

Représentation des connaissances et des raisonnements en agronomie systémique pour l'innovation en agroécologie

Elie NAJM

Équipe GraphIK - INRIA & LIRMM
en collaboration avec l'UMR ABSys

4/11/2021

Introduction

Objectif général : Concevoir un outil d'**aide à la conception de systèmes agroécologiques** qui s'appuie sur la **représentation** des **connaissances** en jeu et des raisonnements sur ces connaissances.

Un outil qui :

- **Propose** à l'utilisateur (*agronome*) **des espèces** qui rendent potentiellement un ou plusieurs services écosystémiques voulus.
- **Justifie le choix** des espèces proposées en rendant visibles les étapes du raisonnement.

Introduction

Cas d'étude : Enherbement d'une parcelle de vigne.



Démarche adoptée :

- Approche **fonctionnelle** par différenciation d'une approche taxonomique
- Calculer des valeurs de services écosystémiques pour les espèces.

Architecture simplifiée de l'outil

Représentation en **logique** (du premier ordre)
et raisonnements basés sur l'inférence logique

Requêtes posées
par l'utilisateur.
*ex : Quelles sont les k
meilleures espèces pour
un service donné ?*

**Connaissances
expertes**

Base de connaissances
Ontologie
Faits

Niveau : connaissances

Niveau : données

BDD 1

BDD 2

BDD 3

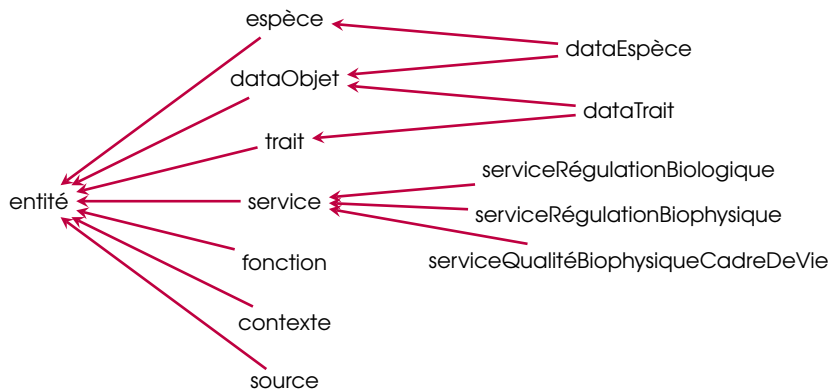
mapping

mapping

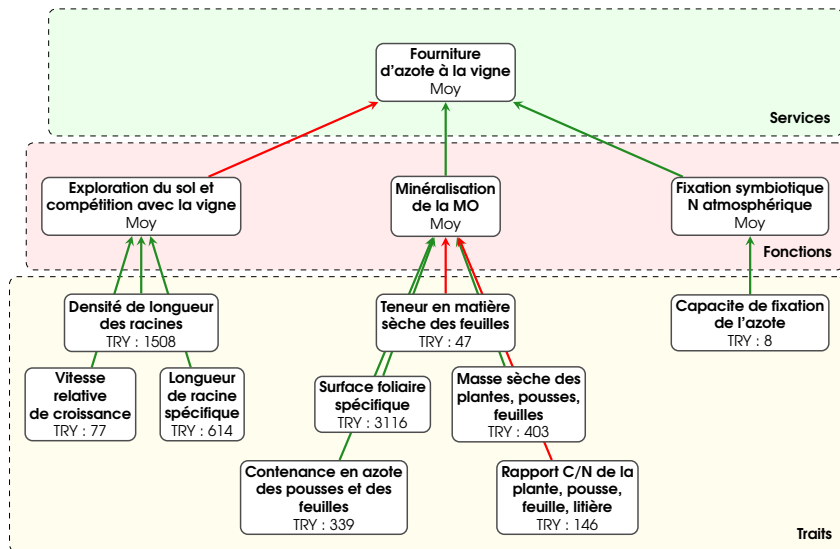
mapping

Construction d'une ontologie

- Définit un vocabulaire en termes de **concepts** (types d'objets) et de **relations** entre des objets
- Décrit des connaissances en utilisant ce vocabulaire sous forme de **faits** et de **règles**



