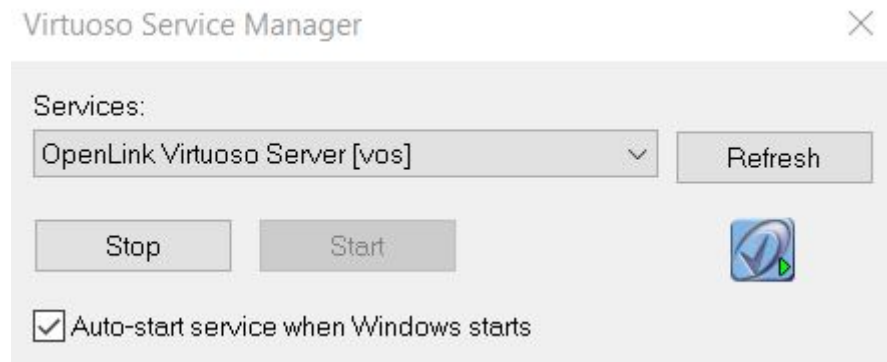
■ Configuration

Une fois le serveur installé, une configuration nous est nécessaire afin de requêter sur notre propre ontologie. Tout d'abord, il faut aller sur l'interface graphique de Virtuoso en cliquant sur son icône figurant dans la barre de tâches.



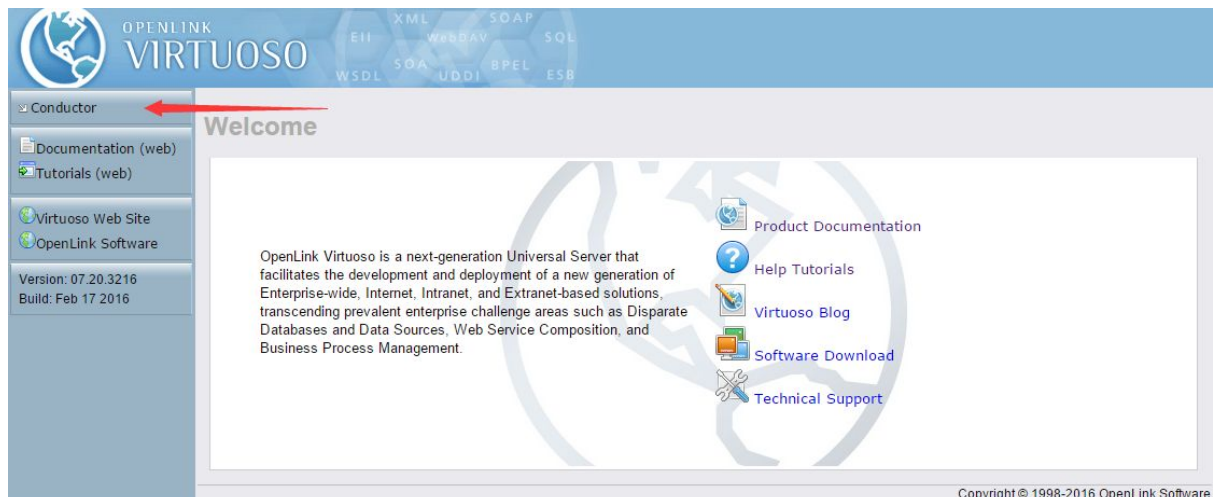
En cliquant ensuite sur « Start », nous activons le serveur.

³ Disponible sur : <https://sourceforge.net/projects/virtuoso/files/virtuoso/>



Cependant, rien ne se passera si l'on attend qu'il réagisse. Il faut en fait aller sur un navigateur qui soit compatible avec Virtuoso. En ce qui nous concerne, la configuration se fait sur Microsoft Edge.

Après avoir lancé le navigateur, il faut saisir dans la barre d'adresse le lien du serveur Virtuoso : « <http://localhost:8890/> ».



Voici l'interface graphique du site nous permettant de créer un point d'accès SPARQL.

