



Scénarisation d'interactions avec les objets du formalisme des Graphes d'Unités et prototypage d'un éditeur de définitions lexicographiques formelles

Romain Gugert

► To cite this version:

Romain Gugert. Scénarisation d'interactions avec les objets du formalisme des Graphes d'Unités et prototypage d'un éditeur de définitions lexicographiques formelles. Interface homme-machine [cs.HC]. 2013. hal-00860767

HAL Id: hal-00860767

<https://hal.inria.fr/hal-00860767>

Submitted on 11 Sep 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Stage de master 2 IAD spécialité C2IHM

Université Pierre et Marie Curie

Encadrants universitaires : Jean-Marc LABAT, Vincent CORRUBLE

Wimmics - Inria Sophia Antipolis et I3S

Encadrants de stage : Maxime LEFRANÇOIS, Alain GIBOIN

Scénarisation d'interactions avec les objets du formalisme des Graphes d'Unités et prototypage d'un éditeur de définitions lexicographiques formelles

Romain GUGERT

À Sophia Antipolis, le 2 septembre 2013

Résumé

Dans ce rapport, nous nous pencherons sur le prototypage d'un éditeur de définitions lexicographiques formelles. Cet éditeur s'appuie sur le formalisme des Graphes d'Unités (M. LEFRANÇOIS), un formalisme de représentation des connaissances linguistiques du Dictionnaire Explicatif et Combinatoire de la théorie Sens-Texte.

Dans un premier temps, nous nous intéresserons à quelques éditeurs de définitions existants qui s'appuient sur la théorie Sens-Texte. Ainsi qu'aux diverses représentations visuelles de Graphes d'Unités linguistiques existantes.

Dans un second temps, nous définirons le cadre de notre étude en listant les utilisateurs visés par cette application et les interactions nécessaires.

Puis, nous ferons un point sur les différents choix que nous avons pris pour les designs. Nous vous présenterons, ensuite, un premier prototype ainsi que les résultats d'évaluation que nous avons pu en faire.

Enfin, en nous appuyant sur ces résultats, nous proposerons des améliorations possibles ainsi qu'une nouvelle maquette.

Mots clés

Interface homme-machine, Graphes d'Unités, Théorie Sens-Texte, Dictionnaire Explicatif et Combinatoire.

Remerciements

Par ces quelques mots, je souhaite vivement remercier toutes les personnes qui m'ont entouré durant ce stage.

Je tiens à remercier tout particulièrement mes responsables de stage, Maxime LEFRANÇOIS et Alain GIBOIN, pour leur encadrement, leur aide et leur disponibilité.

Mes remerciements vont également à tous les membres de l'équipe Wimmics pour leur accueil, leur bonne humeur quotidienne et leur capacité de travail en équipe exemplaire.

J'adresse également mes remerciements à Alain POLGUÈRE et l'équipe du projet RELIEF pour le temps qu'ils m'ont accordé lors de divers entretiens et évaluations.

Je souhaiterais également remercier Nicolas Gapihan pour son sens de l'hospitalité et sa bonne humeur.

Enfin, un grand merci à ma famille qui, malgré la distance, a toujours été près de moi pour me soutenir.

Table des matières

1	Introduction	6
1.1	Contexte du stage	6
1.2	Contexte Scientifique	7
1.2.1	Théorie Sens-Texte	7
1.2.2	Les Graphes d'Unités	9
1.3	Contexte applicatif et problématique	10
2	Formalisation du sens chez les lexicographes : solutions existantes	12
2.1	Projets d'informatisation du DEC	12
2.1.1	DiCo	12
2.1.2	RELIEF	13
2.2	Représentation visuelle sous forme de graphe	15
2.2.1	Le DicoInfo Visuel	15
3	Notre solution et sa mise en œuvre	16
3.1	Analyse des utilisateurs	16
3.2	Scénarios	17
3.2.1	Scénario Actuel	17
3.2.2	Scénario Futur	18
3.3	Représentation et symboles utilisés pour formalisation	21
3.4	Maquette de l'éditeur	22
3.5	Prototype de l'éditeur	26
3.6	Évaluations du prototype	29
3.6.1	Ingénieur de la connaissance	29
3.6.2	Lexicographes	29
3.7	Proposition d'amélioration du prototype	30
4	Conclusion	34
4.1	Bilan	34
4.2	Perspectives	35
4.2.1	Dans la continuité de ce stage	35
4.2.2	Autres approches	35
4.2.3	Bases de connaissances en Graphes d'Unités : Applications	36
A	Documents de conception	39
A.1	Plan et notes de l'entretien avec Alain Polguère - 17/04/2013	39
A.2	Evaluation du Prototype - 14/06/2013	40

B	Maquettes	42
B.1	Hiérarchie des types d'unités	42
B.2	Editeur	44
C	Prototype	47
C.1	Structure actancielle de la sémantique profonde	47
C.2	Définition de la sémantique profonde	51
C.3	Structure actancielle de la sémantique de surface	54
C.4	Correspondances entre les deux niveaux de sémantique	54

Chapitre 1

Introduction

Notre stage a pour objectif la mise au point d'un éditeur de définitions lexicographiques formelles. Cet éditeur s'appuie sur la **Théorie Sens-Texte (TST)** ainsi que le formalisme des **Graphes d'Unités** (un formalisme de représentation des connaissances de la **TST**). Dans ce chapitre, nous présentons l'équipe dans laquelle ce stage a été réalisé. Puis, nous faisons un point sur le contexte scientifique en introduisant **TST** et les **Graphes d'Unités**. Enfin, nous précisons le contexte applicatif et dégagerons une problématique autour de laquelle cette étude s'articule.

1.1 Contexte du stage

Notre stage s'est effectué au sein de l'équipe Wimmics (Web-Instrumented Man-Machine Interactions, Communities, and Semantics)¹, une équipe mixte de recherche dépendante du Laboratoire I3S et du centre de recherche Inria Sophia Antipolis - Méditerranée.

Inria Sophia Antipolis - Méditerranée² est un centre de recherche impliqué dans les sciences informatiques et technologiques dans le bassin méditerranéen. Le centre rassemble plus de 600 personnes incluant 500 scientifiques, qui travaillent au sein de 38 équipes de recherche et sur 3 sites géographiques : Sophia Antipolis, Montpellier et Bologne (Italie). Le centre de recherche Inria couvre les domaines de recherche suivant : Mathématiques Appliqués ; Algorithmie, Programmation, Logiciel et Architecture ; Réseaux, Systèmes et Services, Informatique Distribué ; Sciences Informatiques appliquées à la Biologie, Médecine et l'Environnement ; Perception, Cognition, Interaction.

Le Laboratoire d'Informatique, Signaux et Systèmes de Sophia-Antipolis (I3S)³ est une unité de recherche de près de 300 personnes commune à l'Université de Nice-Sophia Antipolis (UNS) et au Centre National de la Recherche Scientifique (CNRS). L'activité scientifique du laboratoire est organisée en 4 pôles scientifiques regroupant 12 équipes de recherche (dont 4 sont communes avec Inria), elle s'articule autour de trois axes de recherche : Communica-

1. <https://wimmics.inria.fr/>

2. <http://www.inria.fr/centre/sophia/>

3. <http://www.i3s.unice.fr/I3S/index.php>

tion, Réseaux et Logiciel ; Modèles, Information et Calcul pour la Médecine et la Biologie ; Optimisation, Modélisation et Simulation du Monde Réel.

Wimmics conçoit des méthodes, des modèles et des algorithmes pour combiner la sémantique formelle et la sémantique sociale du Web. Elle axe ses recherches autour de deux sujets. Le premier est l'analyse et la modélisation des utilisateurs, communautés et de leurs interactions dans le contexte du Web. Le second cherche à formaliser et raisonner sur des graphes sémantiques hétérogènes. De plus, cette équipe de recherche participe à l'extension, la spécification, l'implémentation, le déploiement et l'enseignement des standards Web du W3C. Enfin, parmi les nombreux outils que Wimmics réalise, nous pouvons citer :

- *Corese*⁴, un moteur de recherche sémantique pour le langage RDF (Resource Description Framework) qui s'appuie sur les graphes conceptuels.
- *DBpedia.fr*⁵, une application visant à extraire un maximum de données au format RDF de la version Française de Wikipédia.
- *ZONE-project*⁶, un outil de veille qui s'appuie sur le Web sémantique dans le but d'améliorer ses capacités d'agrégation.
- *Qakis*⁷, une application qui à l'aide de traitement automatique de la langue et de requêtes SPARQL, répond aux questions de l'utilisateur.

1.2 Contexte Scientifique

1.2.1 Théorie Sens-Texte

Nous allons maintenant vous donner un rapide aperçu de la **Théorie Sens-Texte (TST)**, théorie linguistique sur laquelle repose la représentation utilisée dans ce projet. Pour cette présentation, nous nous sommes appuyés sur [Kahane \(2003\)](#).

La **TST** est un cadre linguistique pour la description du langage naturel, plus exactement un système de règles ayant pour objectif d'exprimer la correspondance entre le texte et le sens. Appliquées de façon séquentielle, ces règles permettent de passer du niveau sémantique au niveau phonétique et vice versa.

La **TST** a été introduite dans [Žolkovskij et Mel'čuk \(1965\)](#). Comme la plupart des théories linguistiques, la **TST** postule plusieurs niveaux de représentation de la langue (cf., fig. 1.1) allant de la sémantique (le sens) jusqu'au niveau phonologique (le texte). Chacun de ces niveaux, sauf le niveau sémantique, est séparé en deux : le niveau profond et le niveau de surface.

Dans le cadre de ce projet, nous nous sommes intéressés principalement à la représentation sémantique qui spécifie le sens d'un énoncé. La structure principale qui constitue la représentation sémantique est un graphe orienté. Chaque noeud d'un graphe sémantique est étiqueté par

4. <http://wimmics.inria.fr/corese>

5. <http://fr.dbpedia.org/>

6. <http://www.zone-project.org/>

7. <http://qakis.org/>

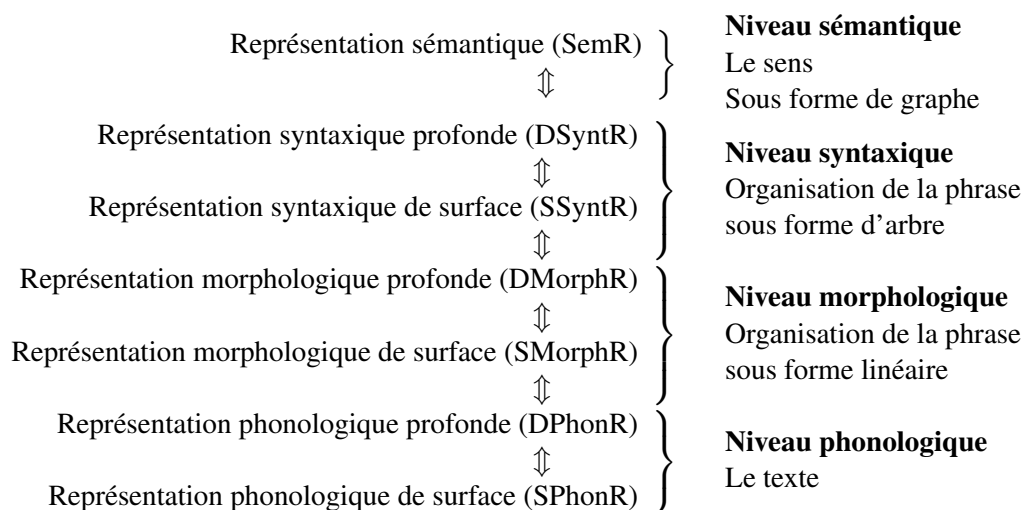


FIGURE 1.1 – Le différents niveaux de représentation de la TST

une unité sémantique spécifique à la langue qui correspond au sens du mot (en conséquence, les unités sémantiques peuvent être accompagnées d'un nombre indiquant les divers sens associés à un mot comme on peut le voir dans les dictionnaires). Chaque unité sémantique possède une structure interne appelée *structure actancielle*. Cette structure est composée de **positions actancielles (PosA)** qui définissent les *actants sémantiques* d'une unité sémantique. Les liens entre une unité sémantique et ses actants sont représentés par des arcs étiquetés des relations sémantiques (1 à 6).

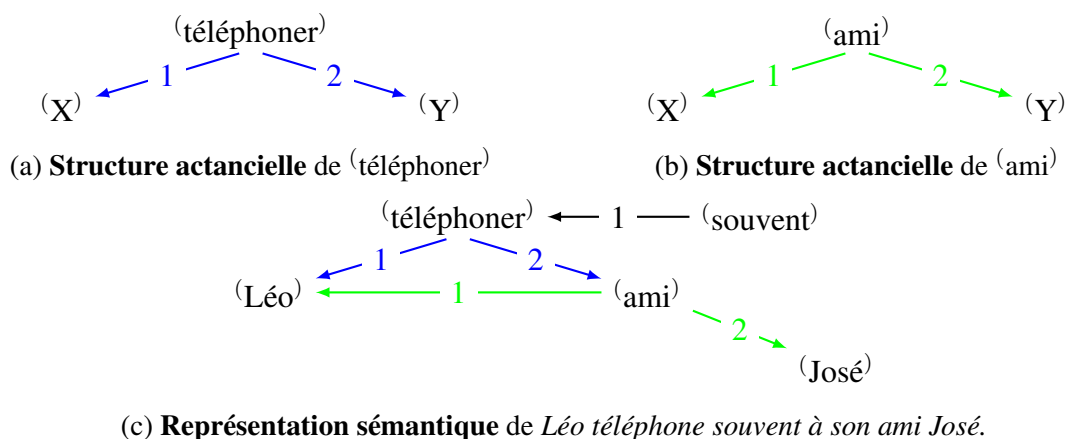


FIGURE 1.2 – Illustration de la structure de la sémantique

Enfin, le dictionnaire de la **TST** se nomme le **Dictionnaire Explicatif et Combinatoire (DEC)** qui décrit le comportement des unités lexicales de la syntaxe profonde. La branche de la linguistique qui traite de l'édition de dictionnaire est appelée la *lexicographie*. Un article dans le **DEC** se présente sous la forme suivante :

- Une partie syntaxique : le tableau de régime qui décrit les expressions possibles du sens dans les phrases.
- Les fonctions lexicales (Mel'čuk, 1996) : qui décrivent les liens entre les unités lexicales

à travers une cinquantaine d'opérateurs linguistiques. Par exemple, la fonction lexicale *Magn* qui met en avant les unités lexicales d'intensification.

- Une partie sémantique : la définition lexicographique composée de la structure actancielle, présentée sous la forme d'une formule explicitant les **PosA** qui la composent et appelée *forme propositionnelle*. Par exemple l'unité lexicale ACHETER possède comme forme propositionnelle : *X achète Y à Z*. Il est important de souligner que notre étude porte sur cette partie.

1.2.2 Les Graphes d'Unités

Dans le cadre de sa thèse, Maxime Lefrançois cherche une forme de représentation des connaissances pour la **TST**. Il a prouvé que les formalismes du web sémantique et celui des graphes conceptuels sont insuffisants pour représenter les définitions lexicographiques. Il a donc établi les bases d'un nouveau formalisme qui permet de représenter, manipuler, interroger et raisonner sur des connaissances linguistiques du **DEC** de la **TST** : le formalisme des **Graphes d'Unités (GU)** (Lefrançois et Gandon, 2013).

