# Ingénierie des Connaissances – HMIN 234

Université de Montpellier 2020/2021

Session Pratique

Équipe pédagogique

Sylvie RANWEZ : sylvie.ranwez@mines-ales.fr

Sébastien HARISPE : sebastien.harispe@mines-ales.fr

**Objectif** : Mise en œuvre des techniques et technologies de l'Ingénierie des connaissances pour la réalisation d'un prototype d'application métier.

**Prérequis** : Connaissances basiques en logiques descriptives, notions de TBOX/ABOX, introduction à l'outil de modélisation d'ontologies Protégé.

**Notions connexes** : Modèle de données RDF, langage de requêtage SPARQL, API de programmation (e.g. Jena).

Nous vous demandons, pour un domaine applicatif qui vous est d'intérêt (e.g., musique, sport, informatique…), de proposer une illustration de mise en œuvre des techniques et technologies de l'Ingénierie des Connaissances. Ce projet sera évalué ; il peut être réalisé en binôme ou trinôme. Les séances pratiques proposées dans le cadre de ce cours seront en partie dédiées à la réalisation du projet.

L'illustration demandée amènera :

1. La définition d'un objectif d'application de l'Ingénierie des Connaissances pour un domaine applicatif spécifique. Cet objectif sera dans la plupart des cas exprimé sous la forme de traitements que l'on souhaiterait automatiser, e.g. classification automatique d'instances, requêtes sur le modèle de connaissances. À titre d'exemple, rappelez-vous l'illustration proposée lors de l'introduction de l'outil Protégé pour la recommandation d'auteurs et de livres. L'objectif que vous définirez sera validé et éventuellement complété par l'équipe pédagogique encadrante.

2. Le développement argumenté d'une ontologie de domaine (TBOX) en considération de l'objectif applicatif défini. La définition de cette ontologie nécessitera nécessairement des choix de modélisation que vous devrez être capable de justifier du fait de l'objectif visé. Pour rappel, la TBOX fait référence à la définition des classes et des propriétés (*data* et *object properties*). Votre ontologie sera nécessairement consistante ; utilisez un raisonneur pour tester que cela est bien le cas.

   L'implantation sera réalisée à partir de l'outil Protégé. Il ne vous est pas demandé d'intégrer des modèles de connaissances ou des vocabulaires externes (e.g. ontologies/vocabulaires récupérés sur le Web). Vous pouvez cependant, si vous le souhaitez, effectuer ce travail d'intégration qui complexifiera naturellement le projet – ce travail sera alors considéré lors de l'évaluation. Vous devez cependant utiliser les vocabulaires RDF, RDFS et OWL intégrés par défaut dans Protégé.

3. La récupération ou la génération d'assertions (ABOX). Les faits de la ABOX peuvent être générés à partir de données issues de *scrapping* ou de sources de données diverses (Open

Data, fichiers CSV personnels). Les faits peuvent aussi être totalement artificiels et être générés par un script que vous développerez – ou à l'aide d'un plugin Protégé. Une partie de la ABOX peut être saisie manuellement via Protégé pour les phases de tests ; vous éviterez cependant les saisies trop nombreuses et chronophages. À noter que l'évaluation de votre projet ne sera pas fonction du nombre de faits qui composent votre base de connaissances ; la base de faits doit permettre d'amener une illustration intéressante et pertinente de votre système. On ne recherche pas l'exhaustivité mais une preuve de concept.

4. La mise en place d'un système basé sur l'ontologie qui permette d'illustrer l'utilisation concrète de l'Ingénierie des Connaissances imaginée. Différents développements peuvent être réalisés pour cette partie.
   a) L'utilisation d'un système d'inférence externe à Protégé via l'utilisation d'une API (e.g. Jena, OWL Ready 2) ou d'une Triple Store (e.g. StarDog). Des informations sur les outils qui peuvent être utilisés sont fournies ci-dessous.
   b) La définition d'éventuelles requêtes SPARQL ou exprimées en logiques descriptives (via Protégé par exemple) ;
   c) Le développement d'une interface (Web ou *standalone*) pour interagir avec votre système.

Nous vous demandons dans un premier temps de réaliser les phases suivantes *via* Protégé :

1. Définition de la TBOX,
2. Définition de la ABOX (quelques exemples pour tester votre modèle),
3. Application d'un raisonneur pour tester la validité des inférences produites,
4. Requêtage via SPARQL ou DL query. À noter qu'afin de requêter les faits inférés par un raisonneur vous devez utiliser le plugin Snap SPARQL (le plugin SPARQL par défaut ne permet pas de requêter ces faits).

Vous pouvez dans un second temps traiter les phases 2 et 3 hors de Protégé. Vous enregistrez pour cela votre TBOX en RDF pour la charger via une API ou un triple store. Vous pourrez par la suite charger votre ABOX et interagir avec votre base de connaissances, e.g. effectuer des raisonnements, et des requêtes sur votre modèle.

L'objectif premier de ce projet est de tester votre capacité à (i) mettre en œuvre des concepts de l'Ingénierie des Connaissances de manière argumentée, et (ii) à réaliser une illustration de leur application. Vous devrez notamment être en mesure d'argumenter l'avantage d'une approche tirant parti de l'Ingénierie des Connaissances par rapport à une approche exploitant un SGBD relationnel ou une base de données NoSQL.

