Intégration de données hétérogènes dans le domaine de l'agronomie

Team WIMMICS

Arnaud Barbe - M2 WIA
Catherine Faron-Zucker - Maître de stage
Damien Graux - Tuteur de stage

Table of Contents

1 Remerciements

Je remercie Catherine Faron-Zucker de m'avoir acceptée en stage pour poursuivre le travail que j'ai effectué en TER. Grâce à elle, j'ai pu connaître le monde de la recherche et j'ai pu participé à la plénière du projet D2KAB pour présenter, discuter et améliorer mon travail. Je remercie Nadia Yacoubi qui m'a soutenu et guidé tout au long de mon stage de recherche. Je remercie Franck Michel pour son aide et ses conseils qui m'ont bien aidé dans la génération des graphes RDF. Je remercie aussi tous les membres de l'équipe WIMMICS qui m'ont accueilli chaleureusement.

2 Introduction

Le blé est l'une des sources de protéine les plus répandues au monde et dont la récolte est un facteur majeur contre la famine. Les producteurs de blé ont sélectionné et diversifié les variétés de blé afin d'obtenir de meilleurs propriétés : une adaptation au changement de climat et au stress hydrique, une hausse de la production de graines, une tolérance à la sécheresse ou à une forte humidité, une résistance aux pesticides, etc. De nos jours, la recherche de nouvelles graines avec les propriétés voulues est devenue urgente à cause du réchauffement planétaire qui nuit aux bonnes conditions climatiques pour que les récoltes de blé puisse s'améliorer au cours du temps.

Les recherches portant sur la génomique de blé ont progressé depuis le séquençage du génome de blé qui a été publié en 2018. Les scientifiques ont pour objectif de pouvoir repérer des gènes d'intérêt permettant de faire émerger des variétés plus productives et plus résistantes aux maladies et au réchauffement. Une grande partie de ces résultats de recherche sont décrits dans des articles scientifiques et d'autres dans des bases de données relationnelles et fichiers en format brut. Dans ce stage, nous nous intéressons aux données issues de campagnes d'observation sur des micro-parcelles[1]. Plus de 20 années d'observations en France sont accessibles depuis le site de l'équipe URGI (Unité de Recherche Génomique Info)[2], une unité de recherche en bio-informatique faisant partie de l'INRAE et dédiée à la génomique et à la génétique des plantes et de ses bio-agresseurs. Ces observations décrivent les stades de croissances des plantes, la fréquence d'attaque de maladies pour certaines variétés, les localisations géographiques des parcelles de culture, les paramètres météorologiques, etc. Ainsi, pour chaque micro-parcelle, plusieurs variables d'observation sont recensées ce qui rend le volume des données d'observations fournies par l'URGI important (de l'ordre du million d'observations). L'objectif de ce travail est d'intégrer les connaissances mises en ligne dans des bases d'articles scientifiques telles que PubMed avec des données observations collectées lors des investigations effectuées périodiquement et mises aussi en ligne dans des formats semi-structurés. L'objectif serait de permettre le développement de modèles combinant des connaissances émanant de

[1] Champ de blé à destination d'expérimentation
[2] https://urgi.versailles.inra.fr/

la littérature scientifique et des données d'observations. Cependant, le processus d'intégration de ces connaissances hétérogènes n'est pas trivial et nécessite la définition d'une approche de transformation vers un format interopérable.

En fin de ce stage, nous avons pu mettre en place une solution pour le lifting[3] des données d'observation en les transformant en graphe RDF. Cette trans-formation requiert (1) la définition d'un modèle sémantique pour capturer les caractéristiques sur lesquelles portent les observations, leur contexte spatio-temporel, etc., et (2) la définition d'un ensemble de règles de transformation basé sur un langage de mapping générique. La solution réutilise l'outil Morph-xr2RML développé au sein de l'équipe WIMMICS, pour la transformation des données semi-structurées en données RDF en considérant les règles définies. Le résultat est un graphe RDF dans lequel des connaissances hétérogènes provenant à la fois de bases génomiques, de la littérature scientifique et de données d'observations sont sémantiquement décrites et intégrées.

Mon stage s'est déroulé au sein de l'équipe WIMMICS[4] (Inria, Laboratoire I3S) sous la supervision de Catherine Faron-Zucker, maître de stage, Nadia Yacoubi, chercheure post-doc et Franck Michel, ingénieur de recherche.

3 État de l'art

3.1 Web sémantique et Web de données

La vision du Web sémantique a été proposée pour la première fois par Tim Berners Lee en 2001. L'idée est de permettre aux machines et agents logiciels d'effectuer des traitements automatiques en exploitant les données du Web. Pour cela, un ensemble de technologies et de standards a été proposée formant la pile du web sémantique. Cette dernière inclut une panoplie de langages. Nous en citons quelques uns dans ce qui suit et présentons, un peu plus loin et plus en détails, les technologies utilisées dans ce travail :

- RDF est le format standard de représentation de méta-données dans le cadre du Web sémantique.
- Le schema RDF-S offre un vocabulaire minimaliste en comparaison avec OWL pour la définition d'ontologie.
- Le langage OWL (Ontology Web Language) est le langage standard de définition d'ontologies. Basé sur un formalisme logique, OWL permet de déclarer des classes, des relations entre classes, des propriétés, des axiomes et des restrictions.
- SKOS est une recommandation du W3C s'appuyant sur le langage RDF pour la déclaration de vocabulaires contrôlés, Thesaurus et taxonomie.

[3] Définition d'un modèle de représentation des données et transformations au format RDF

[4] Web-Instrumented Man-Machine Interactions, Communities and Semantics: https://team.inria.fr/wimmics/

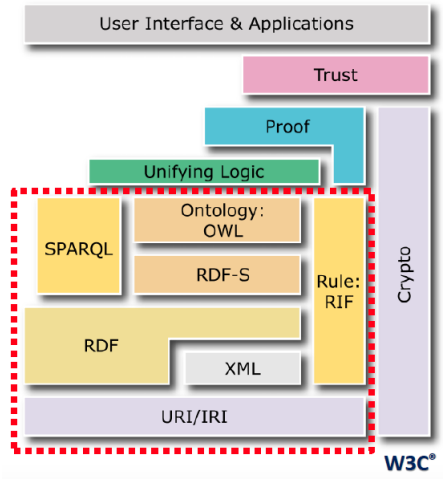

Fig. 2: Pile du web sémantique

La notion d'ontologie est au coeur du Web sémantique, elle représente les connaissances d'un domaine particulier sous forme d'un ensemble structuré de classes, de propriétés et d'axiomes.

Le Web de données ou *Linked Open Data* est l'adoption d'un certain nombre de technologies du Web sémantique pour publier des données sur le Web en adoptant un ensemble de bonnes pratiques.

3.2 RDF, RDF-S et SPARQL

Le *Ressource Description Framework* [RDF], développé par le W3C, est un langage de description de ressources sous forme de triplets (sujet predicat objet) qui permet de former un graphe RDF. Le sujet d'un triplet est obligatoirement un URI, le prédicat d'un triplet est aussi un URI et l'objet d'un triplet peut être soit un URI, soit une valeur littérale telle qu'une chaîne de caractères ou un nombre, ou encore un noeud blanc. Plusieurs syntaxes existent pour le langage RDF. Nous adoptons dans ce rapport la syntaxe Turtle.

Listing 1.1: Exemple de code RDF

```
<http://exemple.com/professor/Susane> rdf:type
    <http://exemple.com/Professor> ;
   <http://exemple.com/teachAt> <http://exemple.com/building/Valrose> .

<http://exemple.com/building/Valrose> a <http://exemple.com/Building> .
```

L'écriture des URI peut être très contraignante. Toutefois, Il est possible de la simplifier en utilisant des préfixes associés à des espaces de noms, que l'on

déclare en début de fichier. Le choix du préfixe est fait par l'utilisateur final tel que le montre l'exemple du listing 1.2.

Listing 1.2: Exemple de RDF avec préfixe

```
@prefix schema: <http://exemple.com/> .
@prefix professor: <http://exemple.com/professor/> .
@prefix building: <http://exemple.com/building/> .

professor:Susane a schema:Professor ;
    schema:teachAt building:Valrose .

building:Valrose a schema:Building .
```

RDF-S est un langage de définition d'ontologies avec un niveau d'expressivité minimal en comparaison avec OWL. Un schéma RDF-S est formé de classes et de propriétés déclarées dans le langage RDF. L'intérêt est de pouvoir typer les ressources et déclarer une hiérarchie de classes et de propriétés ainsi que des signatures de propriétés.

Listing 1.3: Exemple de vocabulaire RDFS

```
<http://exemple.com/Job> a rdfs:Class .
<http://exemple.com/Building> a rdfs:Class .

<http://exemple.com/Professor> a rdfs:Class;
    rdfs:subClassOf <http://exemple.com/Job>.

<http://exemple.com/teachAt> a rdf:property ;
    rdfs:domain <http://exemple.com/Professor> ;
    rdfs:range <http://exemple.com/Building> .
```

SPARQL est un langage d'interrogation et de manipulation de données RDF. Une requête SPARQL de type SELECT exprime des patterns de triplets contenant des variables. Avec SPARQL, il est possible de modifier un graphe à l'aide des instructions INSERT et DELETE qui respectivement ajoute ou supprime des triplets dans un graphe. Il est possible de construire un graphe à l'aide de l'instruction CONSTRUCT. La requête du listing 1.4 permet de lister tous les triplets d'un graphe RDF.

Listing 1.4: Exemple de requête SPARQL

```
select * where{
    ?sujet ?property ?object .
}
```

3.3 R2RML, xR2RML et morph-xR2RML

Le langage R2RML [R2RML] est un langage déclaratif permettant d'exprimer des règles de mapping pour transformer des données provenant de bases de données relationnelle en graphe RDF.

xR2RML est une extension de R2RML qui offre la possibilité de lifter des données structurées, par exemple des données XML, des fichiers CSV et des documents JSON. Le logiciel morph-xR2RML permet de générer des graphes de connaissances RDF à l'aide de règles de mapping écrites elles-même en RDF, plus précisément en xR2RML[5]. Les règles de mapping regroupent des patrons de génération de triplets RDF qui respectent la structure définie dans la spécification[6]:

- Une règle de mapping est une instance de *rr:TripleMap* et est identifiée par un nom.
- Une source logique (ie. *logical source*) qui permet de renseigner quel type de base de données et quelle table nous donne les données.
- Un unique sujet de type *subjectMap* qui décrit le patron de génération des URIs des sujet de triplets RDF.
- Un ou plusieurs *predicateObjectMap* qui permet d'exprimer les patrons de génération des objets des triplets.