Tout d'abord, le formalisme des **GU** se compose d'une structure centrale nommée *hiérarchie des types d'unités* où les types d'unités et leur structure actancielle sont décrites. Un type d'unité est donc constitué d'un ensemble de **PosA**. À la différence de la **TST** où les **PosA** sont uniquement numérotées, ici elles sont nommées par un *symbole* choisi dans un ensemble appelé **symboles d'actants (SymboleA)**. De plus les **PosA** possèdent des *signatures*, ce qui permet de spécifier quels types d'unités peuvent être utilisés à cette **PosA**. Enfin, les **PosA** peuvent être de trois natures : **position actancielle optionnelle (PosAOpt)**, **position actancielle obligatoire (PosAObl)**, **position actancielle interdite (PosAInt)**.

Une relation de spécialisation est introduite sous la forme d'un pré-ordre qui modélise le fait qu'un type d'unité est plus spécifique qu'un second. Un type t_1 est plus spécifique qu'un type t_2 implique que la structure actancielle de t_1 est plus spécifique que t_2 , c'est-à-dire qu'une **PosAOpt** héritée devient une **PosAObl** ou une **PosAInt** ; ou que les signatures deviennent plus spécifiques.

Le formalisme des **GU** introduit un niveau plus profond de représentation de la sémantique qui permet de décrire le sens : la **représentation sémantique profonde (RSemP)** (Lefrançois et al., 2013). Nous établissons ainsi une distinction entre la **représentation sémantique de surface (RSemS)** et la **RSemP** et introduisons donc deux nouveaux types d'unités : pour la **RSemP** les **types d'unités sémantiques profondes (TUSemP)** notés $\langle L \rangle$; et pour **RSemS** les **types d'unités sémantiques de surface (TUSemS)** notés $\langle L \rangle$. La définition lexicographique se trouve dans ce nouveau niveau sémantique, elle se compose d'une équivalence entre deux **GU**, le premier représentant uniquement l'unité lexicale définie, le second symbolisant une paraphrase définissant l'unité sémantique.

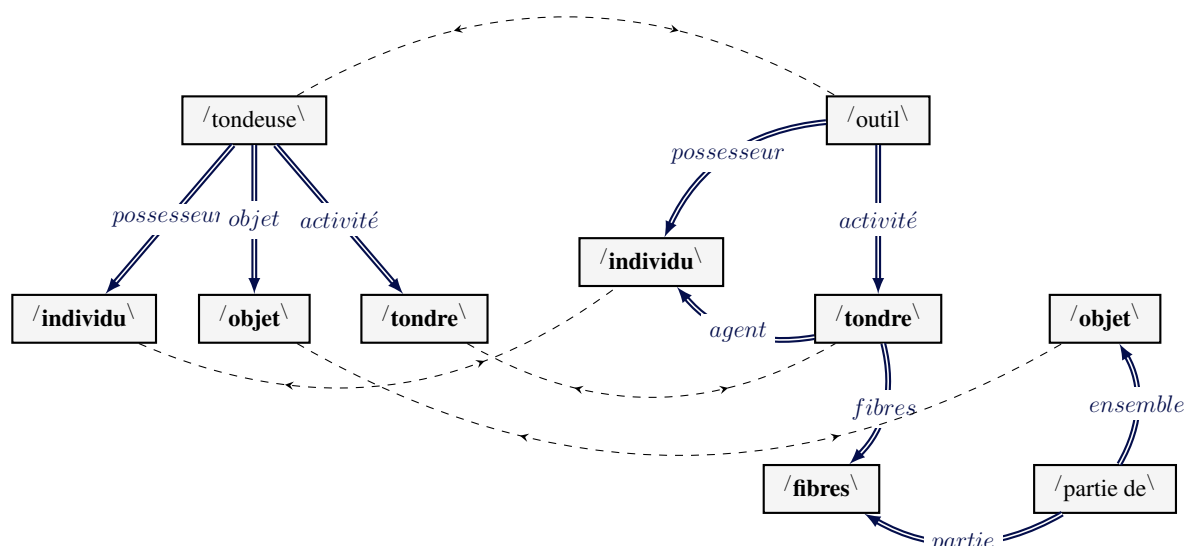


FIGURE 1.3 – Définition de /tondeuse\, à gauche l'unité sémantique définie, à droite sa paraphrase, en pointillé les relations d'équivalences

1.3 Contexte applicatif et problématique

Il y a encore peu de temps, seul le **Dictionnaire Explicatif et Combinatoire (DEC)** (Mel'čuk, 1999) offrait une représentation des entrées du dictionnaire définies selon la **TST**. Ces entrées se présentent sous la forme d'articles édités de manière textuelle et décrivant le comportement des unités lexicales d'une langue. Ce procédé se montre peu formel, la vérification de la cohérence du dictionnaire se révèle très complexe pour les lexicographes quant à son utilisation directe dans le domaine du traitement automatique des langues, elle se montre impossible sans traitement préalable.

Depuis juin 2011, un projet d'envergure de l'ATILF⁸ en lexicographie, le projet **RELIEF**, est en cours de développement. Son but est de fournir une ressource lexicale de nouvelle génération (le **Réseau Lexical du Français (RLF)**) et de travailler à la valorisation de cette ressource, notamment dans le contexte du traitement informatique de l'information textuelle. Ce projet ajoute une couche beaucoup plus formelle que le **DEC**, utilisant un éditeur informatique permettant de lier les diverses unités lexicales entre elles au sein d'un réseau construit à partir des fonctions lexicales. En revanche, ce réseau ne couvre que des informations syntaxiques. Le niveau sémantique est représenté par une forme propositionnelle et par une paraphrase écrite dans un format XML. La méthode actuellement utilisée dans ce projet ne permet pas de tisser des liens entre les unités sémantiques interprétables par une machine. Ne pouvant visualiser facilement ces liens, la tâche de contrôle des définitions se révèle difficile, voire impossible pour les lexicographes. Ce manque de formalisme au niveau de la définition dans un article du dictionnaire nous mène à nous poser la question suivante :

8. ALTIF est une équipe de recherche membre de la fédération CNRS - Institut de Linguistique Française, <http://www.atilf.fr/>

Comment amener les lexicographes à créer des définitions plus formelles dans un dictionnaire ?

Notre ambition est de fournir une ressource lexicale la plus formelle possible afin qu'elle puisse être utilisée dans le traitement automatique de la langue. Pour ce faire, nous souhaitons mettre en place un outil permettant aux lexicographes d'écrire la définition formelle selon le formalisme des **GU** qui permet de représenter des connaissances de la **TST**.

Un des premiers problèmes soulevés par cette démarche est le fait que les **GU** introduisent de nouvelles notions encore inconnues des lexicographes. Un premier objectif de notre projet est d'adapter le mode opératoire de formalisation en **GU** au mode opératoire des lexicographes. Ce qui suppose de modéliser et d'analyser ce dernier sous forme de scénarios, puis d'intégrer dans ces scénarios des opérations de formalisation en **GU**.

De ces divers scénarios, une maquette d'éditeur sera conçue. Cette maquette permettra d'évaluer si les scénarios envisagés conviennent aux lexicographes lors de l'édition de définition.

Enfin, l'idée finale de ce projet est l'intégration de cet éditeur en périphérie du logiciel (*MVS Dicet*) qu'utilise actuellement les membres du projet **RELIEF**. Notre outil permettra d'ajouter aux entrées du **RLF** des définitions décrites selon le formalisme des **GU**.

Chapitre 2

Formalisation du sens chez les lexicographes : solutions existantes

Dans ce chapitre, nous nous intéressons à deux projets d'édition de dictionnaire existants qui s'appuient sur la **TST** : **DiCo** et **RELIEF**. Nous nous appuyons sur ces deux projets afin de mettre en avant l'évolution qu'il y a eu dans ce domaine et la complexité de cette tâche. Puis nous présentons un dictionnaire possédant une représentation visuelle sous forme de graphe – le **DicoInfo Visuel** – et mettons en avant les intérêts d'une telle représentation.

2.1 Projets d'informatisation du DEC

2.1.1 DiCo

DiCo (acronyme pour dictionnaire de combinatoire) ([Polguère, 2000](#)), développé à l'OLST¹, est une base de données qui décrit le lexique noyau du français². Cette base est une version plus formelle mais moins complète que le serait un **DEC**. Il n'y a pas de véritable définition lexicographique, mais un étiquetage sémantique des unités lexicales et une description de leur structure actancielle à l'aide d'une forme propositionnelle.

Ce projet possède deux interfaces d'accès :

- le **DiCouèbe**³ est une interface permettant de consulter en ligne une version compilée du DiCo ;
- le **DiCoPOP**⁴ est une interface en ligne de présentation des données permettant un accès POPulaire (ne présuppose aucune connaissance de la **TST**).

Le DiCo est développé au moyen du logiciel *FileMaker Pro*. L'édition d'une entrée dans le DiCo se fait par un système de fiches (cf., fig. 2.1) contenant la description des unités lexi-

1. <http://olst.ling.umontreal.ca/>

2. Par lexique noyau, nous entendons que DiCo se concentre avant tout sur la description des unités lexicales courantes du français.

3. <http://olst.ling.umontreal.ca/dicouebe/>

4. <http://olst.ling.umontreal.ca/dicopop/>

cales. Chacune de ces fiches est découpée en champs qui correspondent à des catégories d'informations lexicales. Un des premiers problèmes que nous pouvons soulever sur ce système d'édition est le manque de réels liens entre les entrées et de réel formalisme avec lequel raisonner. Chacune des unités lexicales étant décrite uniquement de manière textuelle, il est difficile d'automatiser la création de ces liens et il est encore plus difficile d'en assurer la fiabilité et la cohérence.

Nom	FOUR	a	0	N	V
cg					
étq.	artefact				
fp	: ~ UTILISÉ PAR L'individu X POUR CUIRE LA nourriture OU L'artefact Y				
nb	[- SInstr(cuire)]				
tr	X = I = de N, A-pass Y = II = a+N				
Syn	{QSyn} poêle; cuisinière				
fl	/*Tel que Y est du pain*/ {Y='pain'} à pain /*Tel que Y est des briques*/ {Y='briques'} à briques /*Local dans une boulangerie où se trouve le F.*/ {Dans une boulangerie Sloc} fournil /*Au moyen d'un F./Dans un F.*/ {Adv2/Loc-in} au [-], dans [ART ~] {Qui fonctionne au bois} à bois {Qui fonctionne à l'électricité} électrique postpos {Qui fonctionne avec des micro-ondes} (a) micro-ondes /*2[X] utiliser un F.*/ {Labreal12--I} mettre [N=Y ou < dans ART> ~] //enfourner [N=X]				
ex	Une vraie miché, cuite au four à bois, est de loin meilleure que du pain industriel. Quant à la teneur en vitamines, il n'y aurait aucune différence entre les aliments cuits de manière traditionnelle et ceux cuits dans un four à micro-ondes. Rôti au four c'est également délicieux. Il me mit dans les bras un autre pain qui sortait tout juste du four.				
ph	_comme dans un four_ _être au four et au moulin_ _petit four_ ["Plus de deux cent personnes étaient invitées à déguster leurs chocolats, petits fours et gateaux dans la salle des fêtes de la ville."]				

FIGURE 2.1 – Fiche DiCo de l'unité lexicale FOURa

2.1.2 RELIEF

Dans la continuité du DiCo, nous pouvons aussi citer un projet lexicographique lancé en 2011 et mené par une équipe de quinze lexicographes de l'ATILF : **RE**ssource **L**exicale **I**nformatisée **d'**Envergure **s**ur le **F**rançais (**RELIEF**) (Lux-Pogodalla et Polguère, 2011). Son objectif est de développer une ressource lexicale nommée **RLF**. Cette ressource est sous la forme d'un graphe de type système lexical (Polguère, 2009) tissé par des liens construits à l'aide des fonctions lexicales. **RELIEF** élabore en parallèle une méthodologie et un logiciel d'édition lexicographique : *MVS Dicet*. DiCo ne s'appuyant que sur de simples fiches, nous pouvons le voir comme une première version du projet **RELIEF**. Ici, nous passons donc de fiches à un éditeur complet (cf., fig. 2.2) permettant la création de réels liens entre les entrées au sein du système d'information. Un article dans le **RLF** est constitué de six sections.

1. **Caractéristiques grammaticales** Cette partie consigne les informations d'ordre grammatical, par exemple si un nom est masculin ou féminin.

2. **Définition** À la différence du DiCo, les unités lexicales sont décrites sémantiquement à l'aide d'une définition paraphrasique. Ces définitions sont structurées à l'aide de deux composants :
 - (a) le *definiendum*, décrivant la structure actancielle de l'unité lexicale ;
 - (b) le *definiens*, décrivant la paraphrase associée à l'unité lexicale.
3. **Tableau des régimes** Cette section décrit comment les actants sémantiques de l'unité lexicale sont exprimés dans la syntaxe.
4. **Fonctions lexicales** Cette partie est le cœur du projet **RELIEF**. C'est ici que les liens entre les unités lexicales sont exprimés. Nous ne nous attarderons pas sur les liens que tissent les fonctions lexicales et quelles sont ces dernières, ce sujet n'est pas l'objet de ce stage.
5. **Exemples** Cette partie exemplifie les unités lexicales. Ce qui souvent permet de mieux cerner le sens de cette dernière.
6. **Phraseologie** Cette partie liste les expressions où l'unité lexicale apparaît. Ces expressions sont liées à l'article correspondant dans le **RLF**.

The screenshot displays the MVS Dicet interface for editing a lexical unit. The window title is 'outil - RELIEF - male (ln_fr/mysql/localhost)'. On the left, a tree view shows 'outil' expanded, with sub-items 'outil I' and 'outil II'. The main area is divided into several sections:

- Vocabulaire (28607)**: Contains fields for 'Nom' (filled with 'outil'), 'Préfixe', 'Indice', 'Exposant', 'Statut' (set to '2 : En cours de...'), 'Activité' (set to '1 : Assigné'), and 'Destinataire' (set to 'nomen'). There is also a 'Commentaires' field.
- Création par mvs le 16 juin 2011 17:08:49** and **Modification par cade le 26 mars 2013 10:17:18**.
- Acception (28605)**: Contains fields for 'Numéro' (filled with '1'), 'Statut' (set to '2 : En cours de traitement'), and 'Commentaires' (filled with 'ETIQ').
- Création par mvs le 2011-06-16 17:08:49** and **Modification par cade le 2013-03-26 10:17:18**.
- Définition**: Contains the text '~ utilisé par X=\$1 pour faire Y=\$2' and '-----DiCo :', followed by 'étiquette sémantique : instrument' and '~ UTILISÉ PAR L'individu X POUR FAIRE L'action Y(X)'. Below this is the **Tableau de régime** with the text: 'X = I = de+N | X est un professionnel [outil de menuisier], de N, A-poss', 'Y = II = de+N | Y est une activité (professionnelle) [outil de menuiserie]; pour N [outil pour le travail du bois]; à V-inf, pour V-inf [outil à/pour travailler le bois]'. At the bottom of this section are tabs for [CG], [DF], [NB], [TR], [FL], [EX], and [PH], with [CG] selected.
- Étiquette sémantique**: A table with columns 'Identifi...', 'Nom', 'Probabi...', and 'Commentaires'. The first row is empty.
- Forme propositionnelle** and **Paraphrase**: Both fields are empty.
- Commentaires**: Contains the text '~ utilisé par X=\$1 pour faire Y=\$2' and '-----DiCo :', followed by 'étiquette sémantique : instrument' and '~ UTILISÉ PAR L'individu X POUR FAIRE L'action Y(X)'.