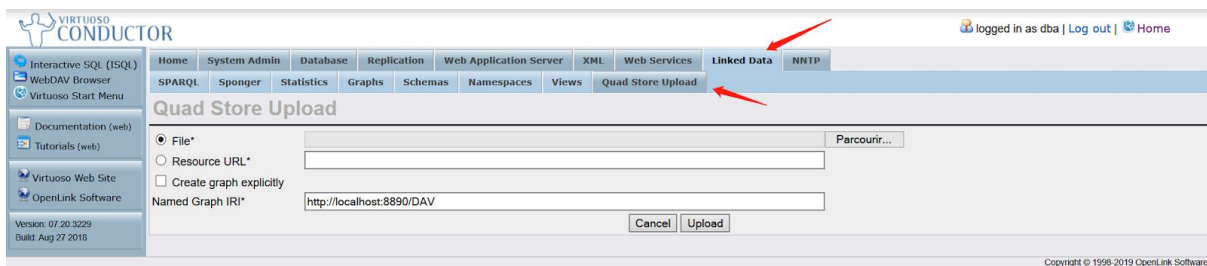
■ Création du point d'accès

Une connexion au serveur est requise avant de faire fonctionner le « Conductor ».

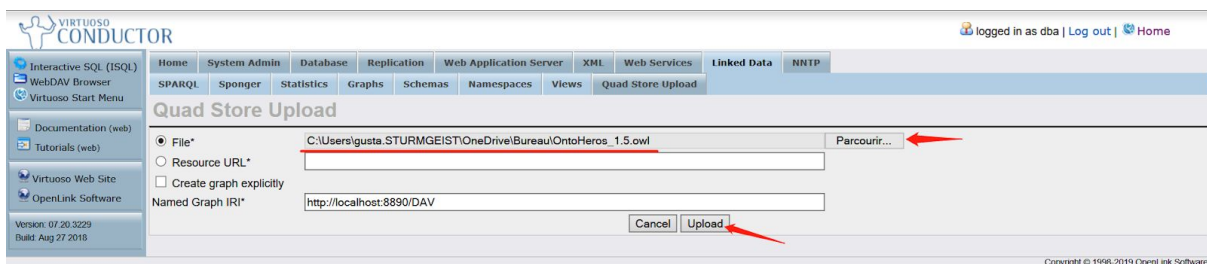


Nous pouvons nous connecter grâce à l'identifiant et au mot de passe fournis en cours, lesquels sont tous les deux : « dba », qui signifie « Database Administrator » (Administrateur de base de données).

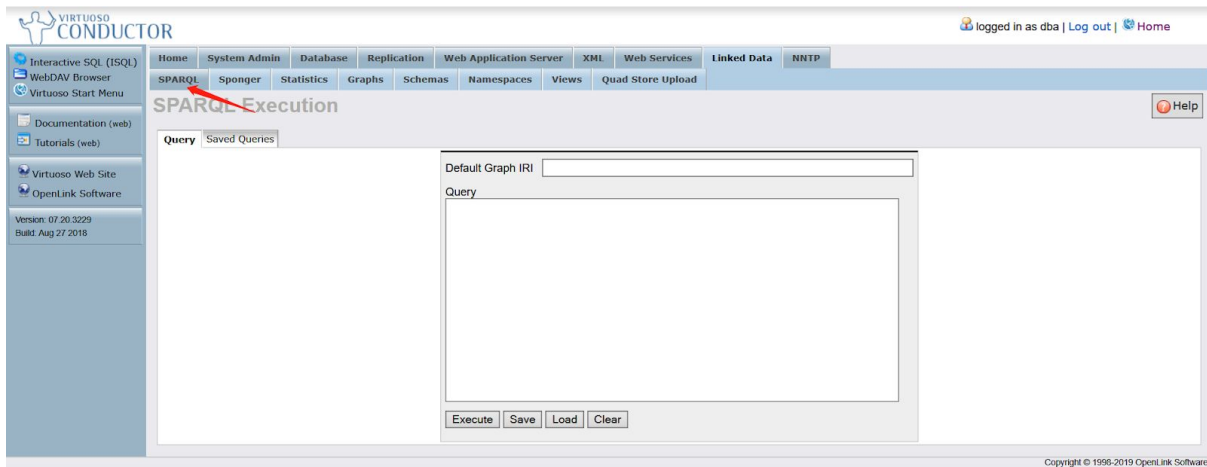
Après nous être connectés, il faut aller sur « Linked Data ».