Conception de schémas traits-fonctions-services



Des schémas aux faits (1/2)

- Association des traits de schémas à ceux de la base de données

```
aPourTraitID("Longueur de racine spécifique", 614, try).  
aPourTraitID("Vitesse relative de croissance", 77, try).
```

- Traduction des liens traits-fonctions-services

```
serviceQualiteBiophysiqueCadreDeVie("Fourniture d'azote à la vigne").  
fonction("Exploration du sol et compétition avec la vigne").  
trait("Longueur de racine spécifique").
```

```
estLieA3fonctions("Fourniture d'azote à la vigne",  
"Fixation symbiotique N atmosphérique",1,  
"Minéralisation de la MO",1,  
"Exploration du sol et compétition avec la vigne",-1,avg).
```

Des schémas aux faits (2/2)

```
estLieA3fonctions("Fourniture d'azote à la vigne",  
"Fixation symbiotique N atmosphérique",1,  
"Minéralisation de la MO",1,  
"Exploration du sol et compétition avec la vigne",-1,avg).
```

Le dernier paramètre désigne la **méthode d'agrégation** choisie pour passer des fonctions aux services (ici la moyenne).

Les chiffres désignent :

- Une **pondération** : ici les 3 services ont le même poids
- **Nature du lien** : chiffre > 0 → lien positif
chiffre < 0 → lien négatif

La "Fourniture d'azote à la vigne" est liée positivement à la "Fixation symbiotique N atmosphérique" et la "Minéralisation de la MO" et négativement à l' "Exploration du sol et compétition avec la vigne" avec la même importance (1) et la valeur de service est calculée en utilisant la moyenne.

TRY et FRED

Bases de données actuelles : **TRY** et **FRED**.

- TRY : **base de données mondiale de traits végétaux**
 - . Plus de 400 ensembles de données intégrés.
 - . 11 millions d'enregistrements pour 2100 traits de 4 millions de plantes individuelles, représentant 160 000 taxons végétaux (principalement des espèces).
- FRED : **base de données sur les traits racinaires**
 - . Données recueillies auprès de plus de 1400 sources de données.
 - . Plus de 150 000 observations de plus de 330 traits racinaires.

Extrait de TRY

TraitID	TraitName	AccSpeciesID	AccSpeciesName	OrigName	OrigValueStr
3106	plant Height Vegetative	48206	sambucus Nigra	maximum Height	NULL
3106	plant Height Vegetative	53949	teucrium Chamaedrys	maximum Height]
3106	plant Height Vegetative	51363	spiraea Alba	height: 3. Typical Maximum (Cm)	simple
3106	plant Height Vegetative	7364	betula Utilis	maximum Height (Cm)	unknown
3106	plant Height Vegetative	14694	corylus Heterophylla	maximum Height (Cm)	unknown
3106	plant Height Vegetative	268329	ulmus Davidiana Var. Japonica	maximum Height (Cm)	unknown
3106	plant Height Vegetative	42187	pinus Tabuliformis	maximum Height (Cm)	unknown
3106	plant Height Vegetative	45434	quercus Mongolica	maximum Height (Cm)	unknown
3106	plant Height Vegetative	42137	pinus Koraiensis	maximum Height (Cm)	unknown
3106	plant Height Vegetative	56305	viburnum Burejaeticum	maximum Height (Cm)	unknown
3106	plant Height Vegetative	79395	conospermum Crassinervium	height(m)	<1 Few Flowers To 1.5
3106	plant Height Vegetative	73686	adenanthos Sericeus	height(m)	<3
3106	plant Height Vegetative	32029	krascheninnikovia Ceratoides	plant Height	?
3106	plant Height Vegetative	44633	psoralea Bituminosa	plant Height	80c
3106	plant Height Vegetative	235402	jacobaea Vulgaris	canopy Height (Cm)	na
3106	plant Height Vegetative	201564	agathosma Abrupta	plant Height (M)	a
3106	plant Height Vegetative	5527	asparagus Aethiopicus	plant Height (M)	a
3106	plant Height Vegetative	219170	cynanchum Obtusifolium	plant Height (M)	a
3106	plant Height Vegetative	103019	fabaceae	plant Height (M)	a
3106	plant Height Vegetative	46287	rhoicissus Tomentosa	plant Height (M)	a
3106	plant Height Vegetative	43714	potentilla Pusilla	vegth [Cm]	na
3106	plant Height Vegetative	41907	picea Abies	h	.
3106	plant Height Vegetative	14789	cotinus Coggygria	canopyheight	>2
3106	plant Height Vegetative	14789	cotinus Coggygria	canopyheight	>2
3106	plant Height Vegetative	14789	cotinus Coggygria	canopyheight	>2