FIGURE 2.2 – L'unité lexicale OUTIL dans l'interface de l'éditeur MVS Dicet.

2.2 Représentation visuelle sous forme de graphe

2.2.1 Le DicoInfo Visuel

Le DiCoInfo (Dictionnaire fondamental de l’informatique et de l’Internet) est un dictionnaire en ligne qui décrit les termes fondamentaux appartenant aux domaines de l’informatique et de l’Internet. L’objectif du DicoInfo *Visuel* (Robichaud, 2011) est une visualisation sous forme de graphe de l’organisation intrinsèque des informations incluses dans le DiCoInfo. Une représentation de ce type dans un dictionnaire facilite la mise en évidence et l’exploration des fonctions lexicales. Elle se montre très utile pour les lexicographes, permettant une vue d’ensemble de la cohérence et de l’exhaustivité des liens entre les unités lexicales. Ce type d’interface se révèle aussi efficace dans la compréhension du réseau lexical par tous et facilite donc l’exploration du dictionnaire.

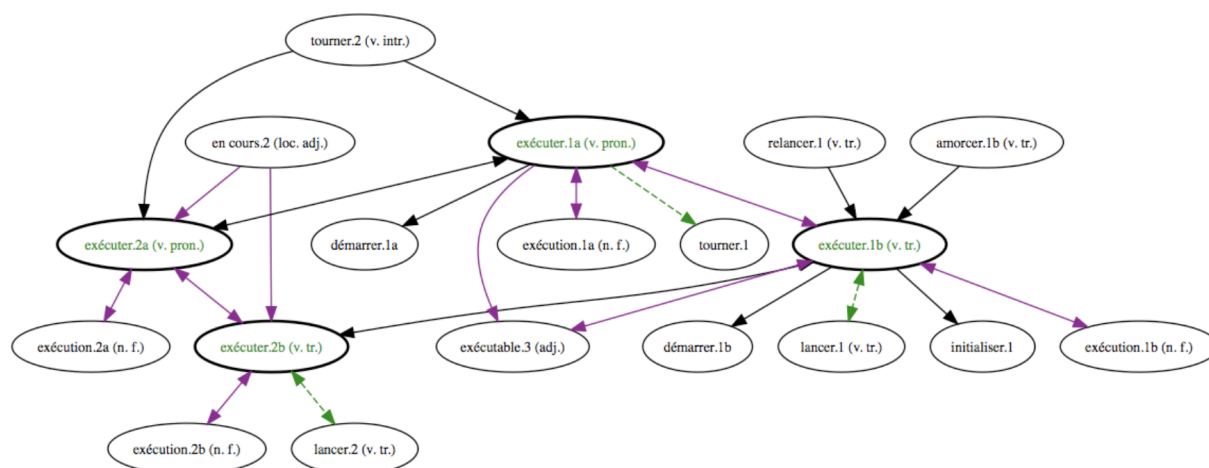


FIGURE 2.3 – Visualisation du vocable “exécuter” dans le DiCoInfo *Visuel*

Chapitre 3

Notre solution et sa mise en œuvre

Dans ce chapitre, nous vous présentons les travaux réalisés au cours de ce stage. En premier lieu, nous définissons le cadre de notre étude en décrivant les utilisateurs visés par cette application et les tâches qui leur incombent à l'aide de scénarios. En second lieu, nous décrivons les divers choix de représentation graphique que nous faisons et présentons une maquette d'un éditeur de définition lexicographique. Ensuite, nous introduisons un prototype vertical et une évaluation de ce dernier. Enfin, à partir des résultats d'évaluation, nous proposons plusieurs améliorations possibles ainsi qu'une nouvelle maquette.

3.1 Analyse des utilisateurs

Le système sera amené à être utilisé par différentes personnes ayant différents rôles dans son utilisation. Nous allons préciser ces types d'utilisateurs.

L'ingénieur des connaissances Ce système vise à créer un dictionnaire sur lequel une machine peut raisonner, un ingénieur des connaissances spécialisé dans les **GU** pourrait être amené à visualiser les données pour vérifier leurs cohérences. Il connaît le domaine de la lexicographie mais ne le maîtrise pas.

Le lexicographe Cet utilisateur est chargé de l'édition des entrées présentes dans le dictionnaire. Il ne connaît pas le formalisme des **GU**, mais il possède les bases nécessaires à l'édition lexicographique mais n'en maîtrise pas toutes les subtilités. Il est aidé dans sa tâche par l'administrateur lexicographe.

L'administrateur lexicographe Il est le spécialiste en lexicographie et le commanditaire du projet. Il forme les lexicographes et attribue et valide leurs tâches.

3.2 Scénarios

Afin de décrire et de comprendre la situation actuelle du travail de définition des lexicographes et comment ces derniers pourraient interagir avec des graphes, nous nous sommes appuyés sur le projet **RELIEF**.

Ce dernier fournit plusieurs documents¹ explicitant les différentes tâches que sont amenés à faire les membres du projet lors de l'écriture d'une définition. Nous avons aussi pu obtenir un entretien (cf., annexe **A.1**) avec Alain Polguère, le responsable scientifique du projet **RELIEF**. L'entretien s'est déroulé de façon informelle et nous l'avons conduit de manière semi-directive, à questions ouvertes. L'analyse de cet entretien nous a permis d'élaborer le scénario actuel des activités des utilisateurs concernés, celles qu'ils réalisent avec leurs outils actuels (sous-section **3.2.1**) et le scénario envisagé ou futur de leurs activités, celles qu'ils pourraient réaliser avec l'éditeur que nous envisageons (sous section **3.2.2**).

3.2.1 Scénario Actuel

À l'heure actuelle, lors de l'édition d'une définition dans le **RLF**, deux rôles se distinguent : l'administrateur et le lexicographe. Nous allons maintenant détailler les tâches qui sont attribuées à chacun dans le projet **RELIEF**.

L'administrateur Il forme actuellement les lexicographes du projet. C'est pourquoi, il possède un rôle central dans l'édition de nouvelles entrées du **RLF**. Il lui incombe d'attribuer les tâches aux lexicographes. C'est-à-dire qu'il attribue les diverses unités lexicales à chaque utilisateur.

Il possède aussi un rôle de correcteur. Il a pour rôle de valider ou invalider les modifications des entrées qui lui sont soumises. Cette validation se fait par le biais de commentaires qui sont associés à chaque entrée et qui permettent un jeu de questions-réponses.

De plus il est l'auteur de la hiérarchie des étiquettes sémantiques (Polguère, 2011) définie à l'aide du logiciel Protégé² mais en utilisant uniquement la relation *rdfs:subClassof*. Il est aussi l'auteur des patrons de formes propositionnelles.

Le lexicographe Avant tout, nous soulignons le fait que certaines des activités décrites sont réalisées à l'aide du logiciel utilisé dans le projet **RELIEF** et que d'autres sont réalisés sous forme de commentaires en attendant l'implémentation des fonctionnalités nécessaires. Ce scénario repose uniquement sur la méthodologie actuellement développé au sein de ce projet et une partie peut donc être amenée à évoluer.

Pour une question de compréhension, nous nous appuyerons sur un exemple afin de vous présenter les diverses tâches liées à l'édition lexicographique, les divers choix que nous propo-

1. <http://www.atilf.fr/spip.php?article908>

2. <http://protege.stanford.edu/>

sons ne sont que des hypothèses. L'écriture de définition repose principalement sur des critères lexicographiques que nous ne maîtrisons pas.

Soit l'unité lexicale PEIGNE_{2A} définie dans Mel'čuk et Arbatchewsky-Jumarie (1999) comme étant un *Outil de tissage qu'une personne X utilise pour démêler les fibres de l'objet Y*.

Dans un premier temps, le lexicographe sélectionne l'étiquette sémantique. Pour ce faire, il s'appuie sur la hiérarchie créée par l'administrateur. Pour PEIGNE_{2A}, la définition nous mène à choisir l'étiquette sémantique /outil\.

Dans un second temps, il détermine le nombre d'actants sémantiques afin de sélectionner la forme propositionnelle la plus adaptée parmi les patrons proposés. Dans notre exemple, nous pouvons observer qu'il y a deux actants sémantiques : la personne X et l'objet Y. La forme propositionnelle associée semble être : \sim de X pour Y \equiv /peigne_{2a}\ de la personne X pour l'objet Y.

Enfin, le lexicographe écrit la définition selon le formalisme du projet Definiens (Barque et Polguère, 2008; Barque et al., 2010). Ce formalisme est au format XML (Extensible Markup Language) et il se compose de deux types de balises :

- La composante centrale de la définition (notée <CC> </CC>) qui représente le genre prochain de l'unité lexicale. Elle comporte un attribut noté *label* qui, grossièrement, fait référence à l'unité lexicale la plus proche. Dans notre exemple l'unité lexicale la plus proche de PEIGNE_{2A} est OUTIL.
- La composante périphérique (notée <CP> </CP>) comporte un attribut noté *role* représentant un rôle sémantique. Pour PEIGNE_{2A}, nous voyons deux rôles sémantiques : le *but* qui serait de démêler les fibres de l'objet Y et le contexte qui est la production de tissus.

```
<CC label="outil">outil utilisé par une personne X</CC>
<PC role="but">pour démêler les fibres de Y</PC>
<PC role="contexte">dans la production de tissus</PC>
```

FIGURE 3.1 – Définition de PEIGNE_{2A} selon le formalisme du projet Definiens

3.2.2 Scénario Futur

Suite à la discussion avec Alain Polguère, l'hypothèse d'un outil se greffant en périphérie du logiciel MVS Dicet a été émise. Nous avons donc développé un scénario en suivant cette hypothèse. C'est pourquoi nous ne détaillerons pas les tâches de l'administrateur qui se feront en amont dans le logiciel, nous ne parlerons donc que des tâches des lexicographes. Cette partie a fait l'objet d'un article publié dont nous sommes co-auteur (Lefrançois et al., 2013).

Reprenons notre exemple avec l'unité lexicale PEIGNE_{2A} définie comme étant un *Outil de tissage qu'une personne X utilise pour démêler les fibres de l'objet Y*. Dans le scénario envisagé, nous utilisons la notion de sémantique profonde introduite par les GU, ce qui adjoint un nouveau

niveau de définition. À la différence du scénario précédent, le lexicographe commence donc par définir le **TUSemP** $/peigne_{2a}\backslash$.

Structure actancielle de la sémantique profonde Dans un premier temps, il choisit le parent le plus proche dans la hiérarchie des **TUSemP** (cf., fig. 3.2b). Cette tâche se rapproche fortement de la sélection de l'étiquette sémantique vue précédemment (cf., section 3.2.1). Dans notre exemple, le lexicographe choisit donc $/outil\backslash$. $/peigne_{2a}\backslash$ hérite alors de la structure actancielle de son parent (cf., fig. 3.2c).

Dans un second temps, le lexicographe spécifie cette structure. Il peut alors modifier la signature des **PosA** en la rendant plus spécifique : (i) en choisissant un **TUSemP** plus bas dans la hiérarchie, (ii) en ajoutant d'autre **TUSemP**. Dans notre exemple, il change la signature de *activité* vers $/démêler\backslash$ et celle de *profession* vers $/tisserand\backslash$. Il peut aussi restreindre des **PosA**. Plus exactement il modifie une **PosAOpt** héritée en **PosAObl** ou en **PosAInt**. Dans notre exemple il modifie la nature de *profession* en **PosAObl**. Enfin, il peut aussi ajouter des **PosA**, par exemple il ajoute la **PosAObl** *objet* à laquelle il associe la signature $/objet\backslash$.

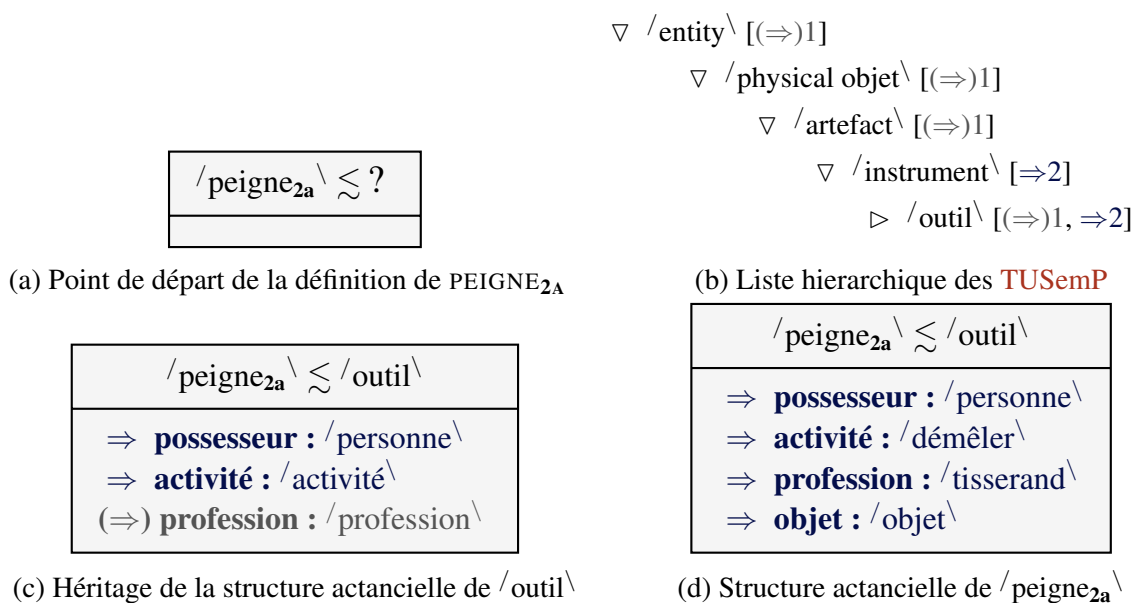


FIGURE 3.2 – Définition de la structure actancielle de $/peigne_{2a}\backslash$.

Édition de la définition formelle Une fois la structure actancielle du **TUSemP** définie, le lexicographe édite la définition. Cette interaction se fait par le biais de manipulations de graphe. Il est possible de sortir chacune des **PosA** afin d'obtenir une visualisation de leur structure actancielle (cf., fig. 3.3a). Le lexicographe peut ensuite fusionner les **PosA** des noeuds (cf., fig. 3.3b) et ajouter des noeuds au besoin (cf., 3.3c).

Pour PEIGNE_{2A}, nous fusionnons la **PosA** *agent* de $/démêler\backslash$ avec la **PosA** *possesseur*³

3. Nous faisons ici un raccourci, nous devrions plutôt parler de la position actancielle possédant le symbole *possesseur*.

de $/peigne_{2a} \backslash$, ce qui correspond au fait que la personne qui démêle est aussi le possesseur du peigne. $/démêler \backslash$ dispose de la **PosA** *fibres* cette dernière n'a pas à être fusionnée avec la **PosA** *objet* de $/peigne_{2a} \backslash$, nous ajoutons un nouveau noeud dans le graphe $/partie\ de \backslash$ dont nous fusionnons les **PosA** avec ceux de $/démêler \backslash$ et $/peigne_{2a} \backslash$.