Nous utilisons dans ce travail l'outil morph-xr2RML développé au sein de l'équipe Wimmics permettant de transformer des données stockées dans une base MongoDB sous forme de documents JSON.

4 Description du travail réalisé

Les objectifs à atteindre durant les 6 mois de stages sont listés ci-dessous:

- Enrichissement du graphe de connaissances RDF WheatKG construit a partir des annotations d'entités agronomiques extraites automatiquement d'articles scientifiques, qui clôturera le travail réalisé au cours du projet de fin d'année,
- Modélisation et refonte de l'ontologie Crop Ontology 321 (CO_321),
- Alignement de l'ontologie CO_321 avec l'ontologie WTO[7] (Wheat Treat Ontology), ontologie décrivant les traits et phénotypes de blé que l'on trouve dans la littérature scientifique,
- Modélisation du graphe de connaissances d'observations issus des campagnes d'investigation de l'URGI sur différents sites de champs de blé en France,
- Validation du graphe de connaissances en se basant sur un ensemble de requêtes métiers fournies par un expert du domaine,
- Mise en place d'un endpoint SPARQL pour rendre accessibles les graphes de connaissances construit.

[5] xR2RML: https://hal.archives-ouvertes.fr/hal-01066663
[6] https://www.i3s.unice.fr/~fmichel/xr2rml_specification_v5.html
[7] http://agroportal.lirmm.fr/ontologies/WHEATPHENOTYPE

L'objectif est d'obtenir un graphe de connaissances de données agronomiques sur le blé. Ce graphe s'appelle le WheatKG (Wheat Knowledge graphe) et s'inscrit dans la base de connaissances du projet D2KAB.

4.1 Planning du stage

Les 3 premières semaines ont été consacrées à compléter le travail réalisé dans le cadre du projet de fin d'études (TER). Les mois de mai à juillet ont été consacrés à la modélisation et au développement du graphe de connaissances RDF des données d'observations fournies par l'URGI. L'objectif intermédiaire était de proposer un modèle qui a été présenté aux journées plénières 2022 du projet D2kab. Les mois de juillet et août ont été consacrés à la refonte de la Crop Ontology 321 et son alignement avec l'ontologie WTO. La mise en place du SPARQL endpoint et de l'interface utilisateur sera réalisé au cours du mois de septembre.

Le produit du stage est disponible sur le github privé de l'équipe WIMMICS à l'adresse https://github.com/Wimmics/d2kab-wheat-kg.

4.2 Enrichissement des annotations relatives aux articles scientifiques

Un travail antérieur portant sur le lifting d'annotations issues d'articles scientifiques avait été mené durant mon projet de fin d'études (TER). L'objectif était de mettre en place un graphe de connaissances RDF permettant de décrire, structurer et intégrer des annotations reposant sur des entités nommées (EN) extraites automatiquement par l'outil Alvis NLP à partir de publications scientifiques portant sur la génétique et le phénotypage de blé. Ces entités nommées se réfèrent à la fois à des noms de gènes, traits, phénotypes, marqueurs génétiques, variétés et taxons impliqués dans la culture du blé. En plus de la tâche de reconnaissance et d'extraction d'EN[8] à partir des résumés d'articles, la plateforme AlvisNLP permet de faire correspondre des entités pré-définies dans des ontologies et des vocabulaires du domaine aux mentions détectées dans les textes (i.e., Liage d'entités). À titre d'exemple, pour les connaissances phénotypiques, l'ontologie WTO (Wheat Trait and Phenotype Ontology) a été utilisée pour le liage des phénotypes et traits dont les mentions ont été détectées dans les résumés d'articles. La figure 3 présente la détection des entités nommées dans un corpus. Une fois extraites et liées, les entités sont stockées dans un format brut rendant difficile leur réutilisation par les chercheurs pour explorer les contextes de co-occurrence entre entités dans le texte. Notre objectif a été de les transformer (lifter) dans un format compatible avec les standards de publication de données liées (RDF) les résultats sorties d'AlvisNLP afin de construire un graphe de connaissances dans lequel des entités provenant à la fois des ontologies du domaine et d'articles scientifiques sont sémantiquement décrites et intégrées. Nous avons ensuite enrichi le graphe RDF avec certaines méta-données relatives aux articles

[8] Entités nommées

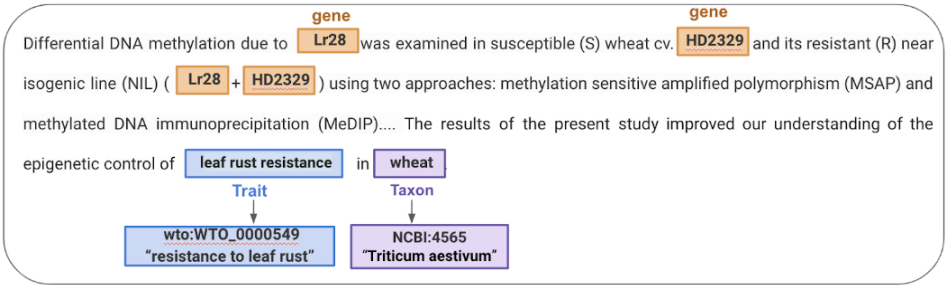

Fig. 3: Reconnaissance d'entités nommées

annotés comme les auteurs, l'année de publication, le journal de parution, etc. Ces méta-données sont accessibles sur le site PubMed[9][10]. La récupération de ces données s'est fait à l'aide d'un SPARQL micro-service [SPµSrv]. La requête du listing 1.5 récupère et construit une ressource pour un identifiant PubMed donné, dans cet exemple "27607596".

Listing 1.5: Requête CONSTRUCT SPARQL µService

```
prefix bibo: <http://purl.org/ontology/bibo/>
CONSTRUCT WHERE {
    ?article bibo:pmid "27607596"; ?p ?o.
}
```

Ces méta-données sont récupérées au format Turtle et sont directement intégrées au graphe de connaissances. Le listing 1.11[11] propose un exemple complet de retour du micro-Service SPARQL[12]. Le graphe de connaissances comporte 8473 articles. Afin de récupérer l'intégralité des méta-données des articles, un script https://github.com/Wimmics/d2kab-wheat-kg/blob/main/Script/alvisNLP_documents.py python a été défini afin de lancer d'une manière itérative ces requêtes. Ce script se base sur le module python SPARQLWrapper [13] qui encapsulent des fonctions permettant d'interroger un SPARQL Endpoint et de convertir le résultat dans le format que l'on souhaite, dans notre cas le format Turtle.

Ainsi, Un ensemble de triplets sérialisés en format Turtle est retourné par le micro-service pour chaque article de l'ensemble d'articles, tout en respectant la hiérarchie des résultats en sortie d'AlvisNLP qui s'organise en différents répertoires contenant chacun 3 fichiers CSV dont un comprenant les identifiants PubMed des articles tel qu'utilisé dans la requête CONSTRUCT du list-

[9] PubMed: https://pubmed.ncbi.nlm.nih.gov/
[10] Aucun export en format RDF disponible
[11] Ce listing est placée en annexe.
[12] SPARQL micro-service: https://sparql-micro-services.org/service/pubmed/getArticleByPMId_sd/
[13] https://pypi.org/project/SPARQLWrapper/

ing 1.5. Le choix de générer plusieurs fichiers est dû aux contraintes de temps d'exécution des requêtes. Le temps d'exécution peut varier entre 1 seconde et 1 minute pour récupérer le résultat en fonction de la disponibilité du endpoint SPARQL. Générer l'intégralité des ressources en une seule fois prend dans le meilleur des cas 8473 / 60 soit 141 minutes. En disposant de plusieurs fichiers, il est plus aisé de relancer le script uniquement sur un fichier en particulier que sur l'ensemble des données. Pour consolider la génération, quelques lignes de gestion d'erreur ont été ajoutées au script. Cette gestion a permis de corriger les erreurs qui empêchaient l'exécution des requêtes SPARQL comme la présence de caractères non autorisés dans les URIs du DOI provoquant une erreur HTTP 500, la durée des articles mis en cache pour la récupération de ces derniers depuis le endpoint SPARQL provoquant une erreur HTTP 404 car non présent dans le cache et enfin la gestion des requêtes mal formées provoquant une erreur 400. Tous les articles déclenchant l'une de ces erreurs n'interrompent pas l'exécution du script et les identifiants des articles sont stockés dans un fichier externe pour pouvoir être récupérés ultérieurement.

De plus, un traitement supplémentaire a aussi été effectué sur les dates de parution des articles scientifiques. Lors de la récupération des méta-données depuis le SPARQL micro-service, les dates étaient des littéraux typés `xsd:string` et le format des dates était inconsistant. Par exemple, la date de parution d'un article pouvait être "Sep 2015" pour un article A, "Sep – Nov 2002" pour un article B, "15 Aout 2019" pour un article C. Pour palier aux différents formats de dates, le choix a été fait de ne considérer que l'année de parution et d'utiliser le type `xsd:gYear` pour chaque date. Ce choix est en accord avec les cas d'utilisation du graphe de connaissances, seule l'année de parution est pertinente dans le filtrage des articles.

Le pipeline de génération du graphe de connaissances a été mis à jour pour la partie du lifting de données provenant d'articles scientifiques. Ses différentes étapes sont illustrées dans la figure 4.

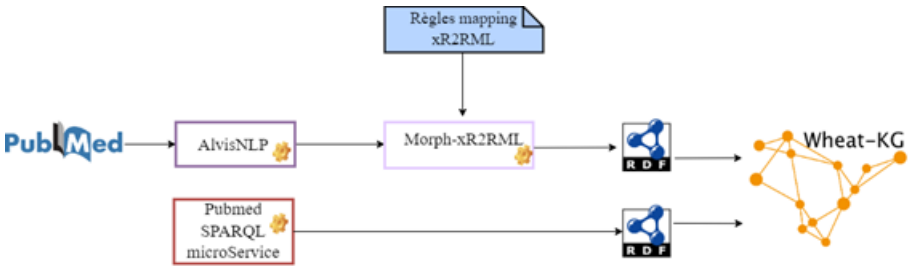

Fig. 4: Pipeline de transformation des données issues de la littérature scientifique

4.3 Modélisation et transformation de la refonte de l'ontologie CO321

Pour compléter le travail effectué et décrit précédemment dans la section 4.2, nous nous sommes intéressés à un nouveau jeu de données issu des campagnes d'investigation menées par l'URGI (Unité de Recherche Génomique Informatique)[14]. Les travaux de l'URGI s'alignent avec les initiatives internationales pour la standardisation des données génomiques de culture de blé et qui ont permis d'aboutir à trois contributions majeures: (1) l'ontologie Crop Ontology (CO), (2) le format d'échange MIAPPE, et (3) l'API Breeding API (BrAPI). Dans cette partie, nous décrivons les modifications apportées à la Crop Ontologie 321, ontologie spécialisé dans la description de variables d'observations de la culture du blé, nécessaire au processus de transformation en graphe de connaissances les données d'observation de culture de blé.