- Valeur de trait pour une espèce = moyenne des valeurs des observations pour ce trait et cette espèce.
- Normalisation des valeurs de traits (valeurs normalisées comprises entre 0 et 1) pour pouvoir les agréger.

Faits extraits des bases

Difficultés rencontrées :

- **Valeurs non standardisées** : types de valeurs différentes pour un même trait, différentes unités, ...
⇒ nettoyage des données.
- **Conditions de culture différentes**
⇒ séparation des valeurs en trois types de conditions de culture : naturelles(n), expérimentales(e) et totales(t), normalisation des valeurs.
- **Valeurs manquantes**
⇒ notion de fiabilité d'une valeur de fonction/service calculée
- **Variations des valeurs** : les valeurs dépendent
⇒ du type de mesure "ValueKindName"
⇒ des sous-traites "DataName"
⇒ d'éléments de contexte (par ex : le % de phosphore dans le sol)

Faits construits à partir des bases de données :

```
aPourEspeceID("Cichorium intybus", "12917", try).  
aPourValeurTraitBase("614", "12917", 0.9, try, "t").  
aPourValeurTraitBase("77", "12917", 0.5, try, "t").
```

Règles génériques

Intérêt : Les règles sont **indépendantes des schémas** et des méthodes d'agrégations choisies.

Différents types de règles :

- Règles traduisant l'ontologie.

```
serviceQualiteBiophysiqueCadreDeVie(Service) -> service(Service)
```

- Règles permettant de passer des traits aux fonctions, puis au services.

```
estLieA3Traits(Fct,Trait1,Lien1,Trait2,Lien2,Trait3,Lien3,Agreg);  
aPourValeurTrait(Trait1,Esp1,Val1);aPourValeurTrait(Trait2,Esp1,Val2);  
not aPourValeurTrait(Trait3,Esp1,_);  
-> aPourValeurFonctionCalculee(Fct,Esp1,Agreg(Val1,Lien1,Val2,Lien2),66).
```

Calcul des valeurs de fonctions

Faits extraits de TRY

```
aPourEspeceID("Cichorium intybus", "12917", try).  
aPourValeurTrait("614", "12917", 0.9, try, "t").  
aPourValeurTrait("77", "12917", 0.5, try, "t").
```

Faits construits à partir des schémas

```
aPourTraitID("Longueur de racine spécifique", 614, try).  
aPourTraitID("Vitesse relative de croissance", 77, try).  
estLieA3Traits("Exploration du sol et compétition avec la vigne",  
"Longueur de racine spécifique", 1, "Densité de longueur racinaire", 1,  
"Vitesse relative de croissance", 1, avg).
```