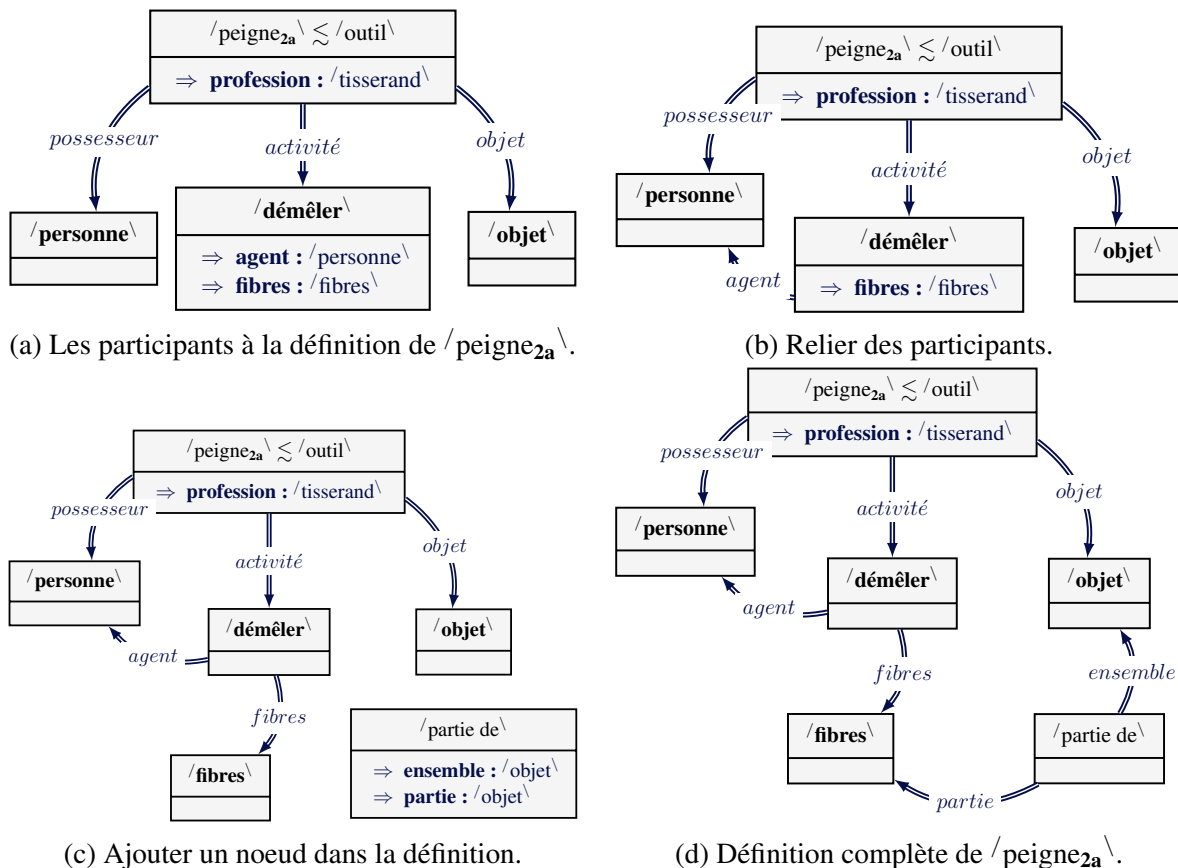
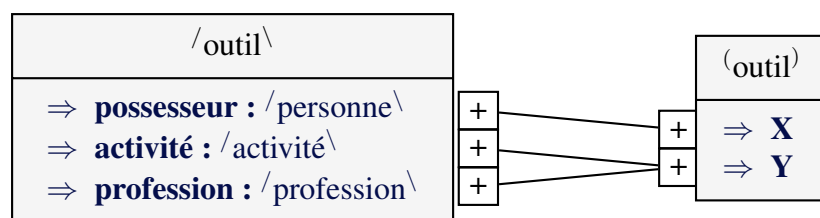


FIGURE 3.3 – Les différentes étapes dans la définition du type d'unité de la sémantique profonde $/peigne_{2a} \backslash$.

Structure actancielle de la sémantique de surface Cette seconde partie correspond à la tâche actuelle de sélection de la forme propositionnelle. Nous ne changeons donc pas le processus que les lexicographes utilisent. Ils déterminent le nombre de **PosA**, et leur nature (optionnelle ou obligatoire). Dans notre exemple, comme vu précédemment, le **TUSemS** ($peigne_{2a}$) possède deux **PosA** obligatoires : X la personne et Y l'objet.

Correspondance entre les deux niveaux de sémantique Enfin, une dernière étape consiste à lier les **PosA** des deux niveaux de sémantique. La liaison semble dans notre exemple plutôt évidente, la **PosA** du **TUSemS** (**PosASemS**) X est liée à la **PosA** du **TUSemP** (**PosASemP**) *posseur* et la **PosASemS** Y à la **PosASemP** *objet*. En revanche, il existe des cas où le choix se montre moins évident, il se peut qu'une **PosASemS** corresponde à plusieurs **PosASemP** (cf., fig. 3.4a).



(a) Correspondance entre les PosASemP et les PosASemS de TOOL.

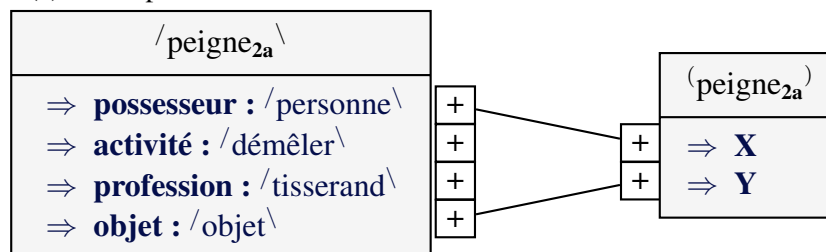
(b) Correspondance entre les PosASemP et les PosASemS de PEIGNE_{2A}.

FIGURE 3.4 – Correspondance entre la structure actancielle d'un TUSemS, et la structure actancielle du TUSemP associé.

3.3 Représentation et symboles utilisés pour formalisation

Dans un premier temps, nous nous sommes penchés sur la représentation du formalisme des **GU**. Afin de répondre à cette problématique, nous avons d'abord étudié les nombreux symboles utilisés dans la lexicographie et dans la **TST**, ceci dans l'objectif d'éviter toute confusion de la part des lexicographes. De plus, nous avons étudié une modélisation de la hiérarchie des types d'unités (cf., Annexe **B.1**) sous forme de graphe afin de comparer les divers symboles et représentations.

Les types d'unités Un point important est la représentation des unités lexicales et unités sémantiques. Une unité lexicale est toujours écrite en majuscules, elle peut être entourée de «¹ ». Nous suivons donc logiquement cette représentation (ex. «¹PEIGNE_{2A}»). Concernant la représentation des **TUSemS**, étant donné que nous nous plaçons dans un niveau de représentation équivalent au niveau sémantique de la **TST**, nous gardons la représentation actuelle, c'est-à-dire l'unité lexicale en minuscules (optionnellement accompagnée d'un numéro pour désambigüiser les divers sens) entourée de «^()» (ex. «^()peigne_{2a}»). Enfin pour la représentation des **TUSemP**, nous décidons d'introduire un nouveau symbole en nous appuyant sur le même principe que les deux notions vues précédemment. Étant à un niveau de représentation linguistique, nous gardons l'unité lexicale en minuscules et entourée de deux caractères en exposants : «^{/peigne_{2a}\}».

La nature des positions actancielle Nous associons à chaque nature des positions actancielle un symbole commun : la double flèche \Rightarrow . Celle-ci symbolise ainsi le lien entre un **TUSemP** et ses **PosA**. Le fait que le symbole soit identique aux trois natures des **PosA** permet de rendre compte que les éléments appartiennent à une même catégorie. Dans ce rapport

nous ne nous soucions pas des arbres syntaxiques mais il est important de souligner qu'il existe d'autres objets (ex. les *circonstants*) qui ne sont pas des **PosA**, dans leur représentation nous pourrions dans ce cas utiliser une simple flèche afin de distinguer les deux objets. Concernant la distinction entre les trois natures des **PosA**, nous choisissons un code couleur. Aux **PosAOpt** nous attribuons la couleur grise, qui est une couleur neutre et marque le côté facultatif de la **PosA**. Pour les **hyperlinkposablPosAObl**, la couleur bleu leur est affectée. Cette dernière est une couleur primaire et exprime une contrainte, par exemple en France les panneaux routiers catégorisant une obligation sont de couleur bleu. Enfin, nous assignons la couleur rouge aux **PosAInt**. Une couleur symbole d'interdiction en France. Enfin, pour les **PosAOpt**, nous adjoignons au symbole de la double flèche des parenthèses (ex. (\Rightarrow)), ceci dans l'objectif d'accentuer l'aspect optionnel et mettre plus en avant cette position qui est la seule à pouvoir changer de nature. De plus, les actants sémantiques optionnels sont placés entre parenthèses dans une forme propositionnelle dans la **TST**.

Les signatures Le cas des signatures est plutôt simple à traiter. Ces dernières, dans le cadre de la sémantique profonde, sont composées de **TUSemP**. Nous nous tournons donc naturellement vers la même représentation : `/signature\`. La seule réelle question soulevée pour la représentation des signatures est : Comment symboliser un ensemble de signatures ? Tout simplement comme les ensembles mathématiques ($\{ \}$). Nous avons remarqué que cette représentation est connue et utilisée en lexicographie.

Noeud dans le graphe Nous choisissons une représentation semblable aux diagrammes de classe UML. Une boîte coupée en deux parties, l'une permettant d'afficher le **TUSemP** (remplaçant le nom de la classe), l'autre contenant la structure actancielle (remplaçant attributs d'une classe). Cette représentation met en avant l'aspect intrinsèque de la structure actancielle et permet d'afficher toutes les informations concernant cette dernière.

<code>/tusemp\</code>	
<code>(\Rightarrow) SymboleA : /signature\</code>	position actancielle optionnelle
<code>\Rightarrow SymboleA : /signature\</code>	position actancielle obligatoire
<code>\Rightarrow SymboleA : { /signature\, /signature\ }</code>	position actancielle interdite

FIGURE 3.5 – Représentation d'un TUSemP

3.4 Maquette de l'éditeur

Dans cette section, nous vous présentons les divers éléments de la principale maquette élaborée. Pour cette présentation, nous suivons le scénario futur proposé dans la section 3.2.2. Nous

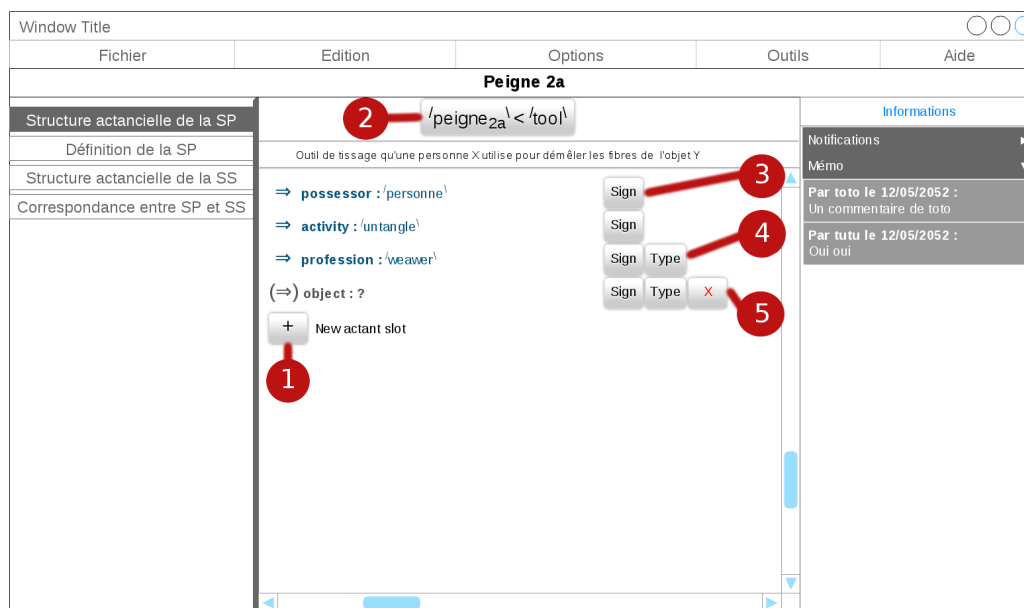
utilisons un menu possédant quatre onglets, représentant chacune des tâches que nous proposons. Ceci amène l'utilisateur novice à suivre le flux de travail. Nous faisons ce choix puisque beaucoup de nouvelles notions sont introduites avec le formalisme des **GU** et nous souhaitons donc guider l'utilisateur au maximum.

Structure actancielle de la sémantique profonde Nous proposons pour cette tâche l'interface présentée dans la figure 3.6a. Cette dernière se constitue d'un élément central représentant la structure actancielle de l'unité lexicale. Une première interaction possible est le choix d'un parent hiérarchique, nous ajoutons donc le bouton ② de la figure 3.6a contenant le nom de l'unité lexicale suivi d'un signe inférieur puis soit de l'ensemble des parents hiérarchiques sélectionnés, soit "Parent Non Défini". La sélection se fait par le biais d'une fenêtre modale (cf., fig. 3.6b), dans laquelle nous présentons tous les **TUSemP** dans une liste hiérarchique (nous proposons aussi de la présenter sous forme de liste alphabétique), à laquelle nous ajoutons une fonctionnalité de recherche. Si le **TUSemP** recherché n'est pas présent, il est alors possible de l'ajouter (cf., fig. 3.6c). Cet ajout se fait sous certaines conditions notamment une validation de la part de l'administrateur. En attendant cette validation le parent hiérarchique estimé le plus proche par l'utilisateur sera utilisé. Pendant la sélection des parents hiérarchiques, en cas de risque d'inconsistance, un point d'exclamation entouré d'un triangle jaune apparaît alors sur le bouton de validation. L'information concernant cette probable erreur est dévoilée au passage de la souris et dans une nouvelle fenêtre modale de confirmation lors de la validation.

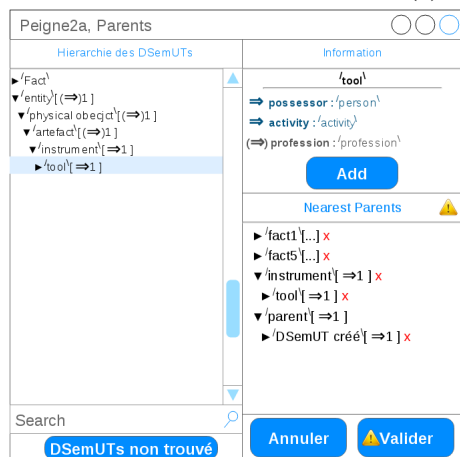
Une fois l'ensemble des parents hiérarchiques sélectionnés, la structure actancielle héritée est affichée. Cette structure est éditée par le biais de différents boutons, nous proposons deux dispositions différentes. La première (cf., fig. 3.6a) où toutes les actions possibles⁴ sont présentes à droite de la **PosA**. La seconde où le symbole représentant la nature et l'ensemble des signatures de la **PosA** sont des boutons, nous ajoutons aussi à la place des boutons d'action de la première solution un bouton d'édition et de suppression de la **PosA**. La modification des signatures se fait par le biais du bouton ③ de la figure 3.6a qui affiche une fenêtre de sélection modale semblable à celle pour le choix des parents hiérarchiques (cf., fig. 3.6b). Quant au changement de nature de la **PosA**, il s'effectue suite à l'activation du bouton ④ de la figure 3.6a ce qui change le symbole et la couleur de la **PosA**.

Enfin l'ajout d'une nouvelle **PosA** s'effectue à l'aide du bouton ① de la figure 3.6a. Une fenêtre modale d'édition de la nouvelle **PosA** est affichée. L'utilisateur sélectionne le **SymboleA**. Les **SymboleA** sont présentés sous forme de liste, les informations relatives à chacun sont affichées suite à une première sélection. Un bouton de validation permet de choisir le **SymboleA** sélectionné précédemment.