Dans cette section, nous nous focalisons sur la refonte de l'ontologie CO_321 ensuite nous présentons la version actuelle et celle future, telle que repensée par les experts du domaine.

4.3.1 L'ontologie Crop Ontology 321 actuelle La Crop Ontology (CO, www.cropontology.org) définit un modèle de connaissances permettant de capturer la sémantique des traits observés, procédures de mesures et échelles de mesures des observations considérées dans le cadre des étude expérimentales faites par les experts. L'agrégation de ces informations forme une variable d'observation qui permet de référencer ce triplet. Ainsi, une classe CO permet de capturer quatre types de connaissances : (i) la variable d'observation, (ii) le trait cible (phénotypique ou environnemental), (iii) la méthode d'évaluation, méthode de mesure ou d'observation, et (iv) son échelle de mesure ou son unité. Un trait peut être formalisé comme étant l'association d'une entité observée comme une partie de la plante (par exemple, feuille, grain et tige) et un attribut ou une qualité à mesurer ou à observer (couleur, poids et hauteur). La méthode peut être un protocole de phénotypage ou un calcul statistique.

La Crop Ontology fournit une plate-forme collaborative à un nombre croissant de communautés de cultures pour développer une série d'ontologies spécifiques aux espèces (blé, maïs, riz, etc.), d'où la proposition de la branche CO_321 qui correspond au génome de blé. Les investigations pluri-annuelles menées par l'URGI utilisent les variables proposées dans l'ontologie CO_321 pour effectuer leurs séries d'observations.

La version actuelle disponible de CO_321 est disponible sur le site d'Agroportal[15].

Sa définition est fournie en OWL[16] et la dernière version en date est celle du 1er avril 2022. Le problème majeur de cette ontologie est le choix de modélisation qui s'est fait sur la définition des instances de variables, de méthodes, de traits

[14] https://urgi.versailles.inra.fr/
[15] Agroportal CO_321 : http://agroportal.lirmm.fr/ontologies/CO_321
[16] Web Ontology Language : https://www.w3.org/OWL/

et de échelles en tant que classes OWL. Le listing 1.6 montre la définition d'une classe OWL représentant une variable CO_321 extraite de la version actuelle de l'ontologie. Comme le montre le listing 1.6, la classe *InfertTN_Ct_InTllrm2*[17] est déclarée comme une sous-classe de la classe `co_321:Variable` ainsi que 3 autres classes de type `owl:Restriction`. Ces restrictions OWL permettent d'exprimer que les instances de la classe *InfertTN_Ct_InTllrm2* sont reliées à la fois au trait *Infertile tiller number*[18], à la méthode *InfertTN Counting*[19] et à l'échelle *infertile tiller/m2*[20] qui sont elles même déclarées en classes OWL.

Listing 1.6: Extrait de la définition d'une variable de la CO_321

```
<https://cropontology.org/rdf/CO_321:0001245>
  a owl:Class ;
  rdfs:subClassOf [
    a owl:Restriction ;
    owl:someValuesFrom <https://cropontology.org/rdf/CO_321:0000803> ;
    owl:onProperty <https://cropontology.org/rdf/variable_of>
  ], <https://cropontology.org/rdf/Variable>, [
    a owl:Restriction ;
    owl:onProperty <https://cropontology.org/rdf/variable_of> ;
    owl:someValuesFrom <https://cropontology.org/rdf/CO_321:0000644>
  ],
  [
    a owl:Restriction ;
    owl:onProperty <https://cropontology.org/rdf/variable_of> ;
    owl:someValuesFrom <https://cropontology.org/rdf/CO_321:0000801>
  ] ;
  rdfs:label "InfertTN_Ct_InTllrm2"@en ;
  dc:contributor "CIMMYT", "Rosemary Shrestha, Julian Pietragalla" ;
  skos:altLabel "INFERTNO_tll_m2"@en .
```

L'exemple de définition de classe du listing 1.6 montre que la distinction classe/instance dans l'ontologie CO_321 ne respecte pas les standards RDFS et OWL. Une classe représente un concept générique qui permet de classifier les ressources en leur donnant un type alors qu'une instance est une concrétisation d'une ou plusieurs classe(s). Ainsi, Il est difficile de réutiliser l'ontologie CO_321 sans devoir créer des ressources utilisant les classes pour respecter les standard RDF-S et OWL. Par exemple, nous pouvons utiliser la classe *InfertTN_Ct_InTllrm2* du listing précédant, pour typer d'autres ressources. Néanmoins, il s'agit bien d'une variable qui permet de quantifier le nombre de talles infertiles par m2 pour laquelle on a spécifié le trait, la méthode de mesure et l'échelle. De plus, l'ontologie est en cours d'évolution. Des variables sont ajoutées, d'autres sont renommées et spécialisées et de nouveaux identifiants uniques ont été attribués.

[17] https://cropontology.org/rdf/CO_321:0001245
[18] https://cropontology.org/rdf/CO_321:0000644
[19] https://cropontology.org/rdf/CO_321:0000801
[20] https://cropontology.org/rdf/CO_321:0000803

4.3.2 La nouvelle Crop Ontology 321

Pour pallier aux problèmes de modélisation de la CO_321 dans sa version actuelle, nous avons proposé les améliorations suivantes:

- Distinguer les classes génériques de l'ontologie CO_321 qui typeront les instances et les instances de ces classes pour qu'elles soient ré-utilisables,
- Ajouter les propriétés `CO_321:hasTrait`, `CO_321:hasMethod` et `CO_321:hasScale` pour réifier la relation n-aire qui existe entre la variable et les trois entités qui la caractérise.

Les informations fournies par l'INRAE pour la refonte de la CO_321 sont disponibles dans un fichier CSV. Nous nous basons sur ce fichier pour générer la nouvelle version de l'ontologie. Dans sa version actuelle, une variable d'observation CO_321 est un triplet contenant un trait observé (e.g., "plant height", "Grain yield"), une méthode de mesure (e.g. "Mat DS87 DT Computation", "SplN Counting") et son échelle (e.g. "0-9 density scale", "g/plant").

L'ensemble des classes de traits de la CO_321 dans sa nouvelle version inclut 8 classes de traits spécialisées: Abiotic Stress, Agronomical, Biotic Stress, Environmental, Morphological, Phenological, Quality et Other, comme le montre la Figure 5. Une classe trait possède un nom, une description, un ou plusieurs synonymes, une abréviation et une ou plusieurs abréviations synonymes. Pour décrire le nom et les différents synonymes, nous utilisons la propriété skos:prefLabel pour le nom et skos:altLabel pour tous les synonymes, y compris les abréviations qui sont une autre manière de nommer le trait. La description du trait se fait via la propriété `rdfs:comment`. Les traits sont aussi répertoriés en fonction de la localisation sur l'individu où il sont observés, par exemple la plante, la feuille ou la tige. La propriété `CO_321:traitLocation` définit la localisation du trait sur l'individu.

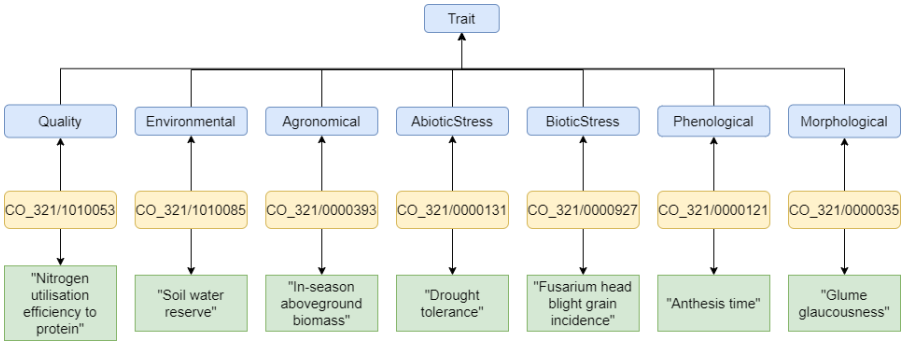

Fig. 5: La classe Trait

Les méthodes de mesure sont découpées en 4 classes de méthodes: Estimation, Computation, Measurement et Counting. La figure 6 présente la hiérarchie de classes de méthodes. Une méthode possède un nom, une description et

une référence bibliographique qui la définit. Le nom est décrit par la propriété `rdfs:label`, la description par la propriété `rdfs:comment` et la référence bibliographique par la propriété `CO_321:reference`.

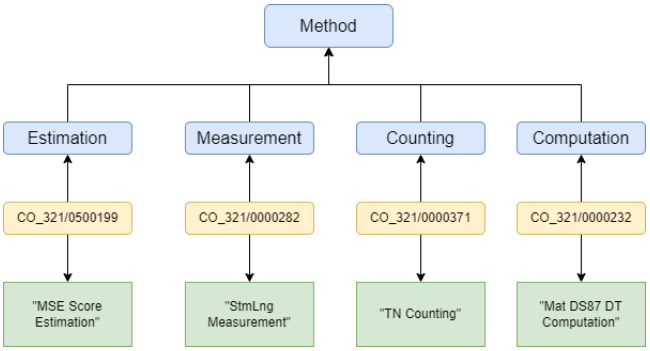

Fig. 6: La classe Method

Enfin, les échelles des mesure sont découpées en 7 classes de méthodes: Ordinal, Date, Duration, Nominal, Time, Numerical et Text. Les types de mesure Duration, Date, Numerical, Time et Date ont une échelle concrète et n'acceptent qu'une valeur. Concernant les types Nominal et Ordinal, les valeurs possibles pour ces échelles sont des valeurs spéciales retranscrites sous forme de numéro entier possible dans un intervalle. Chaque numéro possède une description de sa signification. Pour représenter ces valeurs, une collection skos est définie pour chaque instance d'échelle Ordinal et Nominal.