**Séances Pratiques**

- 17/02 – Introduction à l'outil Protégé, définition des groupes d'étudiants et des objectifs du projet, réalisation des premiers tests de modélisation.
- 03/03 – Informations complémentaires sur Protégé et des outils de type API de programmation. Session projet avec accompagnement de l'équipe encadrante.
- 10/03 – Informations complémentaires sur Protégé et des outils de type API de programmation. Session projet avec accompagnement de l'équipe encadrante. Présentation d'une première version du système.

- 17/03 – Evaluation par l'équipe encadrante (par groupe, les détails seront précisés en séance).

**Evaluation** :

L'évaluation portera exclusivement sur la réalisation du projet ; à noter cependant que les notions introduites lors du cours nécessitent d'être maitrisées afin de pouvoir argumenter les choix techniques réalisés lors du projet.

L'évaluation sera décomposée en deux phases :

- Une évaluation en présentiel le 17/03 pendant laquelle chaque groupe présentera son projet et sera questionné (évaluation détaillée ci-dessous), 50% de la note.
- Une évaluation sur rendu (cf. ci-dessous). 50% de la note

À noter que les membres d'un même groupe ne seront pas distingués lors de l'évaluation - hormis différences notables lors de l'évaluation orale.

Évaluation du17/03

Lors de la séance du 17/03, chaque groupe aura 15 minutes pour présenter son projet à l'équipe encadrante de manière informelle et interactive – inutile de préparer une présentation formelle avec des supports ; cela ne vous empêche cependant pas de préparer et de structurer votre présentation. Cette présentation sera considérée pour l'évaluation à hauteur de 50% de la note finale ; vous serez notamment amenés à nous présenter les objectifs de votre projet et le système mis en place pour les atteindre. Vous devrez être en mesure d'argumenter les choix techniques concernant la modélisation et l'utilisation des technologies de l'Ingénierie des Connaissances. Nous vous demanderons aussi de réaliser quelques tests, e.g. définitions de concepts supplémentaires, réalisation requête de type DL query et/ou SPARQL (de simples requêtes de type SELECT). L'objectif de cet échange est d'évaluer votre compréhension des technologies et techniques mises en œuvre, leurs limites et d'assurer la paternité du travail réalisé.

Évaluation sur rendu

Nous vous demandons de rédiger un rapport qui présente votre projet et la solution proposée.
Ce rapport doit présenter :

- Le contexte métier considéré et les objectifs du projet (vous pouvez vous baser pour cela sur des scénarios d'utilisation).
- La mise en œuvre adoptée soit :
    - la modélisation (TBOX) et les choix associés, les liens éventuels avec d'autres vocabulaires,
    - La constitution de la ABOX,
    - L'intégration des différentes composantes du système (enrichissement de la base, raisonneur, Triple store, API de programmation).
- Une illustration d'utilisation du système en réponse aux objectifs fixés (e.g. requête SPARQL et/ou DL, exemples de classification automatique).
- Une discussion sur le résultat obtenu en présentant les perspectives envisageables (e.g. couplage avec des vocabulaires externes), ainsi que les avantages et limites de l'Ingénierie des Connaissances rencontrés lors de la réalisation de votre projet.
- **Important** : une discussion sur l'intérêt des ontologies par rapport aux bases de données de type SGBDR/BD No SQL (2 à 3 pages).

Le rapport et autres productions associées à la réalisation du projet doivent être déposés sur le Moodle **avant le 22/03 minuit** – en cas de souci, ils peuvent être envoyés par mail à sylvie.ranwez@mines-ales.fr et sebastien.harispe@mines-ales.fr.

- L'archive de type zip aura pour nom IC_NOM1_NOM2_NOM3.zip ; elle contiendra :
  - Le rapport au format pdf (20 pages maximum),
  - L'ontologie au format RDF/XML,
  - Le code source et les données utiles au projet,
  - Si nécessaire, un fichier readme.txt pour détailler les aspects techniques associés à l'installation de votre système.

**Outils**

Vous êtes libres d'utiliser les outils que vous voulez ; quelques exemples d'outils fréquemment utilisés sont précisés ci-dessous :

- Éditeur d'Ontologies : Protégé https://protege.stanford.edu/products.php

- Librairies /API de programmation :
  - Apache Jena (Java) https://jena.apache.org/. A noter que Jena propose aussi des triples stores et la possibilité d'exposer un SPARQL end-point (serveur SPARQL) extrêmement facilement. Les raisonnements proposés par Jena sont limités (ils ne supportent pas les logiques descriptives les plus complexes, cf. documentation).

  - OWL API (Java) http://owlcs.github.io/owlapi/. Permet d'utiliser des raisonneur supportant des logiques descriptives complexes (ne propose pas d'interface SPARQL, mais supporte SPARQL-DL).

  - RDFLib (Python) https://github.com/RDFLib/rdflib Package Python pour la manipulation de RDF (parsers, implantation de SPARQL 1.1)

  - Python Owlready2 https://pythonhosted.org/Owlready2/ Package Python pour manipuler du OWL.

- Raisonneur
  - Fact++ http://owl.man.ac.uk/factplusplus/
  - HermiT http://www.hermit-reasoner.com/command.html

- Système Intégré :
  - StarDog https://www.stardog.com. Consultez la documentation dédiée aux techniques de raisonnement : https://www.stardog.com/docs/
  - Virtuoso : https://virtuoso.openlinksw.com/