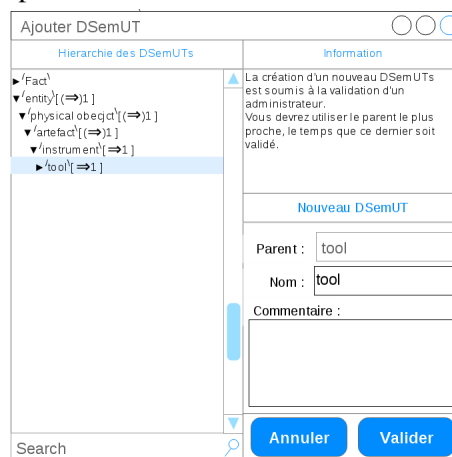
4. Pour rappel les actions d'édition d'une position actancielle sont le changement de nature et la spécification des signatures.



(a) Interface principale



(b) Choix du parent hiérarchique le plus proche



(c) Ajout d'une unité de la sémantique profonde

FIGURE 3.6 – Maquette de l'édition de la structure actancielle de la sémantique profonde

Édition de la définition formelle Pour cette tâche, nous proposons la maquette en figure 3.7. Il s'agit d'une représentation sous forme de graphe de la définition à partir de laquelle chaque élément peut être déplacé, fusionné, etc⁵. Un premier point important est la barre d'outils, les actions suivantes y sont possibles : (i) sortir/ranger toutes les positions actancielle d'un noeud (cf., ④ et ⑤, fig. 3.7) ; (ii) réarranger le graphe horizontalement (cf., ①, fig. 3.7), verticalement (cf., ②, fig. 3.7) ou en étoile (cf., ③, fig. 3.7) ; (iii) ajouter un noeud (cf., ⑥, fig. 3.7). Concernant la fusion de deux noeuds, nous proposons une action de glisser-déposer. Suite à cette action la nouvelle signature sera recalculée et affichée. Un glisser-déposer du nouveau noeud sur l'ancien parent annulera l'action. L'action pour ajouter un noeud affiche une liste des différents **TUSemP**, un glisser-déposer d'un élément de la liste dans le graphe ajoutera un noeud. Nous ajoutons à nouveau un système de notifications permettant de prévenir l'utilisateur en cas d'in-

5. Les diverses manipulations possibles sur le graphe de définitions sont détaillées dans la section 3.2.2

consistance. Les arcs problématiques apparaîtront alors en jaune, le passage du curseur sur un lien affichera le message d'information concernant cette inconsistance.

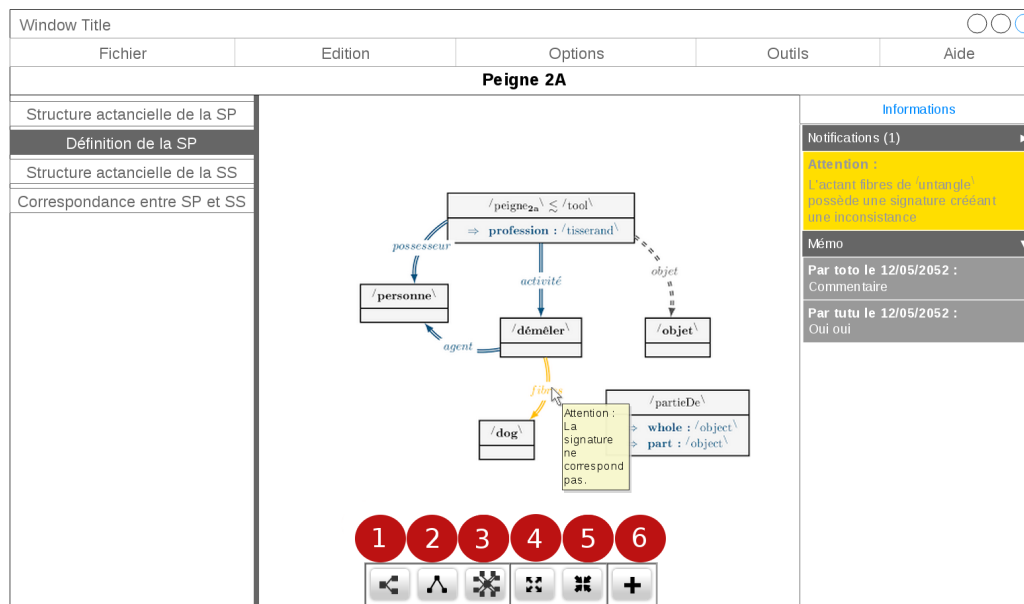


FIGURE 3.7 – Maquette de l'édition de la définition de la sémantique profonde

Structure actancielle de la sémantique de surface Dans cette maquette (cf., fig. 3.8), nous recherchons un maximum de simplicité. Lors de cette tâche, le point central et critique est le choix du nombre de **PosA**. Nous plaçons donc au centre de cette interface le nombre d'actants (cf., ①, fig. 3.8), suivi de deux boutons plus et moins (cf., ③ et ②, fig. 3.8) permettant l'ajout ou la suppression de **PosA**. Il est aussi possible d'entrer directement le nombre d'actants dans le champ indiquant ce nombre. Nous aurions pu nous appuyer sur les patrons de formes propositionnelles de MVS Dicet et les suggérer à l'utilisateur, mais à l'heure actuelle, ces patrons ne sont pas encore implémentés et les formes propositionnelles n'apportent aucune information dans le formalisme des **GU**. En revanche, nous ajoutons une zone de texte (cf., ⑤, fig. 3.8) dans laquelle l'utilisateur entre les informations qui lui semblent utiles. Par exemple, il peut ainsi nommer les **PosA**. Enfin, il est possible de changer la nature d'une **PosA** en actionnant le bouton sur le symbole représentant cette dernière (cf., ④, fig. 3.8).

Correspondance entre les actants des deux niveaux sémantiques La dernière étape de notre maquette correspond à la mise en correspondance des **PosA** des deux niveaux de sémantique (cf., fig. 3.9). Pour ce faire nous symbolisons donc le **TUSemP** et le **TUSemS** sous une représentation de type UML (cf., section 3.3). Devant chacune des **PosA**, nous ajoutons un bouton (cf., ①, fig. 3.9) qui suite à un clic permet de tracer un arc vers la position actancielle correspondante. Si le deuxième clic se produit à l'extérieur d'un bouton, le trait disparaît (annulation de l'action en cours). Nous ajoutons la possibilité de supprimer un arc en cliquant sur

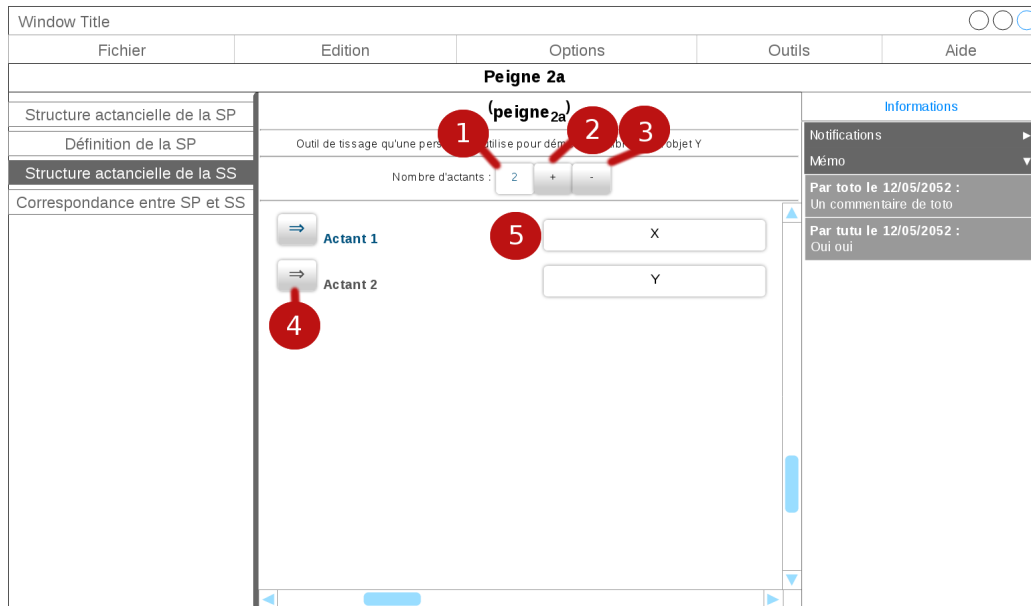


FIGURE 3.8 – Maquette d'édition de la structure actancielle de surface

ce dernier. Dernièrement, en cas d'erreur de la part de l'utilisateur une fenêtre modale apparaît lui indiquant la raison de cette erreur puis l'arc disparaît.

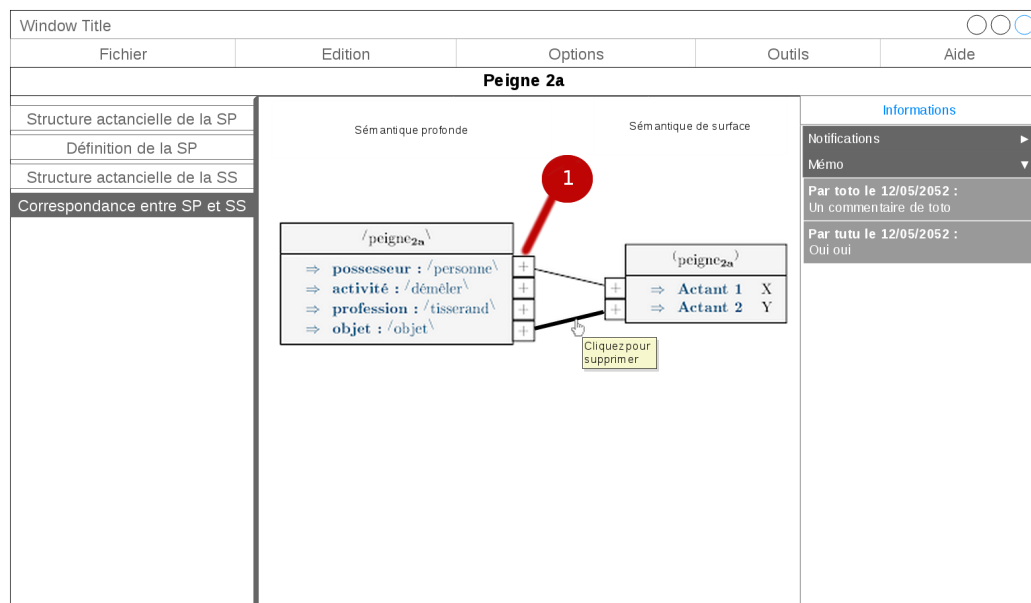


FIGURE 3.9 – Maquette de l'édition des correspondances entre les deux niveaux sémantiques.

3.5 Prototype de l'éditeur

Afin d'introduire cette section, nous évoquerons le fait que le prototype présenté est disponible à l'adresse suivante : <http://romain.gugert.fr/unitgraphs/>. De plus, une

vidéo de démonstration est publiée sur le site de l'équipe Wimmics⁶ et nous l'avons présentée lors de la *3e journée RELIEF*⁷ (13 juin 2013, Nancy).

Lors de votre lecture, vous pourrez remarquer que nous n'avons pas suivi un schéma classique dans la conception d'interface homme-machine puisque nous sommes passés très rapidement au développement d'un prototype. L'une des premières justifications que nous apportons est temporelle. Effectivement, nous n'avons que peu de contacts avec des lexicographes, nous voulions donc mettre à profit notre rencontre avec l'équipe du projet **RELIEF**. Une maquette n'aurait pas été suffisante pour la partie manipulation du graphe lors de l'étape d'édition de la définition formelle. De plus, nous avons choisi de présenter un *prototype vertical*, c'est-à-dire de réduire le nombre de fonctionnalités proposées par rapport à la maquette, car nous ne voulions pas surcharger les utilisateurs d'informations. Le formalisme introduit des concepts encore inconnus des utilisateurs, nous souhaitons avant tout évaluer les interactions, le vocabulaire et les symboles utilisés. Nous avons donc supprimé la possibilité de choisir plusieurs parents hiérarchiques, plusieurs signatures pour une même **PosA** ainsi que le système de notification d'inconsistance.

Concernant les technologies utilisées pour le développement du prototype, nous avons pris la décision de faire une application en ligne. Ce choix nous est apparu le plus cohérent, nous permettant de faire tester à distance l'interface par les utilisateurs. De plus, aucune installation particulière hormis un navigateur web à jour n'est nécessaire. Ayant pour première idée une intégration du système final au sein du projet **RELIEF**, nous nous sommes naturellement orientés vers une base données MySQL qui est le type de base utilisé par **RELIEF**. Côté serveur, nous avons utilisé CodeIgniter⁸, un cadre de développement (framework) PHP. Il a pour avantage d'être libre, de s'appuyer sur une architecture MVC (Modèle-Vue-Contrôleur) et une architecture REST (Representational State Transfer), d'être minimaliste mais puissant et d'être bien documenté. Côté client, pour le JavaScript, nous nous sommes servis de jQuery⁹ qui a pour avantage d'être léger, de fonctionner sur les navigateurs webs les plus utilisés et de posséder beaucoup de modules d'extensions maintenus par une grande communauté. Pour le CSS, nous nous sommes appuyés sur Bootstrap¹⁰, qui permet un développement rapide d'interface sensible à la taille de la fenêtre et garanti un aspect unique quelque soit le navigateur grâce à une compatibilité totale intégrée.

Nous allons maintenant vous présenter les différentes interfaces proposées dans le prototype, les ajouts et suppressions que nous avons effectué entre ces dernières et les interfaces de la maquette proposée dans la section 3.4. Concernant le prototype dans son ensemble, nous avons utilisé un des requêtes AJAX (Asynchronous JavaScript and XML), ceci permettant une interactivité sans changer de page.

6. <http://wimmics.inria.fr/doc/video/UnitGraphs/editor1.html>

7. <http://www.atilf.fr/spip.php?article3895>

8. <http://ellislab.com/codeigniter>

9. <http://jquery.com/>

10. <http://getbootstrap.com/2.3.2/>

Structure actancielle de la sémantique profonde Ici, nous avons opté pour la deuxième solution proposée dans la maquette (cf., annexe C.1). Nous avons utilisé des boutons directement sur chacune des entités éditables plutôt qu'une série de boutons à proximité de chacune des PosA. Concernant les modifications apportées par rapport à la maquette, nous avons supprimé la possibilité d'hériter de plusieurs parents, ainsi que la possibilité de posséder un ensemble de TUSemP comme signature. Ceci permettant une interface minimaliste ce qui nous permet d'approcher de notre objectif de seulement introduire l'utilisateur à ce nouveau formalisme. Dans la continuité de cet objectif, nous avons ajouté des bulles d'informations sur chacun de ces boutons (cf., fig. 3.10), permettant de mettre en exergue l'action associée ou la justification si cette dernière n'est pas possible. Encore une fois, nous justifions ce choix par la nouveauté du formalisme des GU et l'idée d'une interface permettant l'apprentissage de ce dernier.