Ensuite, nous cliquons sur « Quad Store Upload » où s'affiche une case destinée à télécharger notre propre fichier OWL nommé « OntoHeros_1.6.owl » qui est la dernière version.



En cliquant sur « Upload », notre ontologie est bien chargée sur le serveur.



Nous pouvons alors y exécuter des requête SPARQL.

7.2. Requêtes testées et résultats

- ❖ Qu'est-ce que l'on doit prendre comme équipements si on choisit Murad comme héros ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?EquipDeMurad
WHERE { heros:Murad heros:utilise ?EquipDeMurad }
```

Execute Save Load Clear

EquipDeMurad
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Armes_omni
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Bottes_de_chevalier
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Croc_de_Fenrir
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Destructeur_de_formation
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Faucheuse_d_ames
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Muramasa

❖ De quel type est le héros Tulen ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?TypeDeTulen
WHERE { heros:Tulen heros:Type ?TypeDeTulen }
```

Execute Save Load Clear

TypeDeTulen
"Finisseur"
"Persécution"

❖ On voudrait jouer un guerrier de type finisseur. Que propose-t-elle ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?guerrierFinisseur
WHERE{
?guerrierFinisseur rdf:type heros:Guerrier.
?guerrierFinisseur heros:Type "Finisseur"
}
```

Execute Save Load Clear

guerrierFinisseur
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Airi
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Batman
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Maloch
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Murad
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Rourke
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Ryoma
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Zanis

- ❖ On voudrait choisir un héros de type mage qui ne soit pas si fragile (ne pas avoir de « Poise » faible), est-ce qu'il existe ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?MageRobuste ?poise
WHERE
{
  ?MageRobuste rdf:type heros:Mage.
  ?MageRobuste heros:Poise ?poise
  FILTER regex(?poise, "moyen|assezFort|fort|tresFort")
}
```

Execute Save Load Clear

MageRobuste **poise**

Nous avons testé sur Virtuoso cette requête qui devrait fonctionner puisque sa syntaxe est *a priori* correcte. Néanmoins, pour certaines raisons inconnues, aucun résultat ne s'affiche.

SPARQL query:	
<pre>PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> PREFIX heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#> SELECT ?MageRobuste ?poise WHERE { ?MageRobuste rdf:type heros:Mage. ?MageRobuste heros:Poise ?poise FILTER regex(?poise, "moyen assezFort fort tresFort") }</pre>	
MageRobuste	poise
Annette	"moyen"
The_Flash	"fort"
Marja	"moyen"
Sephera	"fort"
Gildur	"fort"

Nous l'avons testé une autre fois sur *Protégé* et cela a fonctionné, ce qui prouve que notre requête est bel et bien correcte. Cependant, nous n'arrivons toujours pas à expliquer la raison.

❖ On veut choisir une héroïne de type mage. Que propose-t-elle ?

Default Graph IRI

Query

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?heroine
WHERE
{
  ?heroine heros:Sexe "feminin".
  ?heroine rdf:type heros:Mage .
}

```

Execute
Save
Load
Clear

heroine
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Alice
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Annette
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Diaochan
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Illumia
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Kahlia
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Kixi
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Lauriel
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Liliana
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Marja
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Natalya
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Sephera
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Veera

- ❖ On veut choisir un héros qui soit facile à manipuler. Que propose-t-elle ?

Default Graph IRI

Query

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>
SELECT ?herosfacile

WHERE
{?herosfacile heros:Difficulty "facile" .
}
```

Execute Save Load Clear

herosfacile
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Amily
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Arthur
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Baldum
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Kil'Groth
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Marja
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Mina
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Omega
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Omen
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Ryoma
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Sephera
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Taara
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Tel'Annas
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Thane
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Toro
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Valhein
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Veera
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Zanis

- ❖ Est-ce que le héros Raz emprunte la voie de dragon ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

ASK
{ heros:Raz heros:emprunte heros:voieDuDragon
}
```

false

- ❖ Alice prend les bottes de paladin comme équipement. Est-ce que c'est correct ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

ASK
{ heros:Alice heros:utilise heros:Bottes de paladin
}
```

true

❖ Le HP d'Arthur est-il supérieur à celui d'Annette ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

ASK
{
heros:Arthur heros:BaseHP ?HPdeArthur.
heros:Annette heros:BaseHP ?HPdeAnnette.
FILTER(?HPdeArthur > ?HPdeAnnette)
}
```

true

❖ Est-ce que tous les mages sont féminins ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

ASK
{ heros:Mage heros:sexe "feminin"
}
```

false

- ❖ Existe-il un héros de type tank qui a une très forte « ability effects » ?