Règle à appliquer

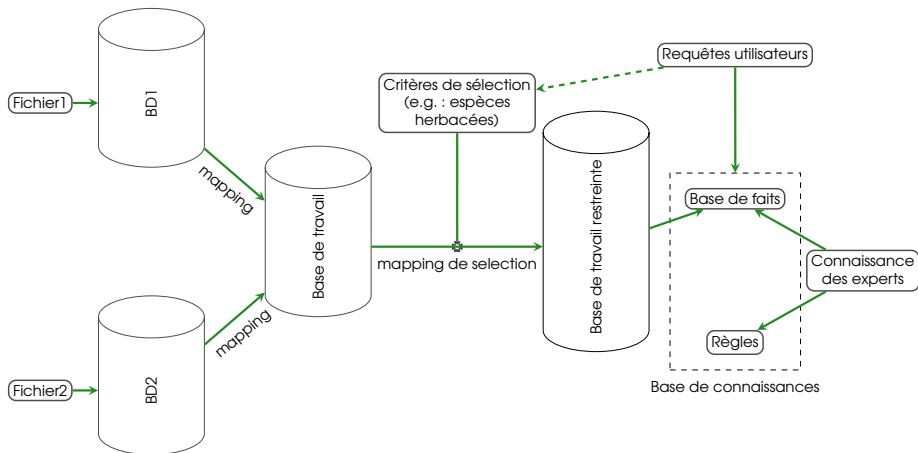
```
estLieA3Traits(Fct, Trait1, Lien1, Trait2, Lien2, Trait3, Lien3, Agreg);  
aPourValeurTrait(Trait1, Esp1, Val1); aPourValeurTrait(Trait2, Esp1, Val2);  
not aPourValeurTrait(Trait3, Esp1, _);  
-> aPourValeurFonctionCalculee(Fct, Esp1, Agreg(Val1, Lien1, Val2, Lien2), 66).
```

Nouveau fait construit

```
aPourValeurFonctionCalculee("Exploration du sol et  
compétition avec la vigne", "Cichorium intybus", 0.7, 66).
```

L'exploration du sol et compétition avec la vigne a une valeur de **0.7** pour l'espèce **Cichorium intybus** avec une fiabilité de **66%** (ou 2/3)

Architecture détaillée



Mappings

Mapping : de façon générale, un mapping d'une structure S_1 vers une structure S_2 est de la forme : $q_{S_1}(x_1, \dots, x_k) \rightarrow q_{S_2}(f_1(x_1), \dots, f_k(x_k))$

x_j : **variable réponse** prenant une valeur lors de l'évaluation de q_{S_1} sur S_1

f_j : fonction qui **transforme cette valeur en une autre**

q_{S_1} : **requête d'interrogation** sur S_1

q_{S_2} : **requête d'insertion** dans S_2

Une structure peut être une base de données relationnelles, un fichier (csv), une base de faits, ...

Nous définissons différents types de mappings :

- Mappings de nettoyage
- Mappings de base de données
- Mappings de sélection
- Mappings data2fact

Mappings de nettoyage



Forme des mappings : $q_{CSV} \rightarrow q_{BD}$

- **Création du schéma BD**

Exemple : TRY(TraitID VARCHAR(20), EspeceID VARCHAR(20), EspeceName VARCHAR(200), Valeur DOUBLE, TypeDeCondition ENUM('n','e'))