FIGURE 3.10 – Bulles d'informations contenues dans l'éditeur de la structure actancielle de la sémantique profonde

Écriture du graphe de définition En ce qui concerne cette interface (cf., annexe C.2), nous avons utilisé *mxGraph*¹¹ qui est une librairie JavaScript permettant la visualisation et la manipulation de graphes. Fonctionnant uniquement du côté client, le choix de cette librairie nous apparaît idéal. Les changements apportés, en comparaison avec l'interface proposée dans la maquette, se trouvent principalement au niveau de la barre d'outil à laquelle nous ajoutons deux fonctionnalités. La première est la possibilité de supprimer un noeud : il est possible d'ajouter un noeud, il est donc indispensable de pouvoir le supprimer. La seconde est l'annulation et le rétablissement des dernières actions effectuées. Notre objectif étant de proposer un prototype minimaliste, on pourrait penser que cette fonctionnalité est secondaire, mais elle se montre essentielle puisqu'elle permet à l'utilisateur de restaurer le système dans un état précédent rapidement. En revanche, les graphes manipulés dans le prototype étant de petite taille, nous avons

11. <http://www.jgraph.com/mxgraph.html>

pris le parti de ne pas implémenter les fonctionnalités de réorganisation du graphe.

Structure actancielle de la sémantique profonde et Correspondance entre les deux niveaux de sémantique À ce sujet, les interfaces que nous proposons (cf., annexes C.3 et C.4) sont identiques à celles présentées dans la maquette. Les choix que nous exprimons restent les mêmes que ceux vu dans la section 3.4. Par conséquent, nous ne nous attarderons pas sur ces derniers.

3.6 Évaluations du prototype

3.6.1 Ingénieur de la connaissance

Les évaluations que nous avons effectuées sur les utilisateurs de type ingénieur de la connaissance ne sont pas réellement révélatrices, pour la simple raison que la seule personne experte dans le domaine des GU est l’auteur du formalisme. De plus, cette personne a été incluse dans toutes les phases du développement. Comme l’utilisateur connaissait déjà le prototype, ces évaluations sont biaisées. En revanche, il était important d’avoir son avis en tant qu’utilisateur potentiel de l’éditeur puisqu’il est le seul capable d’apporter un point de vue expert en GU.

De nos diverses discussions avec l’expert, nous avons pu constater qu’il ne possède aucune difficulté avec la RSemP, le processus proposé lui semble adapté s’il devait être amené à créer une nouvelle définition. Concernant la représentation et la manipulation par le biais de graphe, cette solution semble être la plus adéquate. De plus la représentation de type diagramme de classe lui est familière, pour cet utilisateur cette modélisation est l’une des plus appropriées.

3.6.2 Lexicographes

Dans cette section, nous vous présentons l’évaluation du prototype que nous avons effectuée auprès de cinq membres de l’équipe RELIEF, ainsi que la conclusion que nous tirons suite à la discussion menée avec le responsable scientifique du projet à l’issue de la présentation que nous avons effectuée lors de la 3e journée RELIEF.

Concernant le profil des utilisateurs, tous sont formés depuis deux ans par le responsable scientifique du projet. Leurs cursus avant de rejoindre cette équipe sont variables, certains travaillaient sur des projets de traitement automatique du langage, d’autres possèdent une formation spécialisée dans la lexicographie et d’autres uniquement dans le niveau syntaxique. Étant donné qu’ils participent tous au même projet, où ils effectuent tous les mêmes tâches et possèdent tous au moins une expérience de deux ans dans la lexicographie, nous choisissons de les classer tous dans une même catégorie d’utilisateur.

À propos de la démarche que nous avons adoptée, nous avons conçu une expérience afin d’évaluer : (i) la clarté, pour les lexicographes, de la notion de sémantique profonde ; (ii) le

processus de définition proposé ; (iii) l'interface du point de vue de son utilisabilité. Pour ce faire, dans un premier temps, nous leur avons présenté le scénario d'édition de définition et nous avons discuté des notions abordées et de l'ordre des tâches présentées. Dans un second temps, nous leur avons proposé de créer une définition selon ce modèle à l'aide de notre outil, à chaque étape du processus nous nous sommes intéressés aux points suivants : (i) la tâche est accomplie ; (ii) les termes et symboles sont compréhensibles ; (iii) l'information recherchée est claire.

De cette évaluation beaucoup d'éléments sont ressortis (cf., Annexe A.2). Ici, nous présentons une synthèse et ne détaillons que les points les plus importants. Premièrement, nous nous apercevons que notre démarche est problématique, nous partions avec des idées préconçues notamment que les lexicographes possèdent l'habitude de manipuler des graphes. Les lexicographes représentent très souvent leurs modèles sous forme de graphes dans la littérature, mais n'en manipulent que très rarement, préférant un contexte textuel. En deuxième lieu, le scénario futur proposé ne correspond pas tout à fait aux lexicographes. Le besoin de définir le nombre de positions actanciellles de la **RSemS** très tôt dans le processus s'est révélé très important. De plus, bien qu'ils aient conscience de l'héritage des structures actanciellles, la visualisation de ces dernières ne doit se faire que sous forme de proposition et non apparaître dans la structure de l'unité sémantique éditée. Enfin, il apparaît que les lexicographes ne sont pas prêts à passer à un niveau de représentation plus profond que le niveau actuel, cela impliquerait le développement de nouvelles méthodologies d'édition d'entrées et donc trop de changements dans le projet **RELIEF**.

3.7 Proposition d'amélioration du prototype

La **RSemP** est un niveau de représentation auquel les lexicographes ne sont pas encore formés. Pour accélérer la mise en production, Alain Polguère nous a demandé d'envisager une solution qui conserve la **RSemP** en arrière-plan. Il s'agirait donc d'extraire une partie de la sémantique profonde à partir des divers éléments constitutifs de la sémantique de surface.

Suivant cette hypothèse, nous nous tournons alors vers un nouveau modèle beaucoup plus proche de la méthodologie actuelle du projet **RELIEF**, quitte à perdre une partie des informations nécessaires à l'édition complète des **GU**. Par exemple, dans le graphe représentant la paraphrase, le fait que les signatures de deux **PosA** sont liées. Notre première idée est de revoir le flux de travail. Nous proposons donc d'effectuer les tâches associées à la **RSemS**, qui sont des tâches connues des lexicographes. Pour ce faire, nous ajoutons quelques éléments à leur travail actuel, la nouvelle solution que nous proposons se déroule en quatre étapes. La première étape, ils choisissent le parent le plus proche (nommé *Gener*), cette étape se rapproche fortement du choix du parent hiérarchique vu dans le scénario futur (cf., section 3.2.2). Dans la seconde étape, ils sélectionnent la forme propositionnelle parmi une liste de patrons existants où ceux se reprochant du parent sont mis en avant. Puis, ils définissent la signature des actants (nous nommons

cette partie typage des actants puisqu'il s'agit du terme utilisé par les lexicographes). Enfin ils éditent le *definiens* selon une représentation semblable au formalisme XML qu'ils utilisent dans le projet **RELIEF**.

En ce qui concerne la maquette, nous proposons une simple page d'édition (cf., 3.11) accompagnée de fenêtres modales. L'avantage d'une page unique est le regroupement des informations. Cela permet de visualiser les divers éléments constituant la définition dans une même fenêtre et donc d'en assurer plus facilement la cohérence à un niveau micro.

FIGURE 3.11 – Ecran principal

Pour la sélection du *Gener*, nous proposons deux interactions. La première s'effectue par le biais d'une fenêtre modale (cf., fig. 3.13a), dans laquelle l'utilisateur navigue dans une liste où il choisit l'unité lexicale la plus proche, cette fenêtre se rapproche fortement du système utilisé dans *MVS Dicet* pour naviguer dans la nomenclature¹². La seconde méthode (cf., fig. 3.12) se réalise plus rapidement que la première mais fournit moins d'information sur l'unité lexicale sélectionnée. Elle débute en entrant les premières lettres de l'unité lexicale cherchée puis en cliquant sur l'icône de recherche. Apparaît alors un menu déroulant listant les unités lexicales (sans leur numéro de désambiguïsation) commençant par ces lettres. Au passage de la souris sur une de ces unités lexicales, la liste de ces unités lexicales désambiguïsées se dévoile dans un sous menu et la sélection peut alors avoir lieu par un clic.

La tâche de sélection de la forme propositionnelle s'effectue à l'aide d'une fenêtre modale (cf., fig 3.13b). Une liste de patrons triée suivant le nombre d'actants sémantiques y est présente. Lors de la sélection de la forme propositionnelle, la zone de typage des actants se met à jour, il y apparaît chacun des actants. Si le *Gener* sélectionné possède lui aussi un actant de même nom (X, Y, ..), le type (signature) du parent y est affiché. Il est alors possible de modifier les types de chacune des **PosA** en effectuant le même procédé que pour la sélection du *Gener*.

12. Le terme nomenclature désigne l'ensemble des unités lexicales composant un dictionnaire.

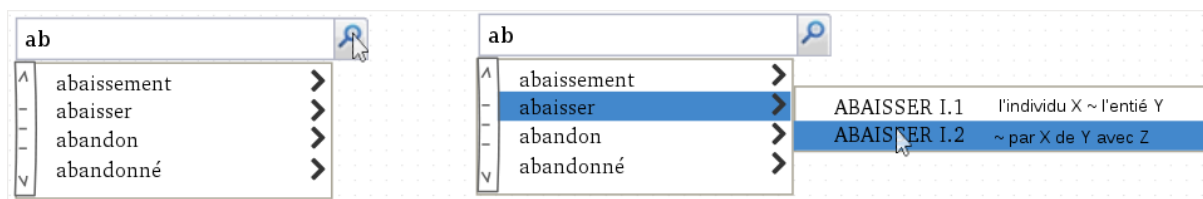
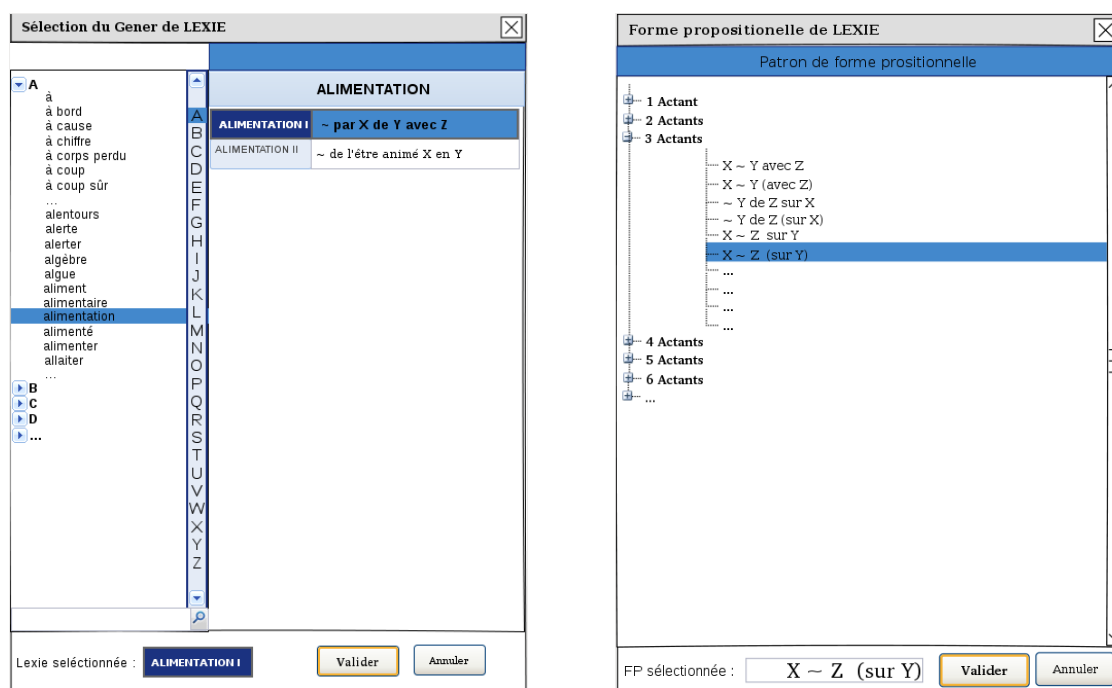


FIGURE 3.12 – Méthode avancée de la sélection d'une unité lexicale

Enfin, l'édition du *definiens* se fait par le biais d'un formulaire. Une première partie décrit la composante centrale, elle se compose de deux champs permettant à l'utilisateur de spécifier le label et le contenu de la balise. Quant aux composantes périphériques, un bouton permet d'en ajouter le nombre désiré. Un premier champ permet de renseigner le rôle, lorsque l'utilisateur entre une donnée, les rôles du Gener sont alors proposés. Cette fonctionnalité permet de suggérer des **SymboleA** qui seront forcément hérités sans forcer la main au lexicographe.



(a) Sélection du Gener

(b) Sélection de la forme propositionnelle

FIGURE 3.13 – Maquettes des fenêtres modales

Concernant la structure des informations que nous pouvons obtenir à l'aide d'une telle interface, elle se présente sous la forme d'un graphe. Par exemple, prenons l'unité lexicale PEIGNE_{2A} possédant les éléments définitionnels suivants :

- Gener : OUTIL ;
- Forme propositionnelle : /peigne_{2a} \ de X_{personne} pour Y_{objet} ;
- Definiens :

```
<CC label="outil">outil utilisé par une personne X</CC>
<PC role="but">pour démêler les fibres de Y</PC>
<PC role="contexte">dans la production de tissus</PC>
```

De la forme propositionnelle, le fait que deux **SymboleA**, X et Y, possèdent comme signature `/personne\` et `/objet\` est facilement déductible. De plus, il est possible d'extraire les **PosA** possédant comme **SymboleA** but et contexte. En revanche, inférer leur signature `/démêler\` et `/production\` se révèle extrêmement délicat, surtout dans le cas où plusieurs unités sémantiques sont associées à une même unité lexicale. De plus, une relation entre `/démêler\` et `/objet\` (également entre `/production\` et `/tissus\`) est visible mais l'extraire se révèle une tâche complexe. Il faudrait que la structure de `/démêler\` (`/production\`) soit définie et créer un système de règles permettant d'induire les liens entre les **PosA**.

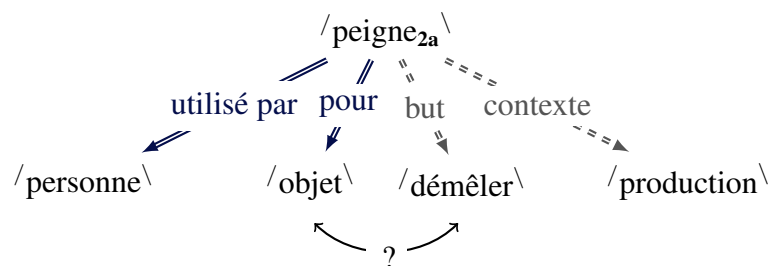


FIGURE 3.14 – Représentation obtenue à l'aide de l'éditeur

Chapitre 4

Conclusion

4.1 Bilan

Lors de ce stage, nous avons pu amorcer un premier pas vers une formalisation plus importante des dictionnaires s'appuyant sur la **TST**. Afin de vous présenter le travail qui a été produit, nous avons exposé un bref aperçu de l'état actuel de l'édition d'entrée dans un dictionnaire s'appuyant sur la **TST** par le biais du projet **RELIEF**. Puis, nous nous sommes penchés sur la scénarisation d'interactions que des lexicographes seraient amenés à effectuer avec le formalisme des **GU**, un formalisme de représentation des connaissances établi au cours de la thèse de Maxime LEFRANÇOIS. Nous soulignerons le fait que les scénarios d'interactions que nous avons proposés ont fait l'objet d'une publication (Lefrançois *et al.*, 2013). Nous avons ensuite proposé une première maquette d'un éditeur de définition utilisant toutes les composantes de ce formalisme. À partir de cette maquette, nous avons conçu un prototype vertical que nous avons présenté lors de la *3e journée RELIEF*. Ce prototype a été créé dans le but d'une évaluation, que nous avons pu mener auprès de cinq lexicographes du projet **RELIEF**, visant à estimer la viabilité du modèle et de l'interface. Les résultats que nous avons obtenus suite à l'évaluation du prototype nous ont permis de mettre en exergue le fait que les lexicographes ne sont pas préparés à ce formalisme et que nous devons nous rapprocher de leur méthodologie actuelle, quitte à mettre de côté une partie du formalisme des **GU**. Nous avons donc proposé un nouveau scénario ainsi qu'une nouvelle maquette s'appuyant sur l'hypothèse d'inférer la **RSemP** à partir de la **RSemS**.