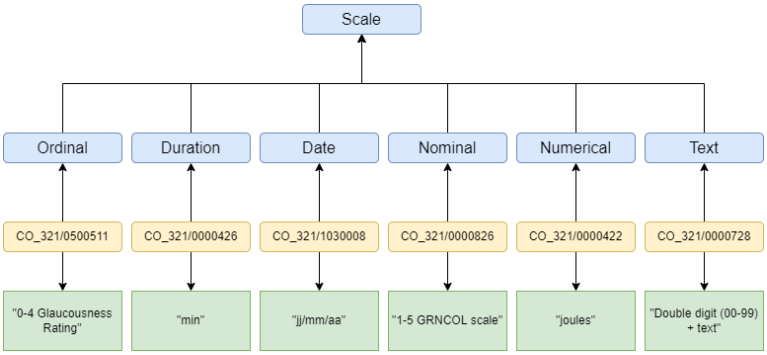

Fig. 7: La classe Scale

Les classes et les 3 nouvelles propriétés de la nouvelle version de l'ontologie CO_321 ont été définies manuellement. De plus, nous avons proposé de re-

grouper l'ensemble des instances des traits, des méthodes, et des échelles définies dans CO_321 dans un Thesaurus SKOS. Le langage SKOS est un standard W3C qui permet de définir des vocabulaires de termes contrôlés et normalisés ce qui s'aligne avec les initiatives internationales visant la formalisation et la standardisation des connaissances portant sur les variables d'observations de la culture de blé. Ainsi, le peuplement de l'ontologie CO_321 dans sa nouvelle version consiste en la création du Thesaurus qui sera généré automatiquement de sorte que chaque concept du Thesaurus sera typé par une classe CO_321. La figure 8 présente un exemple d'utilisation de la refonte CO_321.

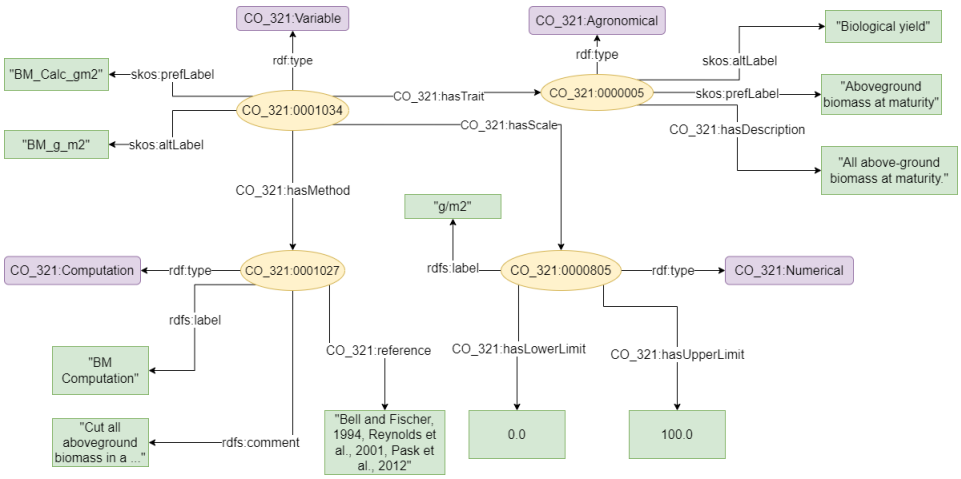

Fig. 8: Graphe RDF d'une variable CO_321 tel que défini dans la nouvelle version

4.3.3 Résultats

Le Thesaurus SKOS généré compte 18 149 triplets RDF. Le temps de prétraitement du CSV contenant la refonte de la CO_321 est inférieur à 1 seconde et le temps de génération par morph-xR2RML est d'environ 4 secondes. Le tableau 1 représente la distribution des triplets du Thesaurus en fonction des classes qu'ils décrivent.

Nb de triplets	18 149
Nb instances de la classe Variable	823
Nb instances de la classe Trait	411
Nb instances de la classe Method	570
Nb instances de la classe Scale	187

Table 1: Caption

L'essentiel du Thesaurus est dominé par les variables d'observation. Un exemple de variable de la CO_321 est présenté dans le listing 1.12. Parmi les différents Traits, nous recensons 60 instances de la classes AgronomicalTrait, 18 instances de la classe AbioticStressTrait, 135 instances de la classe BioticStressTrait, 1 instance de la classe EnvironnementalTrait, 60 instances de la classe MorphologicalTrait, 26 traits PhenologicalTrait, 109 QualityTrait et 2 Other Trait. Un exemple de trait de la CO_321 est présenté dans le listing 1.13. Au niveau des méthodes de mesure, le Thesaurus comporte 66 ComputationMethod, 21 CountingMethod, 318 EstimationMethod et 165 MeasurementMethod. La méthode de mesure par estimation est la plus utilisée lors des observations. Un exemple de méthode de la CO_321 est présenté dans le listing 1.14. Enfin, concernant les échelles de mesure, nous avons 1 DateScale, 5 DurationScale, 9 NominalScale, 89 NumericalScale, 78 OrdinalScale et 5 TextScale. Un exemple de ressource OrdinalScale de la CO_321 avec l'une de ses valeurs est présenté par le listing 1.7.

Listing 1.7: Exemple d'une instance de la classe co_321:Scale CO_321

```
<http://cropontology.org/rdf/CO_321/0000458>
     a      CO_321:OrdinalScale , skos:Collection ;
     skos:member <http://www.cropontology.org/rdf/CO_321/0000458/SE> ,
         <http://www.cropontology.org/rdf/CO_321/0000458/E> ,
         <http://www.cropontology.org/rdf/CO_321/0000458/SS> ,
         <http://www.cropontology.org/rdf/CO_321/0000458/S> ,
         <http://www.cropontology.org/rdf/CO_321/0000458/P> ;
     skos:prefLabel "1-5 GHABIT scale" ;
     CO_321:hasDecimalPlaces
            "1.0" ;
     CO_321:hasLowerLimit
            "0.0" ;
     CO_321:hasUpperLimit
            "0.0" .

<http://www.cropontology.org/rdf/CO_321/0000458/SE>
     a      skos:Concept ;
     rdf:value "SE" ;
     rdfs:comment "semi-erect (SE)" .
```

4.4 Modélisation et transformation des données d'observation de l'URGI

Dans cette section, nous présentons les choix de modélisations effectués pour la transformation des données d'observation de culture de blé provenant des campagnes d'expérimentation de l'URGI. Cette transformation utilise les modifications apporté à la Crop Ontologie 321 introduit dans la section 4.3. Rappelons que les travaux de l'URGI s'alignent avec les initiatives internationales pour la standardisation des données phénotypiques et qui ont permis d'aboutir a

trois contributions majeures: (1) l'ontologie Crop Ontology (CO), (2) le format d'échange MIAPPE, et (3) l'API Breeding API (BrAPI).

MIAPPE[21] (Minimal Information About Plant Phenotyping Experiment), définit un format d'échange standardisé qui permet de spécifier la structure des informations expérimentales telles que l'objectif d'une campagne d'investigation, les auteurs, le lieu et le calendrier, ainsi qu'une description minimale des observations incluses.

D'un autre cote, BrAPI[22] (Breeding API) est un projet d'API[23] RESTful[24] visant à unifier et rendre accessible dans un format interopérable les données provenant des différentes cultures de plantes.

Bien que toutes ces initiatives visent à favoriser le partage et l'interopérabilité des données expérimentales produites par l'URGI et d'autres institutions de recherche a l'échelle mondiale, l'intégration de ces sources avec d'autres produites et/ou décrites dans la littérature scientifique n'est pas un processus trivial vu la complexité des données et leur hétérogénéité. Nous décrivons dans ce qui suit les données sources fournies par l'URGI.

4.4.1 Les données sources

Les données sources consistent en un ensemble de données d'observations structurées et mises en ligne sous différent formats par l'institut URGI. Ces données d'observations font parties d'une étude réalisée par une institution sur un site de culture utilisant un matériel biologique géolocalisé dans une région de France. Le matériel biologique est une graine ayant un numéro d'accession qui est un identifiant homologué d'une espèce de blé. Dans le cadre d'une campagne d'investigation, une étude répond à une question scientifique donnée.

Il existe plusieurs moyen d'accéder à ces données. Le site web de l'URGI Versailles[25] propose une interface web pour naviguer à travers ces données sous forme de champs à renseigner. Il est possible d'exporter en format CSV un fichier contenant les données d'observation d'un site particulier de culture de blé. C'est un export minimaliste où seules les données d'observations et la graine de l'espèce observée sont renseignées.

La BrAPI[26] fournit un point d'accès standardisé aux données. Ainsi, plusieurs points d'entrées sont fournis pour effectuer des requêtes et récupérer les données en format JSON. Cependant, pour pouvoir récupérer un jeu de données complet, il est nécessaire de faire plusieurs appels successifs à l'API. Par exemple, si l'on souhaite récupérer les informations d'une étude et les données d'observations qui lui sont relatives, il faut cibler et récupérer les informations de l'étude, conserver son identifiant, puis récupérer les informations des observations qui ont pour étude, l'identifiant précédemment récupéré. De plus, les données qui nous

[21] www.miappe.org
[22] https://brapi.org/
[23] Interface de programmation d'application.
[24] Ensemble de contraintes architecturales.
[25] URGI Versailles : https://urgi.versailles.inra.fr/ephesis/ephesis/viewer.do
[26] Breeding:https://urgi.versailles.inra.fr/faidare/swagger-ui.html

intéressent sont uniquement les données d'observation du blé. BrAPI recense aussi des données d'observation de culture de maïs et d'autres investigations menées par l'INRAE. Il est donc nécessaire de cibler et filtrer les campagnes d'investigation lors des invocations de l'API BrAPI. Enfin, un export direct depuis la base de données de l'URGI respectant le format d'échange MIAPPE en format CSV traitant directement les fichiers soumis par les scientifiques lors des campagnes d'investigation. Cet export est privé et nous a été fourni par le partenaire de l'INRAE.