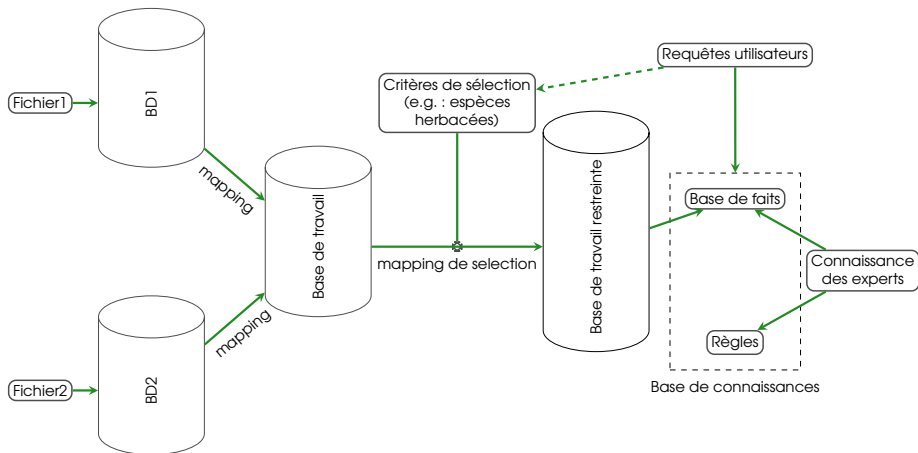
- **Mapping de nettoyage**

$q_1(x_1, \dots, x_k) = \text{line}(X_{i_1}, \dots, X_{i_k}); \text{valide}(X_{i_1}, \dots, X_{i_k})$

$q_2(x_1, \dots, x_k) = \text{INSERT}(f_1(X_1, \dots, X_k), \dots, f_{k'}(X_1, \dots, X_k))$

- . f_i = filtres sur les valeurs
- . $\text{valide}(X_{i_1}, \dots, X_{i_k})$ = conditions à satisfaire pour autoriser l'insertion

Architecture détaillée



Mappings de bases de données



Forme des mappings : $q_{BD} \rightarrow q_{BD}$

S_1 et S_2 sont deux bases de données relationnelles, donc q_{S_1} est une requête SQL de la forme SELECT et q_{S_2} une requête d'insertion de la forme INSERT

Mapping de base de données :

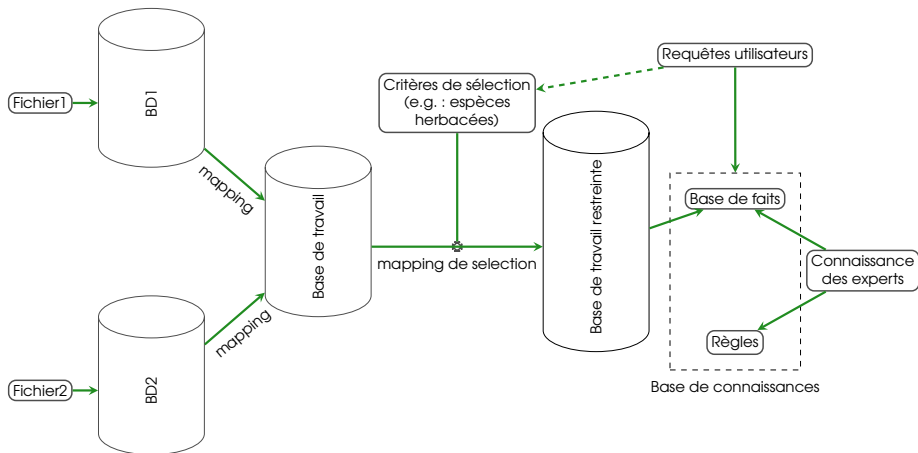
$q_1(x_1, \dots, x_k) = \text{SELECT } x_1, \dots, x_k \text{ FROM (table) WHERE (conditions) GROUP BY}$

$q_2(x_1, \dots, x_k) = \text{INSERT INTO (table) VALUES ('x'_1, \dots, 'x'_k)}$

Intérêt : faire des manipulations qui concernent plusieurs lignes de la même table à la fois.

Ex : agréger des valeurs de plusieurs lignes.