Enfin, le formalisme semble encore trop jeune pour être complètement accepté au sein de projet tel que **RELIEF**, qui développe une méthodologie de travail depuis plus de dix ans. Nous n'avons pas pu atteindre notre objectif de faire accepter les **GU** par les lexicographes. En revanche, nous avons ouvert des voies de recherches et éclairci les éléments pouvant être intégrés à l'édition lexicographique.

Pour conclure, la conception d'interface homme-machine pour des experts dans un domaine se révèle une tâche complexe imposant une acquisition des connaissances importante. Cepen-

dant, la difficulté que nous avons rencontrée dans ce projet ne se limite pas à l'acquisition des connaissances d'un domaine précis, l'objectif n'est pas d'informatiser une tâche connue par des experts mais de transformer leurs méthodes en les formant à de nouveaux concepts. Introduire des éléments encore inconnus aux utilisateurs se montre extrêmement délicat, il est important de trouver un juste milieu entre les interactions avec les nouvelles notions et l'ancienne méthodologie. Afin de trouver ce juste milieu, il est important d'interagir avec l'utilisateur un maximum par le biais d'entretiens, d'évaluations, etc. Si nous étions amenés à effectuer une tâche identique, notre approche serait différente préférant une démarche moins intrusive dans le domaine et une communication beaucoup plus importante avec les experts.

4.2 Perspectives

4.2.1 Dans la continuité de ce stage

Dans la continuité de ce stage, il est encore nécessaire de soumettre à une évaluation la dernière maquette proposée. Dans le cas où cette évaluation se révèle être un succès, nous pourrions essayer d'ajouter un peu plus d'éléments permettant d'introduire la notion de sémantique profonde auprès des lexicographes. Pour le moment, ne pouvant intégrer notre outil au logiciel *MVS Dicet*, nous devons développer les scénarios des utilisateurs autres que le lexicographe (Ingénieur de la connaissance, Administrateur) et concevoir une maquette leur présentant les données. De plus, dans l'objectif de faire de cette interface un véritable outil, nous devons encore approfondir les informations qu'il est possible d'extraire à partir de cette interface et sous quel format les stocker.

Une autre question que nous n'avons pas abordée lors de ce stage est : quelle représentation utiliser pour l'affichage des informations contenues dans un dictionnaire édité à l'aide des **GU** ? La création de dictionnaire s'effectue pour deux types d'utilisation, l'une est dans un but de traitement automatique, la seconde et probablement la plus évidente vise un public humain. Il est important de se pencher sur cette question, puisqu'une visualisation et exploration adéquate d'un dictionnaire rendraient compte de la force du formalisme des **GU** et permettraient une acceptation plus facile de la part des lexicographes de ce dernier.

4.2.2 Autres approches

La collecte des informations lexicales est une tâche difficile. En effet, effectuée manuellement, elle nécessite des compétences d'experts. Or, comme nous avons pu le voir lors de l'évaluation de notre prototype, les lexicographes ne sont pas encore prêts à manipuler les entrées d'un dictionnaire au niveau de la **RSemP**. Dans cette section, nous discuterons de l'idée de nouvelles approches afin d'apporter de la matière et de persuader ces derniers d'atteindre un tel niveau de profondeur dans leur définition.

Une extraction des données du RLF

Une extraction des données entrées dans le **RLF** accompagnée d'un traitement automatique permettrait de construire une première base de connaissance conçue selon le formalisme des **GU**. Notre idée est de s'appuyer principalement sur le réseau que forme les fonctions lexicales pour en inférer une première base. Cette base serait probablement inconsistante et nécessiterait une édition manuelle. Évidemment, une telle approche exige encore plusieurs années de recherche puisque la base **RLF** n'est pas encore entièrement construite, les fonctions lexicales se montrent complexes et nombreuses et les éléments nécessaires et suffisants assurant la possibilité de raisonner sur les **GU** ne sont pas encore établis.

Un système collaboratif

Une autre démarche peut être orientée vers système collaboratif ouvert à tous (tel Wikipédia) où ce sont des utilisateurs non experts qui définissent les entrées. Une telle démarche a déjà été effectuée, nous pouvons citer par exemple JeuxDeMots¹ (Lafourcade et Joubert, 2008). JeuxDeMots est un projet contributif proposant une application ludique accessible sur le web. Ce sont les joueurs qui vont alimenter la base, en fournissant des associations par le biais de réponses à divers énoncés. Le principal avantage d'une telle approche est le fait qu'elle ne nécessite aucun expert et que sous la condition d'une importante communauté, la base se remplit rapidement. En revanche, assurer la fiabilité et la qualité des informations recueillies risque de se révéler très complexe.

4.2.3 Bases de connaissances en Graphes d'Unités : Applications

Précédemment, nous avons vu d'autres méthodes pouvant être utilisées à moyen terme afin d'obtenir une première base de connaissance. Dans cette section nous présentons quelques applications pouvant être développées à l'aide d'une telle base.

Dans ce rapport, nous avons présenté uniquement la partie sémantique des **GU**. Mais ce formalisme permet de représenter tous les niveaux de la langue, c'est pourquoi il est possible de créer un système de traitement automatique de la langue basé uniquement sur ce formalisme.

Le projet **RELIEF** axe la majorité de son activité autour de la Langue Française, mais deux autres projets s'appuyant sur la même méthodologie et le même logiciel ont été lancés dans le but de créer deux nouvelles ressources lexicales : un Réseau Lexical de l'Espagnol (RLE), un Réseau Lexical du Coréen (RLC). Dans une même perspective multilingue, il serait possible d'établir une correspondance entre les unités sémantiques d'un dictionnaire au format des **GU** et de créer une application de traduction automatique s'appuyant sur le sens.

1. <http://www.jeuxdemots.org/>

Bibliographie

- BARQUE, L., NASR, A. et POLGUÈRE, A. (2010). From the Definitions of the 'Trésor de la Langue Française' To a Semantic Database of the French Language. *In Proceedings of the XIV Euralex International Congress*, Fryske Akademy, page 245–252, Leeuwarden (Pays-Bas). Anne Dykstra et Tanneke Schoonheim, dir.
- BARQUE, L. et POLGUÈRE, A. (2008). Enrichissement formel des définitions du Trésor de la Langue Française informatisé (TLFi) dans une perspective lexicographique. 22.
- KAHANE, S. (2003). The meaning-text theory. *Dependency and Valency. An International Handbook on Contemporary Research*. Berlin : De Gruyter.
- LAFOURCADE, M. et JOUBERT, A. (2008). JeuxDeMots : un prototype ludique pour l'émergence de relations entre termes. *In JADT'08 : Journées internationales d'Analyse statistiques des Données Textuelles*, pages 657–666.
- LEFRANÇOIS, M. et GANDON, F. (2013). Rationale, concepts, and current outcome of the unit graphs framework. *In Proc. of the 9th International Conference on Recent Advances in Natural Language Processing (RANLP)*, Hissar, Bulgaria.
- LEFRANÇOIS, M., GUGERT, R., GANDON, F. et GIBOIN, A. (2013). Application of the unit graphs framework to lexicographic definitions in the RELIEF project. *In Proc. of the 6th International Conference on Meaning-Text Theory (MTT 2013)*, Prague, Czech Republic.
- LUX-POGODALLA, V. et POLGUÈRE, A. (2011). Construction of a French Lexical Network : Methodological Issues. *In Proceedings of the International Workshop on Lexical Resources*, Ljubljana.
- MEL'ČUK, I. (1984–1999). *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques*, volume I–IV. Les Presses de l'Université de Montréal, Montréal.
- MEL'ČUK, I. A. (1996). Lexical functions : a tool for the description of lexical relations in a lexicon. *Lexical functions in lexicography and natural language processing*, 31:37–102.
- MEL'ČUK, I. A. et ARBATCHEWSKY-JUMARIE, N. (1999). *Dictionnaire explicatif et combinatoire du français contemporain : recherches lexico-sémantiques*, volume 4. PU Montréal.
- POLGUÈRE, A. (2011). Classification sémantique des lexies fondée sur le paraphrasage. *Cahiers de lexicologie*, 98:197–211.
- POLGUÈRE, A. (2000). Une base de données lexicales du français et ses applications possibles en didactique. *Revue de Linguistique et de Didactique des Langues (LIDIL)*, 21:75–97.

- POLGUÈRE, A. (2009). Lexical systems : graph models of natural language lexicons. *Language resources and evaluation*, 43(1):41–55.
- ROBICHAUD, B. (2011). A graph visualization tool for terminology discovery and assessment. *In Proceedings of the 5th International Conference on the Meaning Text Theory*, page 243–1252, Barcelona, Spain.
- ŽOLKOVSKIJ, A. et MEL'ČUK, I. A. (1965). O vozmožnom metode i instrumentax semantičeskogo sinteza. 5:23–28.

Annexe A

Documents de conception

A.1 Plan et notes de l'entretien avec Alain Polguère - 17/04/2013

L'entretien s'est déroulé de façon informelle. Il a été conduit de manière semi-directive, à questions ouvertes, et permet de décrire la situation actuelle du travail de définition des lexicographes et comment ces derniers s'imaginent interagir avec des graphes. Le plan que nous fournissons sert de fil conducteur, certaines questions sont amenées à n'être que partiellement traitées.

Plan

1. Expliquer le but de notre projet et l'idée actuelle de manipulations de graphes.
2. Lui proposer un scénario dans lequel il doit définir une nouvelle entrée.
3. Lui montrer les graphes actuels (graphes hiérarchiques). Quelles pourraient être ses attentes ?
4. Qui pourrait être amené à utiliser les définitions sous forme de graphes ?
5. Connaît-il des logiciels de manipulations des graphes ?
6. Pourrait-il nous donner des pointeurs vers un ou deux lexicographes de la Théorie Sens-Texte ?
7. Serait-il possible de le revoir pour d'autres entretiens ?
8. (BONUS) Questions concernant MVS Dicet.

Notes

Le réseau sémantique est très riche, si on l'affiche complètement il se montrera illisible. L'idée de le structurer en sous graphe à l'aide d'un système de boîte pouvant être fermées ou ouverte est émise.

L'édition de définition se déroule en trois étapes :

1. Choix de l'étiquette sémantique dans hiérarchie créée à l'aide de protégé par Alain Polguère.
2. Trouver le nombre d'actants, écrire la forme propositionnelle. Alain Polguère a dans l'idée d'implémenter des patrons de formes propositionnelles qui seraient présentés dans une liste ordonnée selon le nombre d'actants.
3. Paraphrase au format XML selon le formalisme du projet Definiens.

La notion de hiérarchie existe déjà, en revanche l'idée d'associer des patrons de structures actanciennes aux étiquettes sémantiques semble intéressante. Concernant les graphes, la représentation de ces derniers se fait dans leur tête. L'idée de présenter les définitions sous forme de graphe lui semble intéressante pour de la validation et éviter les incohérences existantes dans le DEC. Il a déjà utilisé yED (http://www.yworks.com/en/products_yed_about.html) afin de manipuler des graphes.

Actuellement, les lexicographes du projet RELIEF commencent à apprendre la méthodologie que propose Alain Polguère. Ce dernier nous conseille de le contacter ou de contacter Lucie Barque.

Il y a une possibilité d'intégration d'une application au sein de MVS Dicet, le logiciel a été étudié pour. Concernant MVS Dicet, il est encore en développement. Pour le moment seules les étiquettes sémantiques sont implémentées, la définition sera ajoutée prochainement.

A.2 Evaluation du Prototype - 14/06/2013

Le modèle du processus d'édition à base de Graphes d'Unités

Au sujet du modèle, la notion de sémantique profonde n'est vraiment pas facile à appréhender pour les lexicographes. Ne possédant encore aucune méthodologie afin de déterminer cette dernière, ils se retrouvent perdus et cherchent à retourner vers les notions qu'ils connaissent. Pour illustrer nos propos nous pouvons mettre en avant que beaucoup d'entre eux ont fait l'amalgame entre les positions actanciennes de la sémantique profonde et les actants profonds de la syntaxe. En revanche, la notion d'héritage semble compréhensible, les lexicographes utilisent une fonction lexicale ("gener") représentant cet héritage.

Concernant l'ordre dans le processus, les lexicographes ont pour habitude de définir le nombre d'actants présents en premier. Le choix du nombre d'actants doit être effectué très tôt dans le processus puisque leur processus de réflexion se base principalement sur ce point de départ. Choisir le parent, avant de déterminer le nombre d'actants, est une option acceptable. De plus l'héritage de la structure actancielle du parent leur semble tout à fait logique puisque actuellement il s'appuie sur le "gener" pour construire la structure actancielle.

Enfin, à propos des croquis demandés, la plupart ont dessiné un arbre syntaxique, d'autre ont préféré s'abstenir de dessiner étant déroutés suite à l'explication du processus.

L'interface

Concernant la représentation des positions actanciennes, un modèle tabulaire ne semble pas adapté. Les lexicographes préféreraient un graphe sémantique. Ce graphe devrait être ordonné comme leur graphe sémantique actuel ou représenté sous forme textuelle (type forme propositionnelle).

De plus, le fait de ne pas pouvoir valider sans explication la modification d'une position actancielle est un réel manque.

Enfin le terme "Modifier" sur la fenêtre d'édition n'est pas ce qu'il convient, ils souhaiteraient plutôt "Valider".

Concernant la manipulation du graphe de définition, encore une fois le besoin d'un graphe représentant la sémantique de surface est important. Beaucoup d'évaluateurs se sont retrouvés troublés face à cet outil. Sortir les éléments de la boîte représentant la structure actancielle de l'unité sémantique ne semble pas adapté. Un système organisant automatiquement et sortant chacune des positions actanciennes semble préférable. Le bouton de suppression d'un noeud n'a

pas été utilisé, les évaluateurs ont préféré utiliser la fonction de retour en arrière. L'ajout d'un élément dans le graphe afin de lier deux signatures n'a pas été effectué, cela est sûrement dû au fait qu'ils ne connaissent pas le formalisme et cette action n'est donc pas connue. Le bouton d'ajout d'un élément a été trouvé rapidement mais le glissé déposé pour l'ajout de l'élément semble déconcertant.

Synthèse

La nouvelle notion introduite par les graphes d'unités (la sémantique profonde) semble pour le moment difficile à assimiler pour les lexicographes. De plus ils ne voient pas l'intérêt d'atteindre un niveau si profond dans leur dictionnaire.

Suite à une discussion avec Alain Polguère, l'idée d'utiliser les données du projet RELIEF afin de générer automatiquement les graphes d'unités a été évoquée. RELIEF et les graphes d'unités étant tous deux en cours de développement, cette idée demande encore quelques temps de recherche avant d'être exploitée.