4.4.2 Le modèle RDF et l'ontologie cible

Les choix de modélisation de ce travail ont été influencés par le principe de la réutilisation d'ontologies existantes ayant été adoptées par la communauté des chercheurs du domaine pour représenter les données d'observations. De ce fait, nous nous sommes principalement basés sur deux ontologies du domaine: (1) la première est l'ontologie PPEO (Plant and Phenotype Experiment Ontology) [PPEO] développée par les membres de l'INRAE qui est une ontologie décrivant les expérimentations respectant la convention MAIAPPE et leurs observations sur un site de culture, et (2) la seconde est l'ontologie SOSA (Sensor, Observation, Sample and Actuator) [SOSA] qui est un standard W3C proposant un modèle pour la description de données de capteurs. L'intérêt de l'utilisation conjointe de ces deux ontologies est que d'une part SOSA offre des classes et des propriétés génériques permettant de modéliser tout type de données d'observations portant sur des caractéristiques d'intérêt particulières (*Feature of Interest*) pouvant avoir des propriétés observables (*Observable Property*) qui elles feront l'objet de mesure ou d'évaluation selon des procédures particulières; et d'un autre côté, l'ontologie PPEO définit des classes spécifiques au cas d'usage des expérimentations menées dans le cadre d'une culture dans le domaine agronomique.

Dans ce travail, le modèle proposé comporte ainsi deux modules ontologiques: (1) le premier permet de décrire les données relatives à l'expérimentation basée sur l'ontologie PPEO et (2) le second permet de décrire les observations dans cette expérimentation. Le schéma de la figure 9 présente le premier module ontologique réutilisant des classes et des propriétés PPEO pour décrire les différentes méta-données relatives à une expérimentation.

Les observations font partie d'une étude (instance de la classe `ppeo:study`) qui elle-même fait partie d'une campagne d'investigation (instance de la classe `ppeo:investigation`). Elle est menée par un institut de recherche (instance de la classe `ppeo:institution`). Une étude possède un nom, un identifiant, une date de début et fin ainsi qu'une description de l'objectif de l'étude. Le schéma de la figure 10 présente un exemple d'une étude modélisé avec PPEO.

L'institut rattaché à une investigation rassemble les informations de l'établissement en charge de l'étude, la position géographique des champs de blé (instance de la classe *ppeo:location*) ainsi que les personnes (instance de de la classe *ppeo:person*) qui ont participé et leurs rôles (instance de la classe *ppeo:role*) au sein de l'étude. Les observations d'une étude sont organisées en unités d'obser-vations (instance de la classe *ppeo:observation_ unit*) et peuvent être vues comme les différents

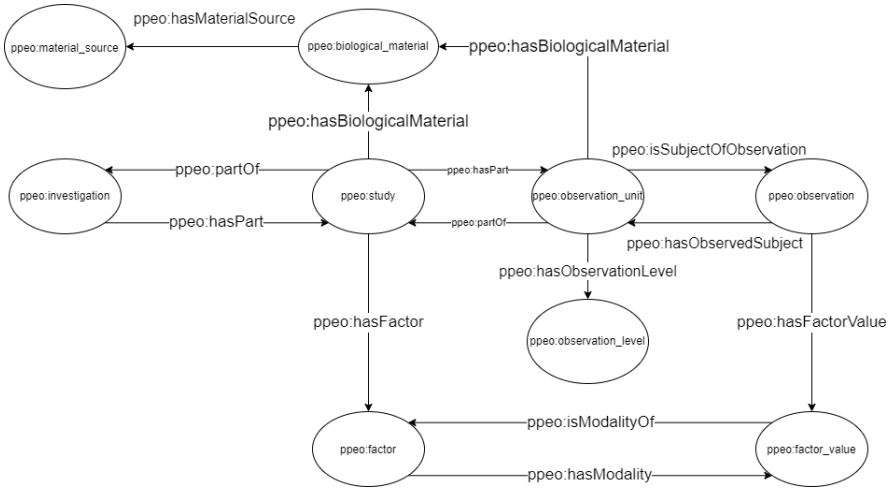

Fig. 9: Modélisation d'une study dans PPEO

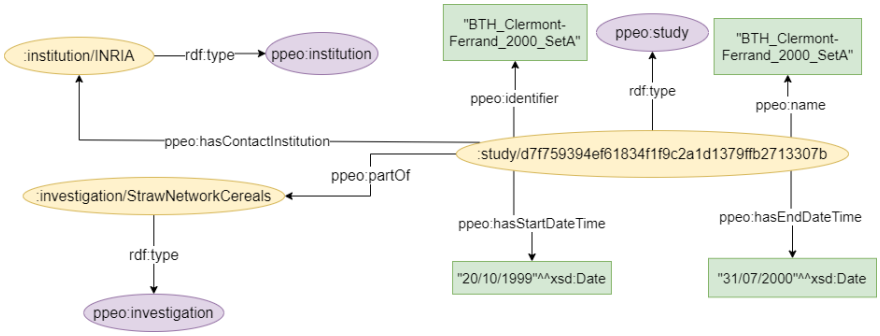

Fig. 10: Le Graphe RDF décrivant une Study

paramètres appliqués sur un site de culture pour un même matériel biologique pour une observation. Un exemple de paramètre d'une unité d'ob-servation est la présence de pesticides ou l'utilisation d'engrais particulier, ainsi que le niveau d'observation qui sera décrit plus tard dans ce rapport. Le matériel biologique associé à une unité d'observation est divisé en deux partie: (1) d'une part il y a la graine de blé utilisé (instance de la classe *ppeo:biological_material*) et (2) la variété ou espèce de blé (instance de la classe *ppeo:material_source* à laquelle cette graine fait partie. La représentation de ces unités d'observation est nécessaire car elles permettent de réifier une relation n-aire. Plusieurs observations sont effectuées sur un échantillon de graine pour une même étude sur un même site. La figure 11 présente un exemple d'unité d'observation.

Une correction a été apportée à l'ontologie PPEO concernant le matériel biologique. En effet, lors de la définition de l'ontologie, une inversion a été

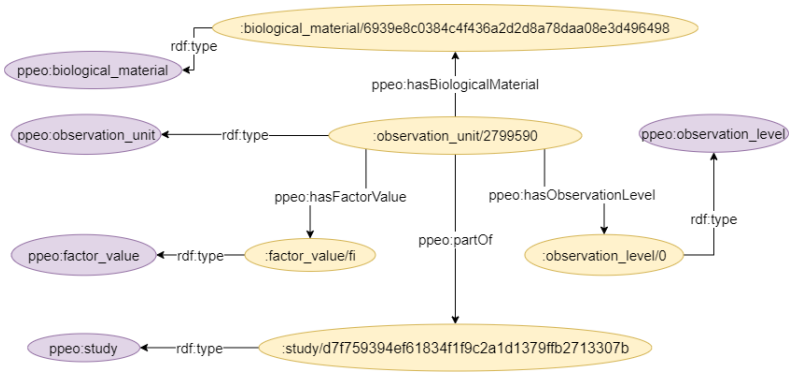

Fig. 11: Graphe RDF représentant une unité d'observation

fait au niveau des propriétés qui définissent les `ppeo:biological_material` et `ppeo:material_source`. L'ontologie PPEO définissait le `ppeo:material_source` comme étant une graine et le `ppeo:biological_material` comme étant l'espèce ou variété du blé. Cette inversion a été mis en évidence lors de la présentation de l'avancée du lifting du dataset par un des membres du projet D2KAB. De plus, aucune propriétés n'avaient été défini pour décrire le numéro de lot, nom de l'accession et numéro de l'accession d'une graine. Par conséquent, une révision de la PPEO a été soumis aux membres fondateurs de l'ontologie en ajoutant les propriétés manquantes ainsi que la correction du domaine et du range des propriétés. La figure 12 présente un exemple de matériel biologique.

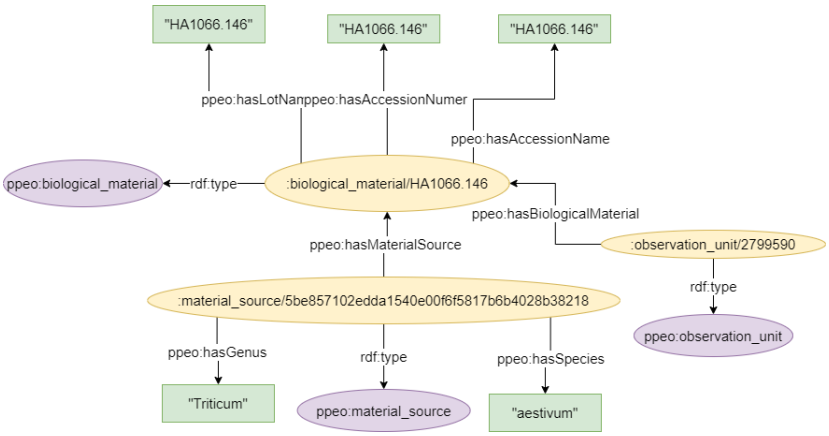

Fig. 12: Exemple de plant material

La sémantique du processus d'expérimentation étant capturée, il est maintenant nécessaire de décrire les observations et leurs résultats. Les informations

précédentes sont décrites à l'aide de l'ontologie PPEO et rassemble les métadonnées relatives au processus d'expérimentation. Concernant les observations, nous avons défini un alignement entre PPEO et SOSA pour pouvoir décrire les observations avec SOSA. Le schéma 13 exprime comment une observation est décrite avec SOSA. Cette alignement n'est pas fait directement dans les ontologies.

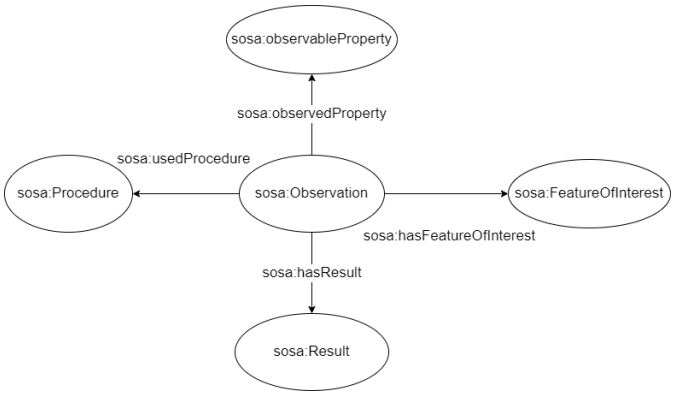

Fig. 13: Modélisation d'une observation dans SOSA

Comme énoncé précédemment, les observations sont rassemblées dans une unité d'observation. Une observation est une instance de la classe *sosa:Observation* et *ppeo:observation*, elle est décrite par les informations : (1) le trait observé qui est la caractéristique d'intérêt de l'observation, (2) de la propriété observable associée, (3) de la procédure effectué pour relever la valeur et (4) de la valeur relevée lors de l'observation. Les observations utilisent les éléments du Thesaurus de la CO_321 défini précédemment en partie 4.3. Le tableau 2 présente l'alignement PPEO, SOSA et CO_321 entre les différentes classes.