Architecture détaillée



Mapping de sélection

Forme des mappings : $q_{BD} \rightarrow q_{BF}$

Intérêt : Prendre en compte les critères de sélection avant le calcul des valeurs normalisées. Ces critères de sélection sont donnés par l'utilisateur en utilisant le vocabulaire ontologique. Ex : se restreindre aux espèces herbacées (requête q_{BF})

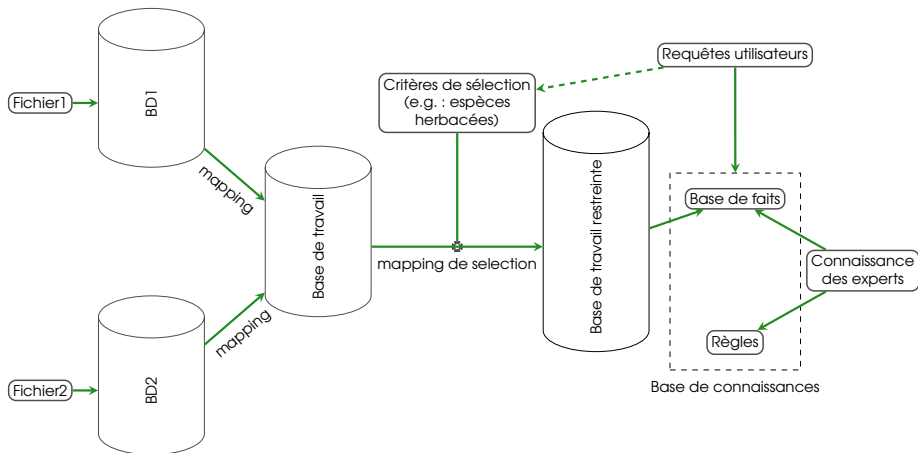
q_{BD} = SELECT AccSpeciesID FROM BD WHERE woodiness = 'H'

q_{BF} = FormeDeCroissance(X, "herbacee")

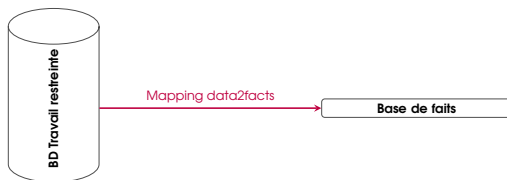
Idée :

- Associer un critère de sélection (requête sur le vocabulaire de l'ontologie) à une requête en base de données.
- Utiliser ce type de mapping "à l'envers" (en réécriture) plutôt que de les déclencher
- Matérialiser la requête sous forme de vues (**vueHerbacee**) pour restreindre les tuples d'intérêt dans BD Travail.

Architecture détaillée



Mappings data2facts



Forme des mappings : $q_{BD} \rightarrow q_{BF}$

Les mappings data2facts opèrent sur BD Travail restreinte et peuvent faire appel aux vues créées par les mappings de sélection (vueHerbacee), ce qui permet de limiter les entités d'intérêt et de faire des requêtes d'agrégation limitées à ces entités d'intérêt.

ex : calcul de la valeur normalisée d'un trait pour une espèce

Mapping de data2facts :

q_{BD} = SELECT TraitID, AccSpeciesID, ValeurNormalisee FROM Try INNER JOIN vue ON
Try.AccSpeciesID = vue.AccSpeciesID

q_{BF} = aPourValeur(TraitID, AccSpeciesID, ValeurNormalisee, Try)

Exemple

Exemple de requête : Quelles sont les 10 meilleures espèces pour le service de fourniture d'azote à la vigne avec une fiabilité d'au moins 50% ?

Traduction en une requête de base de données : `SELECT * FROM aPourValeurService WHERE Service = "fourniture azote a la vigne" AND Fiabilité > 50 ORDER BY aPourValeurService.Valeur DESC LIMIT 10`

Résultat

Espece	Valeur ▾ 1	Fiabilité
lotus corniculatus	0.77680105	89
trifolium repens	0.76497017	78
trifolium pratense	0.75703032	78
trifolium hybridum	0.72317251	55
trifolium dubium	0.72164679	67
lathyrus pratensis	0.71902614	55
anthyllis vulneraria	0.71794294	67
trifolium campestre	0.70497417	55
medicago lupulina	0.70018017	67
onobrychis viciifolia	0.69490763	55