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

ASK
{
?tank rdf:type heros:Tank.
?tank heros:AbilityEffects "tresFort"
}
```

Execute Save Load Clear

true

- ❖ Je voudrais savoir tous les héros de type Assassin dont le base HP dépasse 3300 et leurs niveau de difficulté.

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?assassin ?HP ?difficulte
{
?assassin rdf:type heros:Assassin.
?assassin heros:BaseHP ?HP.
?assassin heros:Difficulty ?difficulte
FILTER(?HP > 3300)
}
```

Execute Save Load Clear

assassin	HP	difficulte
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Amily	3375	"facile"
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Riktor	3465	"assezDifficile"
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#The_Flash	3370	"difficile"
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Wukong	3305	"moyen"

- ❖ On voudrait choisir un héros de type tireur qui est facile à manipuler en regardant une liste montrant leur valeur de « BaseAttack » par ordre croissant.

Default Graph IRI

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX
heros:<http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#>

SELECT ?tireur ?difficulte ?attack
{
  ?tireur rdf:type heros:Tireur.
  ?tireur heros:BaseAttack ?attack.
  ?tireur heros:Difficulty ?difficulte.
}
ORDER BY ?attack
```

Execute Save Load Clear

tireur	difficulte	attack
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Fennik	"difficile"	161
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Rourke	"moyen"	163
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Tel'Annas	"facile"	163
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#The_Joker	"difficile"	165
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Capheny	"moyen"	169
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Moren	"assezDifficile"	169
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Wisp	"assezFacile"	171
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Slimz	"assezFacile"	172
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Violet	"difficile"	172
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Yorn	"difficile"	174
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Valhein	"facile"	178
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Lindis	"assezDifficile"	184
http://www.semanticweb.org/noah/ontologies/2019/2/untitled-ontology-8#Elsu	"difficile"	188

8. Conclusion

Notre ontologie, nommée « OntoHeros » et élaborée conjointement par Rasgouyan Seyedeh, SUN Chen et GAO Shuai, a passé itérativement 18 versions pour arriver finalement à la version finale « OntoHeros_1.8 ». Ce projet, cristallisant en lui toute notre sueur et tous nos temps consacrés, s'avère être fructueux en pouvant répondre à nos besoins. Il mérite donc d'être développé ultérieurement par tous ceux passionnés de ce sujet.

9. Bibliographie et sitographie

- ❑ YANG, Yue, 2018. « Research on Esports and Esports Industry in the New Era ». In: CHINA SPORT SCIENCE vol.38, No.4, 8-21. Pékin : China Institute of Sport Science, mars 2018.
- ❑ Vincenzo Daponte, Gilles Falquet. Une ontologie pour la formalisation et la visualisation des connaissances scientifiques. 29es Journées Francophones d'Ingénierie des Connaissances, IC 2018, AFIA, Jul 2018, Nancy, France. pp.129-136.
- ❑ <https://www.arenaofvalor.com/web2017/herolist.html>
- ❑ http://victoria.nauroy.net/index.php/Groupe_2
- ❑ https://www.researchgate.net/publication/318208041_Creation_d'une_ontologie_en_OWL_avec_Protege_45
- ❑ <http://www.yoyodesign.org/doc/w3c/rdf-sparql-query/>
- ❑ http://www.imgt.org/IMGTEducation/Enseignements/_FR/archives/MEDBioinfo_ontologies-protege_2010.pdf
- ❑ https://fr.wikipedia.org/wiki/Resource_Description_Framework
- ❑ https://en.wikipedia.org/wiki/Arena_of_Vvalor
- ❑ <http://vowl.visualdataweb.org/webvowl.html>
- ❑ https://fr.wikipedia.org/wiki/Web_sémantique
- ❑ <https://www.youtube.com/watch?v=4Wv1z5UWVX0>