La figure 14 propose un exemple d'un observation.

PPEO	SOSA	CO_321
ppeo:observed_variable	sosa:ObservableProperty	CO_321:Variable
ppeo:method	sosa:Procedure	CO_321:Method
ppeo:trait	sosa:FeatureOfInterest	CO_321:Trait
ppeo:scale	-	CO_321:Scale

Table 2: Alignement PPEO, SOSA et CO_321

En résumé, le modèle est scindé en deux parties: (1) une première partie qui consiste à décrire l'expérimentation à l'aide de l'ontologie PPEO et (2) une partie décrivant les observations et leurs valeurs à l'aide de l'ontologie SOSA. Le schéma

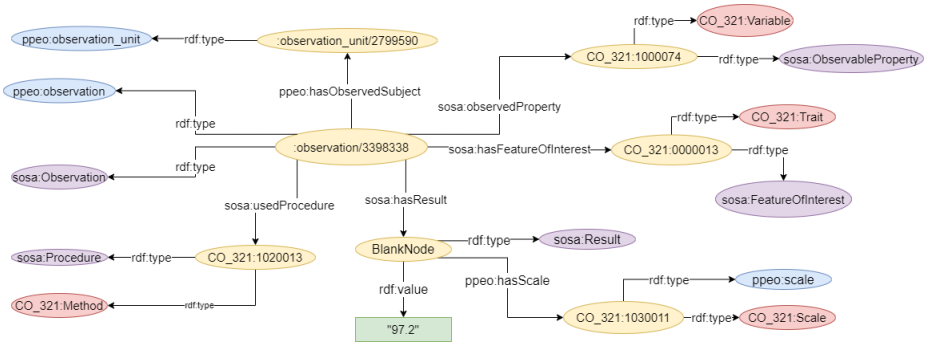

Fig. 14: Exemple d'observation

15 propose une vision global du modèle décrivant les données d'observations du graphe de connaissances.

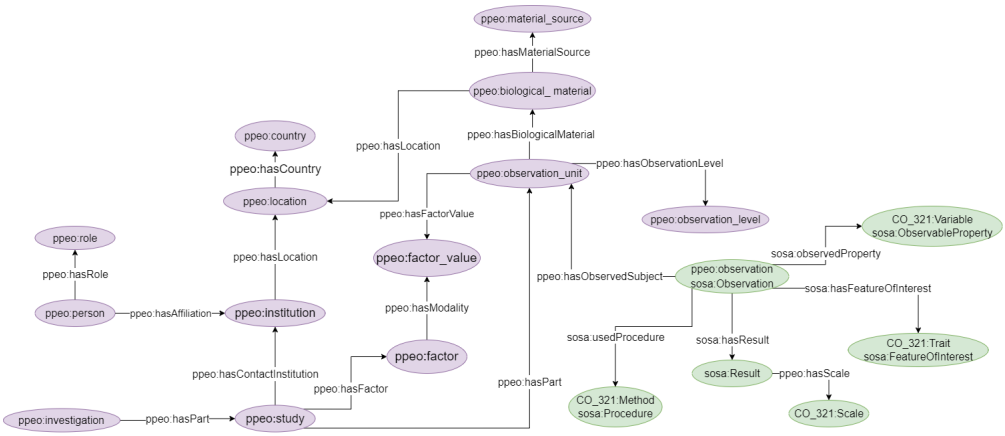

Fig. 15: Modèle global du graphe de connaissances

4.4.3 Pré-traitement et règles de mapping

Pour générer le graphe de connaissances à partir des données d'observation fournies par l'URGI, nous utilisons la même procédure que pour générer le graphe de connaissances d'annotations d'articles scientifiques, c'est-à-dire, effectuer un pré-traitement des fichiers CSV brut pour en former de nouveaux qui seront importés dans une base mongoDB, puis utiliser morph-xR2RML pour générer le graphe de connaissances à l'aide des règles de mapping xR2RML. Le script python `https://github.com/Wimmics/d2kab-wheat-kg` de pré-traitement génère 5 fichiers rassemblant les différents concepts provenant des 1631 CSV du jeu de

données : un fichier regroupant toutes les studies, un fichier regroupant toutes les observations, un fichier regroupant toutes les unités d'observation, un fichier regroupant tous les coordonnées GPS des sites et enfin un fichier regroupant le matériel biologique et matériel source. Ces fichiers CSV sont ensuite chargés dans une base mongoDB dans une collection nommée. Le listing 1.15 présente un exemple de règle de mapping qui exprime un patron de génération de triplets. Une seule source logique peut être interroger par TripleMap.

Notre jeu de données comporte 1631 fichiers CSV ce qui implique de relancer autant de fois le logiciel morph-xR2RML engendrant la génération de doublons de triplets RDF pour des ressources communes.

Les informations décrivant les *study*, *plant material*, données GPS ont un fichier CSV qui leur est propre.

Tous les URI des ressources respectent les patrons présentées dans le tableau 3. Les *observation* et *observation_ unit* ont déjà un identifiant qui leur est attribué lors du stockage en CSV des expérimentations; nous utilisons cet identifiant pour construire l'URI de ces dernières. Ces URI sont uniques et il ne peut y avoir deux *study* qui ont des *observation* et *observation_ unit* en commun. Pour contourner les problèmes d'encodage de caractères accentués et caractères spéciaux dans les URI, nous utilisons un hash sha1. Le hash propose une signature unique avec un risque de collision faible qui nous servira d'identifiant pour les ressources *study*, *material source*, *person* et *role*.

Study	http://www.urgi.versailles.inrae.fr/study/hash(name)
Investigation	http://www.urgi.versailles.inrae.fr/investigation/{name}
Biological Material	http://www.urgi.versailles.inrae.fr/biological_material/{lot_number}
Material Source	http://www.urgi.versailles.inrae.fr/material_source/hash(DOI)
Institution	http://www.urgi.versailles.inrae.fr/institution/{name}
Personne	http://www.urgi.versailles.inrae.fr/person/hash(name)
Role	http://www.urgi.versailles.inrae.fr/role/hash(name)
Observation level	http://www.urgi.versailles.inrae.fr/observation_level/{number}
Observation unit	http://www.urgi.versailles.inrae.fr/observation_unit/{id}
Observation	http://www.urgi.versailles.inrae.fr/observation/{id}
Factor	http://www.urgi.versailles.inrae.fr/factor/{name}
Factor Value	http://www.urgi.versailles.inrae.fr/factor_value/{name}

Table 3: Template des URI du graphe

La figure 16 présente le pipeline de transformation et génération des données du graphe de connaissances représentant les observations.

4.4.4 Résultats et statistiques

Le graphe généré comprend 4 144 418 triplets RDF pour un total de 327 Mo d'espace disque pris par les fichiers turtle. Le tableau 4 présente quelques statistiques sur le nombre d'entité du graphe.

Fig. 16: Pipeline de génération du graphe de connaissances des données d'observation

Nombre de triplet généré	4144418
Nb Study	816
Nb Investigation	1
Nb Observation unit	134309
Nb Observation	520038
Nb Biological Material	1840
Nb Material Source	1840
Nb Person	5
Nb Institution	1
Nb GPS location	11

Table 4: Statistique des ressources du second graphe généré

Les observations et les unités d'observation forment la majeure partie du jeu de données ce qui explique leur nombre important comparé aux autres ressources. Concernant les temps de pré-traitement des CSV et de génération du graphe, nous avons 26 secondes de pré-traitement et 5 minutes de génération du graphe. Les temps restent raisonnables pour une génération complète d'un graphe de connaissances de 337 Mo.

4.4.5 Validation et exploitation du graphe RDF

Plusieurs requêtes métier, définies par un expert exprimant des besoins de recherche avec différents critères, nous sont fournies par nos partenaire de l'INRAE. Ces requêtes exprime les attentes des scientifiques sur l'utilisation du graphe de connaissances et permettent de valider la formalisation que nous proposons de cette connaissance. Nous présentons dans ce qui suit chacune de ces requêtes et les résultats obtenus ou les raisons de l'échec de la requête.

– *Quelles sont les variables observées dont le trait est une maladie par année et par lieu, pour les accessions "Charger", "Apache" et "Tremie".*

La requête SPARQL correspondante à cette requête métier est présentée dans le listing 1.8. Le résultat attendu est l'ensemble des variables d'observation qui ont pour trait observé une maladie répertoriée dans l'ontologie CO_321 pour les graines "Charger", "Apache" et "Tremie". L'intention de cette requête est de relever l'impact du trait de maladie sur le blé suivant les conditions de pousse.

Listing 1.8: Requête 1 SPARQL

```
SELECT DISTINCT ?study ?date ?trait ?name ?label WHERE {
    ?study a ppeo:study; ppeo:hasEndDateTime ?date.
    ?obs a sosa:Observation;
            ppeo:hasObservedSubject ?observationUnit;
           sosa:hasFeatureOfInterest ?trait;
            sosa:observedProperty ?variable.
        ?trait a CO_321:BioticStressTrait; skos:prefLabel ?name.
    ?variable a CO_321:Variable; skos:prefLabel ?label.
    ?observationUnit a ppeo:observation_unit;
                       ppeo:hasBiologicalMaterial ?biologicalMaterial;
                       ppeo:partOf ?study.
        ?biologicalMaterial a ppeo:biological_material;
                           ppeo:hasAccessionName ?accession.
FILTER(?accession in ("CHARGER", "APACHE", "TREMIE")).
}
```

Un extrait du résultat de la requête du listing 1.8 est présenté dans le tableau 5. Le lieu n'est pas utilisé dans cette requête car il est encore absent du jeu de donnée.