Etat actuel : toute la chaîne fonctionne mais en mettant bout à bout différents outils
⇒ **Tâche 1** : implémenter cette architecture sur un seul outil (Graal)

Validation de l'approche (1/3)

Méthodologie pour la validation des résultats :

- **Sélection de 12 espèces** à la fois bien renseignées dans TRY et supposées connues par les experts
- **Évaluation par l'outil** de ces espèces pour un certain nombre de fonctions et services
- **Classement** en 3 catégories (* = mauvais, ** = moyen, **** = bien + réponse "ne sais pas") des espèces **par les experts**
- **Comparaison** entre le classement de l'outil et celui des experts

Problèmes

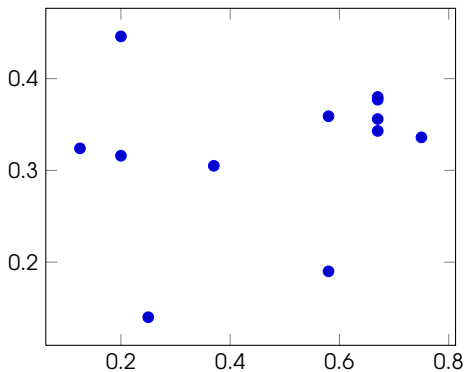
- Peu de réponses des experts : 8 réponses au questionnaire pour 25 experts sollicités
- Les réponses des experts ne s'accordent pas entre elles sur l'évaluation des espèces.

Validation des résultats (2/3)

Classements pour la fonction de **minéralisation de la MO**

Espèce	*	**	***	NSP	Moy	Valeur	Fiabilité
Luzerne lupuline	0	3	3	2	0.75	0.336	100
Trèfle douteux	0	4	2	2	0.67	0.343	100
Trèfle fraise	0	4	2	2	0.67	0.377	80
Trèfle violet	0	4	2	2	0.67	0.356	100
Trèfle rampant	0	4	2	2	0.67	0.38	100
Lotier corniculé	0	5	1	2	0.58	0.359	100
Luzerne cultivée	0	5	1	2	0.58	0.19	60
Plantain lancéolé	2	1	1	4	0.37	0.305	100
Navet potager	2	2	0	4	0.25	0.14	40
Féтуque élevée	3	2	0	3	0.2	0.446	60
Féтуque rouge	3	2	0	3	0.2	0.316	100
Dactyle pelotonné	3	1	0	4	0.125	0.3247	100

Validation des résultats (3/3)



⇒ **Tâche 2** : terminer la validation

Retours des utilisateurs (1/2)

Prise en compte des retours des utilisateurs pour :

- **Compléter** les connaissances : pour pallier le problème des valeurs manquantes
- **Contredire** les faits produits par le mécanisme d'inférence

On suppose que l'utilisateur peut agir sur les 3 niveaux : traits, fonctions et services.

Utilisateur = source de données (qui utilise le vocabulaire de l'ontologie).

Notion de préférence : on définit au départ une préférence par défaut entre les bases

ex : *TRY > Util1 > FRED > Util2*

```
basePrioritaire(try, util1) .
```

```
basePrioritaire(util1, fred) .
```

```
basePrioritaire(fred, util2) .
```

L'utilisateur peut affiner les préférences par défaut.

ex : *Préférences Util1 : Util1 > TRY > Util2 > FRED pour le trait "longueur de racine spécifique". La valeur de trait prise en compte sera celle de la base prioritaire (Util1) si elle en a une, sinon la deuxième (TRY), ...*

⇒ **Tâche 3** : Développer cette notion de prise en compte des retours et préférences utilisateurs

Retours des utilisateurs (2/2)