Date	id Trait	Nom du trait	Nom de la Variable
2006-07-31	CO_321/0000919	Septoria tritici blotch incidence	ST-SCORE_score
2008-07-31	CO_321/0000937	Powdery mildew incidence	PM-SCORE_score
2008-07-31	CO_321/0000687	Leaf rust notes	rb
2008-07-31	CO_321/0000907	Stripe rust notes	YR-SCORE_score

Table 5: Extrait des Résultats de la Requête métier 1

– *Quelles sont les études pour lesquelles on observe des mesures pour les variables "frost" et "lodging".*

```
SELECT DISTINCT ?study ?variableName
WHERE {
    ?study a ppeo:study; ppeo:hasEndDateTime ?date.

    ?observationUnit a ppeo:observation_unit;
                       ppeo:hasBiologicalMaterial ?biologicalMaterial;
                       ppeo:partOf ?study.

        ?biologicalMaterial a ppeo:biological_material;
                           ppeo:hasAccessionName ?accession.

    ?observation a sosa:Observation;
                  sosa:observedProperty ?variable;
                      ppeo:hasObservedSubject ?observationUnit.
```

```
    ?variable a CO_321:Variable; skos:altLabel ?variableName.

        FILTER(regex(str(?variableName), "Frost") ||
            regex(str(?variableName), "lodging"))

}
```

La requête retourne un résultat de 314 Study dans laquelle l'une ou l'autre ou les deux variables d'observation a été mesurée. Un extrait du résultat de cette requête est présenté dans le tableau 6.

study	Nom de la variable
<study/24c8f9cb51fe23b1935567d5f465f629926dd4c8>	Frost susceptibility
<study/016fb74306fda62e34c0433c847a26fb975b80e7>	Susceptibility to lodging
<study/babfd59a1852720de0e78c678dc7d5467b27074c>	Susceptibility to lodging
<study/957aa7939a21b809306a2d54dd6e4a9c51bbcf9d>	Frost susceptibility
<study/e233f45a4f9b7d8b676341e26469a87fd71ac8fd>	Frost susceptibility

Table 6: Requête métier 2

– *Quelles sont les accessions qui sont testées plus de 5 années consécutives dans l'ensemble du dataset?*

L'intention de cette requête est d'évaluer la pertinence de l'utilisation d'une graine.

```
SELECT DISTINCT ?accession WHERE{
  ?bioMat ppeo:hasAccessionName ?accession.

  ?s1 a ppeo:study; ppeo:hasEndDateTime ?d1.
  ?s2 a ppeo:study; ppeo:hasEndDateTime ?d2.
  ?s3 a ppeo:study; ppeo:hasEndDateTime ?d3.
  ?s4 a ppeo:study; ppeo:hasEndDateTime ?d4.
  ?s5 a ppeo:study; ppeo:hasEndDateTime ?d5.

  ?obsUnit1 a ppeo:observation_unit;
    ppeo:partOf ?s1;
    ppeo:hasBiologicalMaterial ?bioMat.
  ?obsUnit2 a ppeo:observation_unit;
    ppeo:partOf ?s2;
    ppeo:hasBiologicalMaterial ?bioMat.
  ?obsUnit3 a ppeo:observation_unit;
    ppeo:partOf ?s3;
    ppeo:hasBiologicalMaterial ?bioMat.
  ?obsUnit4 a ppeo:observation_unit;
```

```
    ppeo:partOf ?s4;
    ppeo:hasBiologicalMaterial ?bioMat.
  ?obsUnit5 a ppeo:observation_unit;
    ppeo:partOf ?s5;
    ppeo:hasBiologicalMaterial ?bioMat.

    filter(year(?d1) - year(?d2) = 1)
    filter(year(?d2) - year(?d3) = 1)
    filter(year(?d3) - year(?d4) = 1)
    filter(year(?d4) - year(?d5) = 1)
}
```

– *Quelles sont les accessions qui font l'objet d'un changement d'investigation d'une study à l'autre (eg. de INRA Wheat Network not BRC accession (B and C series) à INRA Small Grain Cereals Network)*

L'intention de cette requête est de récupérer les graines qui sont passées du réseau de graine expérimental au réseau de graine agréé. La vérification de cette requête n'est pas encore possible car nous ne disposons que d'une investigation dans le dataset, donc uniquement un réseau de graine. Le listing 1.9 propose une formalisation sparql de cette requête.

Listing 1.9: Requête métier 3

```
SELECT ?accession WHERE {
  ?study1 a ppeo:study; ppeo:partOf ?investigation1.
  ?investigation1 a ppeo:investigation; ppeo:hasName ?network1.
  ?investigation2 a ppeo:investigation; ppeo:hasName ?network2.
  ?study2 a ppeo:study; ppeo:partOf ?investigation2.
  ?obsUnit1 a ppeo:observation_unit; ppeo:hasBiologicalMaterial ?bioMat;
      ppeo:partOf ?study1.
  ?obsUnit2 a ppeo:observation_unit; ppeo:hasBiologicalMaterial ?bioMat;
      ppeo:partOf ?study2.
  ?bioMat a ppeo:biological_material; ppeo:hasAccessionName ?accession.
  filter(?network1 = "INRA Small Grain Cereals Network")
  filter(?network2 = "INRA Wheat Network not BRC accession (B and C
      series)")
}
```

– *Quelles sont les valeurs observées des traits de qualité (eg. dureté du grain CO_321:0000072 et qualité boulangère CO_321:0500001) sur l'ensemble du dataset, avec les informations d'accessions, lieux, années, traitements, etc. afin de pouvoir étudier le lien entre ces traits.*

```
select (year(?date) as ?year) ?accession ?factor ?trait ?value where {
```

```
  ?obs a sosa:Observation; sosa:hasFeatureOfInterest ?trait;
      ppeo:hasObservedSubject ?obsUnit; sosa:hasSimpleResult ?value.
  ?obsUnit a ppeo:observation_unit; ppeo:hasBiologicalMaterial ?bioMat;
      ppeo:partOf ?study; ppeo:hasFactorValue ?factor.
  ?bioMat a ppeo:biological_material; ppeo:hasAccessionName ?accession.
  ?study a ppeo:study; ppeo:hasEndDateTime ?date.
  filter(?trait in (<http://www.cropontology.org/rdf/CO_321/0000072>,
      <http://www.cropontology.org/rdf/CO_321/0500001>))
}ORDER BY ?date
```

4.5 Alignement entre la Wheat Trait Ontology et la Crop Ontology 321

Suite à la génération des graphes de connaissances d'annotations textuelles provenant d'articles scientifiques et de données d'observation de champs de blé, cette partie s'intéresse à l'alignement des deux ontologies Wheat Trait Ontology (WTO) et la Crop Ontology 321 qui ont été présentées précédemment dans ce travail de recherche. Développée chacune par une équipes de recherche, elles ont été utilisés différemment. WTO est utilisée pour annoter les phénotypes et traits dans la littérature scientifique alors que la CO_321 est utilisée pour décrire les traits observés lors des expérimentations sur des champs de culture de blé. L'alignement des deux ontologies est une étape fondamentale pour l'intégration des deux graphes générés dans ce travail.

4.5.1 Problématique

Les scientifiques souhaitent déduire de nouvelles connaissances en agronomie en interrogeant à la fois des données agronomiques provenant de la littérature scientifique et des données agronomiques provenant des observations réalisées lors des expérimentations, d'où tout l'intérêt d'aligner les ontologies WTO et CO_321.

Il s'agit de détecter les entités similaires/équivalentes dans les deux ontologies. Cet alignement à été fait à la main par les experts du domaine ayant participé au développement des deux ontologies. Ainsi, ils ont revu à la main chaque hiérarchie d'ontologie pour leur attribuer une relation de correspondance du concept qui s'en approche. Au moins 3 types d'alignement ont été définis par les experts entre des classes deux ontologies dans leurs versions actuelles:

- Un matching exact qui correspond au fait que les deux entités sont équivalentes. Par exemple, le trait "*aluminium resistance*" de la WTO possède une correspondance dans la CO_321 qui est l'entité "*Aluminum tolerance*". Cet exemple met en évidence les nommages différents que peuvent adopter les experts et, par conséquent, la difficulté d'automatisation de cette correspondance.
- Un matching de subsumption entre une classe CO_321 et une classe WTO. Dans ce cas, une relation de mapping asymétrique de type subsomption est

définie entre une classe CO_321 et une classe WTO pour indiquer que la premiere est plus générale que la seconde. A titre d'exemple, la classe de trait "*co_321:Grain weight*" de l'ontologie CO_321 est plus générale que la classe de trait "*WTO:Thousand kernel weight*".
– Un matching de subsumption pour indiquer une relation asymétrique dans le cas ou La classe WTO est plus générale que la classe CO_321. A titre d'exemple, la classe WTO "*WTO:glutenin content*" est plus générale que la classe *CO_321:Glutenin:Glu-D3 composition*.

4.5.2 Choix technique

La définition de la CO_321 et WTO étant en skos, nous bénéficions des relation de matching définie par le langage et les utilisons. Nous avons fait le choix de nous concentrer sur les relations simples de proximité des traits et phénotypes. Cette proximité se traduit par l'utilisation des relations: (1) skos:exactMatch dans le cas où le phénotype et le trait sont les-mêmes, (2) skos:broadMatch dans le cas d'une généralisation de l'un par rapport à l'autre, (3) skos:narrowMatch dans le cas d'une spécialisation de l'un part rapport à l'autre et enfin (4) skos:closeMatch pour les relations plus ambiguë.

Toutes les informations concernant l'alignement entre les deux ontologies sont répertoriées dans un fichier Excel et servira à la génération automatique du fichier turtle de l'alignement.

4.5.3 Résultat

Un fichier turtle contenant l'alignement entre les phénotypes de la WTO et les traits de la CO_321 recense 288 relation exactMatch, 31 relation broadMatch et narrowMatch, et 88 relation closeMatch pour un total de 443 triplets générés. Ces chiffres s'expliquent par la symétrie des relations pour le cas des exactMatch et closeMatch et la relation inverse des broadMatch et narrowMatch. La requête du listing 1.10 nous permet de vérifier que l'alignement fonctionne entre les connaissances provenant de la WTO et celle provenant de la CO_321.

Listing 1.10: Requête test de l'alignement

```
select ?obs ?doc where {
    ?annotation a oa:Annotation;
        oa:hasBody ?phenotype;
        oa:hasTarget ?target.
    ?target oa:hasSource ?partDoc.
    ?partDoc frbr:partOf+ ?doc.
    ?doc a fabio:ResearchPaper.

    ?phenotype skos:exactMatch ?trait.

    ?obs sosa:hasFeatureOfInterest ?trait.
}LIMIT 100
```

Observation	Article PubMed
<http://www.urgi...inrae.fr/observation/3707477>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3356213>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3620210>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3707480>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3315563>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3315561>	<https://pubmed.ncbi...nih.gov/32719416>
<http://www.urgi...inrae.fr/observation/3663762>	<https://pubmed.ncbi...nih.gov/32719416>

Table 7: Résultat de la requête du listing 1.10

Le tableau 7 présente un échantillon du résultat de la requête.

Dans le cas ou on exploite un alignement de type subsumption, il est évident que les résultats de requêtes vont inclure les données relatives au concept lui-même et tous les concepts qui subsument dans l'autre ontologie.

5 Conclusions

Globalement, l'objectif de lifting de données a été atteint avec la construction du graphe de connaissances Wheat-KG qui montre bien l'adoption les technologies du web sémantique au sein de la recherche en agronomie. J'ai beaucoup appris durant ce stage notamment comment se déroule une activité de recherche sur un projet réunissant plusieurs laboratoire. J'ai pu défendre mon travail face à des experts du domaines lors de la plénière D2KAB et ainsi progresser dans la présentation de mon travail. J'ai progressé énormément dans la création et correction de graphes de connaissances, notamment face aux différents problèmes d'inconsistance de données que l'on retrouve fréquemment dans les données du monde réel. Les divers processus de nettoyage des données sans les altérer m'ont permis de m'améliorer dans ce domaine. J'ai aussi eu l'immense honneur de faire partie des auteurs d'un article publié dans la conférence PFIA-2022 qui a valorisé le travail que j'ai effectué lors de mon TER et montre l'intérêt des scientifiques à disposer de graphe de connaissances pour leurs recherches.

5.1 Perspectives

Afin de compléter le travail effectué, la mise en place d'un SPARQL endpoint est prévue comme la perspective a court terme afin de pouvoir interroger le dataset produit et relever d'éventuelle manque. Actuellement, pour pouvoir interroger avec les dataset, il est nécessaire de récupérer localement le dump du graphe de connaissances Wheat-KG.

Comme autre perspective, la mise en place d'une interface utilisateur, comme celle proposée par le site de l'URGI, permettrait l'adoption plus générale de l'approche des construction de graphe de connaissances. Un système permettant de traduire des requêtes exprimer en langage naturel vers le langage SPARQL faciliterait l'utilisation du graphe de connaissances.

References

RDF. Wikipedia. *Resource Description Framework* https://fr.wikipedia.org/wiki/Resource_Description_Framework.

R2RML. Souripriya Das, Seema Sundara, Richard Cyganiak. *R2RML: RDB to RDF Mapping Language* https://www.w3.org/TR/r2rml/.

Xr2rml. Franck Michel, Loïc Djimenou, Catherine Faron-Zucker et Johan Montagnat. *xR2RML: Relational and Non-Relational Databases to RDF Mapping Language* consultable en ligne sur https://www.i3s.unice.fr/~fmichel/xr2rml_specification_v5.html.

DCT. DCMI Usage Board. *Dublin Core Metadata Terms* https://www.dublincore.org/specifications/dublin-core/dcmi-terms/

SKOS. W3C semantic web. *SKOS Simple Knowledge Organization System* https://www.w3.org/2004/02/skos/

NCBI. National Center for Biotechnology Informations. *NCBI Taxonomy* https://www.ncbi.nlm.nih.gov/taxonomy

AlvisNLP. Ba Mouhamadou, Bossy Robert, Nédellec Claire. *Customized automatic corpus annotations using AlvisNLP/ML [2017]* https://github.com/Bibliome/alvisnlp

SOSA. Sensor, Observation, Sample and Actuator, W3C https://www.w3.org/TR/vocab-ssn/

PPEO. Plant and Phenotype Experiment Ontologyhttps://github.com/MIAPPE/MIAPPE-ontology

CO321. Carlos Guzmán, Hector González, Enrique Autrique, Javier Pena, Pawan Singh, Matthew Reynolds, Tom Payne, Velu Govindan. Wheat Ontology - CO_321 http://agroportal.lirmm.fr/ontologies/CO_321/?p=summary

SPµSrv. Franck Michel, "Retrieve metadata from PubMed about an article given its Pubmed ID" accessible depuis https://sparql-micro-services.org/service/pubmed/getArticleByPMId_sd/

SPWrapper. dayures, https://sparqlwrapper.readthedocs.io/en/latest/index.html

URGI. https://urgi.versailles.inra.fr/ephesis/ephesis/viewer.do#dataResults/trialSetIds=8

GnpIS. Cyril Pommier, Célia Michotey, Guillaume Cornut, Pierre Roumet, Eric Duchêne, Raphaël Flores, Aristide Lebreton, Michael Alaux, Sophie Durand - Applying FAIR principles to plant phenotypic data management in GnpIS. Plant Phenomics, Science Partner Journals, 2019 - https://hal.inrae.fr/hal-02624031/document
Erik Kimmel, et al.

6 Annexe

Listing 1.11: Exemple de retour du sparql micro-service

```
@prefix schema: <http://schema.org/> .
@prefix ma: <http://www.w3.org/ns/ma-ont#> .
@prefix fabio: <http://purl.org/spar/fabio/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix bibo: <http://purl.org/ontology/bibo/> .
@prefix dc: <http://purl.org/dc/terms/> .


<http://doi.org/10.1094/MPMI-22-11-1366> dc:creator "Jia H" ;
    dc:creator "Cho S" ;
    dc:creator "Muehlbauer GJ" ;
    dc:issued "2009 Nov" ;
    dc:language "eng" ;
    dc:source "Molecular plant-microbe interactions : MPMI" ;
    dc:title "Transcriptome analysis of a wheat near-isogenic line pair
        carrying Fusarium head blight-resistant and -susceptible
        alleles." ;
    bibo:doi "10.1094/MPMI-22-11-1366" ;
    bibo:issn "0894-0282" ;
    bibo:issue "11" ;
    bibo:numPages "1366-78" ;
    bibo:pmid "19810806" ;
    bibo:volume "22" ;
    fabio:hasPubMedId "19810806" ;
    fabio:journal "Molecular plant-microbe interactions : MPMI" ;
    schema:authorList _:b2559315 ;
    schema:url <https://pubmed.ncbi.nlm.nih.gov/19810806> ;
    rdf:type schema:ScholarlyArticle ;
    rdf:type bibo:AcademicArticle ;
    rdf:type fabio:ResearchPaper .
```

Listing 1.12: Exemple de variable CO_321

```
    <http://www.cropontology.org/rdf/co_321/1000251>
      a      skos:Concept, CO_321:Variable ;
      skos:altLabel "Susceptibility to cercosporella (plantlet stage)" ;
      skos:prefLabel "pvp" ;
      CO_321:created "21/09/2017"^^xsd:date ;
      CO_321:hasMethod <http://www.cropontology.org/rdf/co_321/1021003> ;
      CO_321:hasScale <http://www.cropontology.org/rdf/co_321/1030102> ;
      CO_321:hasTrait <http://www.cropontology.org/rdf/co_321/1010139> ;
      CO_321:hasXReference "WIPO:0000251" .
```

Listing 1.13: Exemple de Trait CO_321

```
<http://www.cropontology.org/rdf/co_321/0000029>
  a       skos:Concept , CO_321:MorphologicalTrait ;
  rdfs:comment "Length of coleoptile (a protective sheath that
      surrounds the shoot tip and the embryonic leaves of the young
      shoot of grasses)." ;
  skos:altLabel "ColeopLng" ;
  skos:prefLabel "Coleoptile length" ;
  CO_321:entity "Coleoptile" ;
  CO_321:hasAttribute "Length" .
```

Listing 1.14: Exemple de Methode CO_321

```
<http://www.cropontology.org/rdf/CO_321/0000364>
  a       skos:Concept , CO_321:MeasurementMethod ;
  rdfs:comment "Standard method for spectral reflectance." ;
  CO_321:reference "Pask et al., 2012 (Ch. 7), Reynolds et al., 2001
      (Ch. 5)" ;
  skos:prefLabel "Canopy spectral reflectance Measurement" .
```

Listing 1.15: Exemple de règle de mapping pour générer les unités d'observation

```
<#Source>
    a rr:TripleMap;
    xrr:logicalSource [xrr:query """db.Observations.find()""";];
    rr:subjectMap [
        rr:template
            "http://www.urgi.versailles.inrae.fr/observation/{$.observation_id}";
        rr:class sosa:Observation, ppeo:observation;
    ];
    rr:predicateObjectMap [
        rr:predicate ppeo:hasObservedSubject;
        rr:objectMap [
            rr:template
                "http://www.urgi.versailles.inrae.fr/observation_unit/{$.observation_unit}";
        ];
    ];
    rr:predicateObjectMap [
        rr:predicate sosa:hasFeatureOfInterest;
        rr:objectMap [xrr:reference "$.trait"; rr:termType rr:IRI;];
    ];
    rr:predicateObjectMap [
        rr:predicate sosa:observedProperty;
        rr:objectMap [xrr:reference "$.variable"; rr:termType rr:IRI;];
    ];
    rr:predicateObjectMap [
        rr:predicate sosa:usedProcedure;
        rr:objectMap [xrr:reference "$.method"; rr:termType rr:IRI;];
    ];
```

```
    rr:predicateObjectMap [
        rr:predicate sosa:hasSimpleResult;
        rr:objectMap [xrr:reference "$.value";];
    ].
```

Abstract. Dans ce rapport, vous trouverez mon implication au projet D2KAB sur la transformation de données agronomique durant mon stage. Le stage se déroule dans un laboratoire de l'I3S au sein de l'équipe WIMMICS sous la supervision de Catherine Faron-Zucker, le maître de stage. Le sujet du stage est l'intégration de données hétérogènes dans le domaine de l'agronomie. Le but est de générer des graphes RDF de cette connaissance pour donner un accès uniforme à ces données, lié des données provenant d'article scientifique et de campagne d'observation, et enfin déduire de nouvelles connaissances. Toutes ces informations concernent la culture du blé.

Abstract. In this report, you will find my contribution in the d2kab project about data lifting of agronomical data during my internship. It takes place in the WIMMICS team, at the I3S laboratory, under the lead of Catherine Faron-Zucker, the internship supervisor. The subject of the internship is data integration of heterogeneous data from agronomical domain. The goal is to generate a RDF graph of that knowledge to be able to give a uniform acces to those data, to link data from scientific paper and observation campaign, and infer new knowledge. All the information is about wheat culture.