À l'aide des règles suivantes, la valeur de trait prise en compte sera celle de la base prioritaire (de l'utilisateur s'il en a mis une, sinon celle par défaut) si elle en a une, sinon la suivante, ...

```
aPourEspeceID(Esp, EspID, Base);  
aPourValeurTraitBase(Trait, EspID, Val, Base);  
not aPourValeurTraitPrioritaire(Trait, EspID, Val, Base)  
-> aPourValeurTrait(Trait, Esp, Val)  
  
valeurTraitPrioritaireSur(Trait, EspID, Val1, Base1, Val2, Base2)  
->aPourValeurTraitPrioritaire(Trait, EspID, Val2, Base2)  
  
TraitPrioritaireBase(Trait, Base1, Base2)  
aPourValeurTraitBase(Trait, EspID, Val1, Base1);  
aPourValeurTraitBase(Trait, EspID, Val2, Base2);  
-> valeurTraitPrioritaireSur(Trait, EspID, Val1, Base1, Val2, Base2)  
  
basePrioritaire(Base1, Base2);  
aPourValeurTraitBase(Trait, EspID, Val1, Base1);  
aPourValeurTraitBase(Trait, EspID, Val2, Base2);  
not TraitPrioritaireBase(Trait, Base2, Base1)  
-> valeurTraitPrioritaireSur(Trait, EspID, Val1, Base1, Val2, Base2)
```

Explication à l'utilisateur

But : Donner une justification d'une réponse à une requête de l'utilisateur en lui montrant les étapes du raisonnement qui ont permis de déduire cette réponse. Répondre à la question "Pourquoi ce fait?"

Définition : Une explication d'un fait $p(a)$ est un couple (A,R) où A est un ensemble d'atomes de la base de faits saturée et R une règle telle que appliquée à A , elle nous donne $p(a)$.

Exemple :

Soit $\mathcal{K} = (F, \mathcal{R})$ avec $F = \{t(a, b), t(a, c), q(b), q(a)\}$ et $\mathcal{R} = \{R_1, \dots, R_5\}$ défini ci-dessous.

$$R_1 = t(X, Y), q(Y) \rightarrow r(X, Y)$$

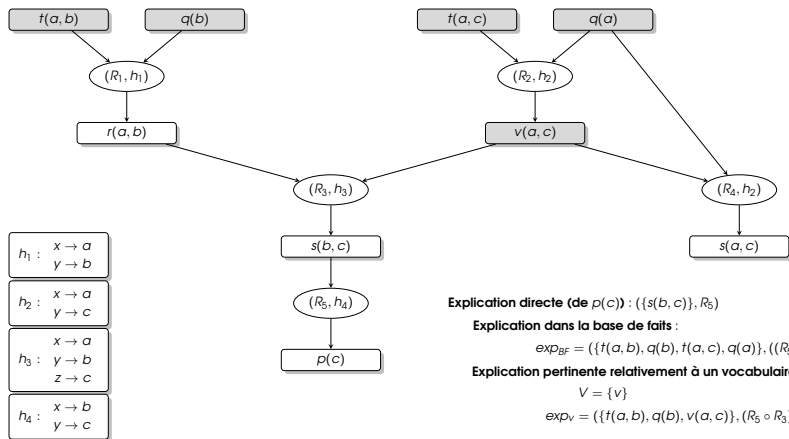
$$R_2 = t(X, Y), q(X) \rightarrow v(X, Y)$$

$$R_3 = r(X, Y), v(X, Z) \rightarrow s(Y, Z)$$

$$R_4 = v(X, Y), q(X) \rightarrow s(X, Y)$$

$$R_5 = s(X, Y) \rightarrow p(Y).$$

Explications à l'utilisateur



⇒ **Tâche 4** : proposer un mécanisme d'explication générique (basé sur une notion de pertinence)

Prise en compte des pratiques culturelles

⇒ **Tâche 5** : Intégrer dans le raisonnement les pratiques culturelles et les éléments de contexte (climat, type de sol, ...)

Une fonction potentielle peut être inhibée par des pratiques agricoles ou des éléments de contexte.

Par exemple : *"une espèce est en compétition avec la vigne sauf si elle n'est pas présente quand la vigne a besoin de la ressource"*