Les lexicographes ne souhaitant pas manipuler ce formalisme directement, nous avons émis l'hypothèse qu'il serait possible d'extraire la sémantique profonde à partir des divers éléments constitutifs de la sémantique de surface. Nous nous tournons alors vers un nouveau modèle beaucoup plus proche de la méthodologie actuelle du projet RELIEF, quitte à perdre une partie des informations dont les graphes d'unités ont besoin.

Annexe B

Maquettes

B.1 Hiérarchie des types d'unités

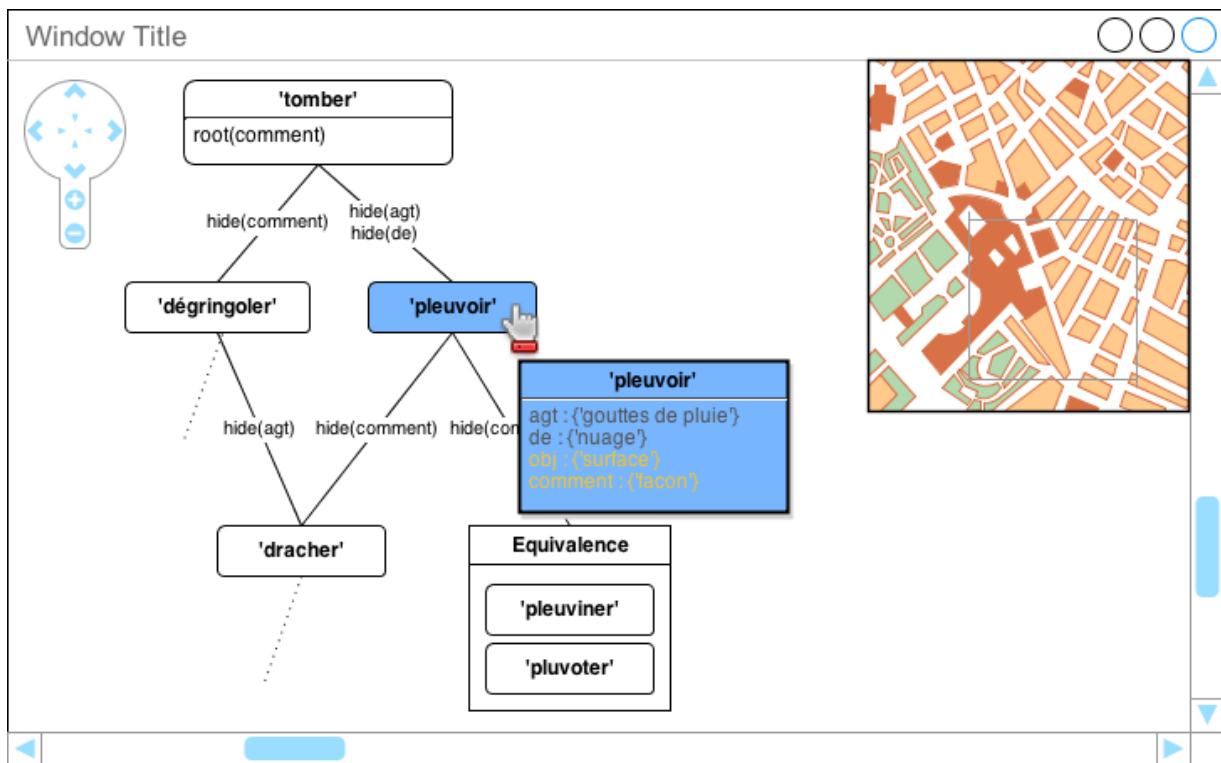


FIGURE B.1 – Maquette de la hierarchie des types d'unités sans détail

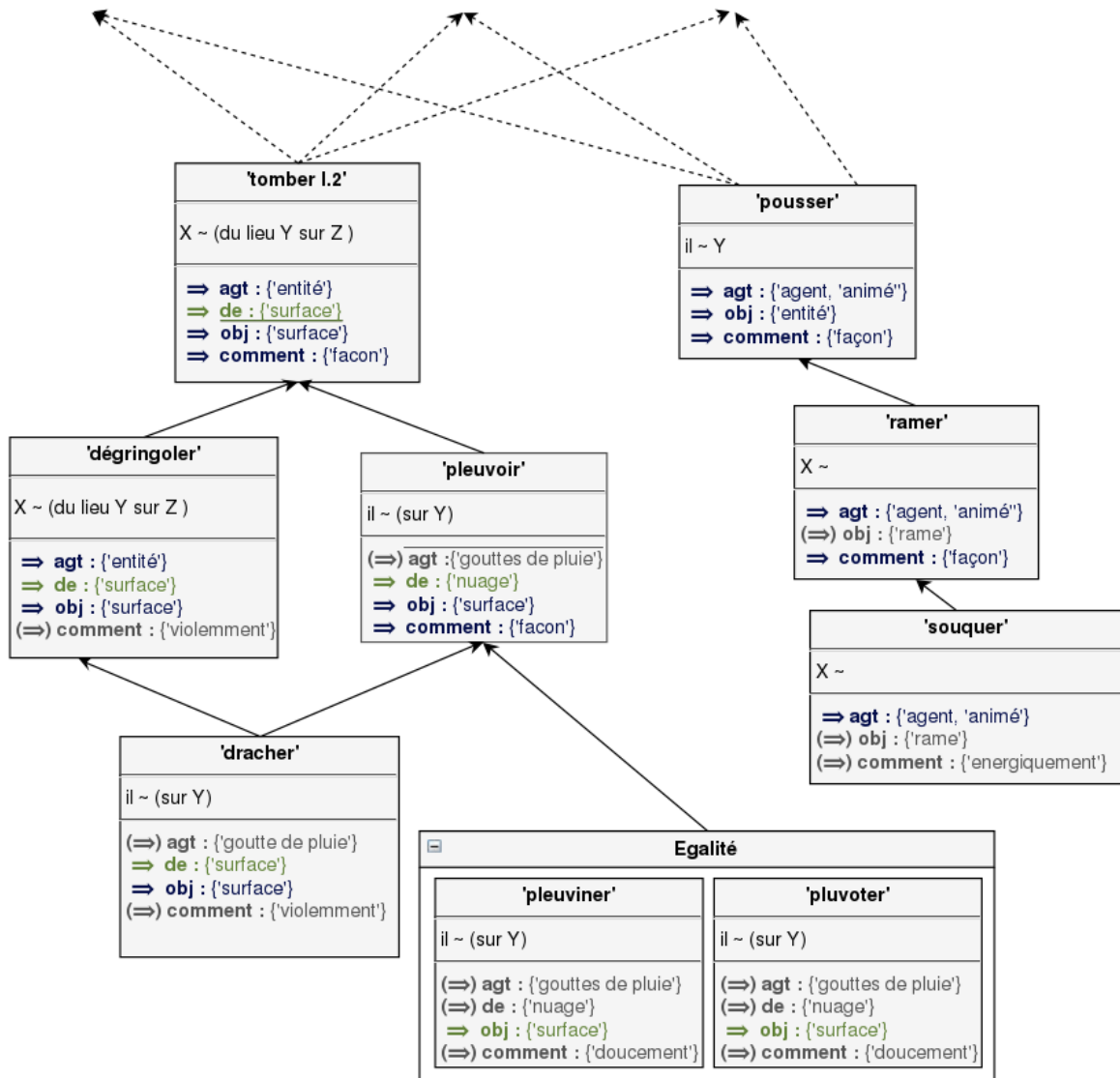


FIGURE B.2 – Maquette finale de la hiérarchie des types d'unités

B.2 Editeur

Ajouter DSemUT

Hierarchie des DSemUTs

- /Fact\
 - ▼ /entity\[(⇒)1]
 - ▼ /physical object\[(⇒)1]
 - ▼ /artefact\[(⇒)1]
 - ▼ /instrument\[(⇒)1]
 - /tool\[(⇒)1]

Information

La création d'un nouveau DSemUTs est soumise à la validation d'un administrateur. Vous devrez utiliser le parent le plus proche, le temps que ce dernier soit validé.

Nouveau DSemUT

Parent :

Nom :

Commentaire :

Search

Annuler **Valider**

FIGURE B.3 – Ajout d'une unité à la structure actancielle de la sémantique profonde

Peigne2a, Parents

Hierarchie des DSemUTs

- /Fact\
 - ▼ /entity\[(⇒)1]
 - ▼ /physical object\[(⇒)1]
 - ▼ /artefact\[(⇒)1]
 - ▼ /instrument\[(⇒)1]
 - /tool\[(⇒)1]

Information

/tool

- ⇒ possessor : /person\
- ⇒ activity : /activity\
- (⇒) profession : /profession\

Add

Nearest Parents ⚠

- /fact1\[...] x
- /fact5\[...] x
- ▼ /instrument\[(⇒)1] x
 - /tool\[(⇒)1] x
- ▼ /parent\[(⇒)1]
 - /DSemUT créé\[(⇒)1] x

Search

DSemUTs non trouvé **Annuler** **Valider**

FIGURE B.4 – Choix du parent hiérarchique le plus proche dans la sémantique profonde

Window Title				
Fichier	Edition	Options	Outils	Aide
Peigne 2a				
Structure actancielle de la SP	<div style="border: 1px solid gray; padding: 5px;"> <code>/peigne2a\ < /tool\</code> </div>			Informations
Définition de la SP	<div style="border: 1px solid gray; padding: 5px;"> Outil de tissage qu'une personne X utilise pour démêler les fibres de l'objet Y </div>			Notifications Mémo
Structure actancielle de la SS	<div style="border: 1px solid gray; padding: 5px;"> \Rightarrow possessor : <code>/personne\</code> \Rightarrow activity : <code>/untangle\</code> \Rightarrow profession : <code>/weaver\</code> (\Rightarrow) object : ? <div style="margin-top: 10px;"> <div style="border: 1px solid gray; padding: 2px 5px;">+</div> New actant slot </div> </div>			Par toto le 12/05/2052 : Un commentaire de toto Par tutu le 12/05/2052 : Oui oui
Correspondance entre SP et SS	<div style="border: 1px solid gray; padding: 5px;"> <div style="display: flex; justify-content: space-around; align-items: center;"> <div>Sign</div> <div>Sign</div> <div>Sign</div> <div>Type</div> <div>Sign</div> <div>Type</div> <div style="background-color: red; color: white; padding: 2px 5px;">X</div> </div> </div>			

FIGURE B.5 – Maquette de l'édition de la structure actancielle de la sémantique profonde

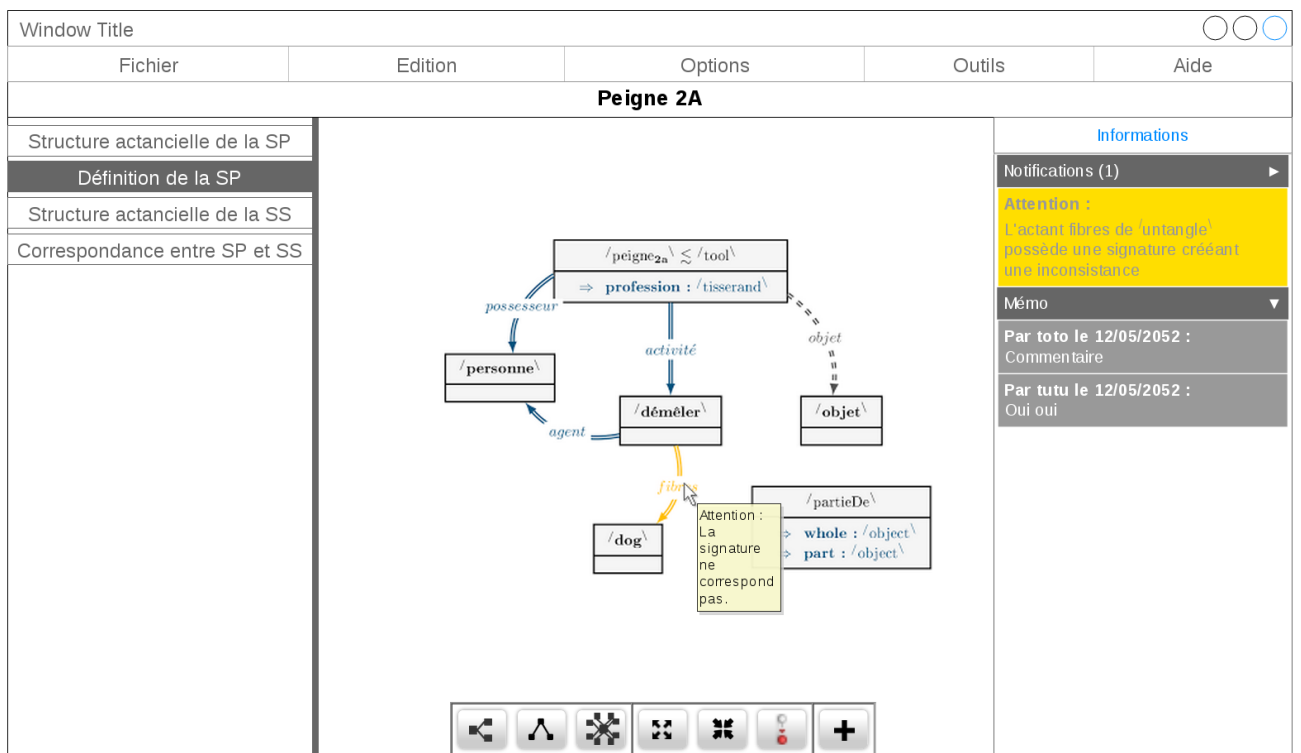


FIGURE B.6 – Maquette de l'édition de la définition de la sémantique profonde

Window Title				
Fichier	Edition	Options	Outils	Aide
Peigne 2a				
Structure actancielle de la SP	(peigne_{2a})			Informations
Définition de la SP	Outil de tissage qu'une personne X utilise pour démêler les fibres de l'objet Y			Notifications
Structure actancielle de la SS	Nombre d'actants : 2 + -			Mémo
Correspondance entre SP et SS	<div>⇒ Actant 1 X</div> <div>⇒ Actant 2 Y</div>			Par toto le 12/05/2052 : Un commentaire de toto Par tutu le 12/05/2052 : Oui oui

FIGURE B.7 – Maquette d'édition de la structure actancielle de surface

Window Title				
Fichier	Edition	Options	Outils	Aide
Peigne 2a				
Structure actancielle de la SP	Sémantique profonde			Sémantique de surface
Définition de la SP	<div>⇒ possesseur : /personne\</div> <div>⇒ activité : /démêler\</div> <div>⇒ profession : /tisserand\</div> <div>⇒ objet : /objet\</div>			<div>⇒ Actant 1 X</div> <div>⇒ Actant 2 Y</div>
Structure actancielle de la SS	<div>⇒ possesseur : /personne\</div> <div>⇒ activité : /démêler\</div> <div>⇒ profession : /tisserand\</div> <div>⇒ objet : /objet\</div>			<div>⇒ Actant 1 X</div> <div>⇒ Actant 2 Y</div>
Correspondance entre SP et SS	<div>⇒ possesseur : /personne\</div> <div>⇒ activité : /démêler\</div> <div>⇒ profession : /tisserand\</div> <div>⇒ objet : /objet\</div>			<div>⇒ Actant 1 X</div> <div>⇒ Actant 2 Y</div>

FIGURE B.8 – Maquette de l'édition des correspondance entre les deux sémantiques.

Annexe C

Prototype

C.1 Structure actancielle de la sémantique profonde

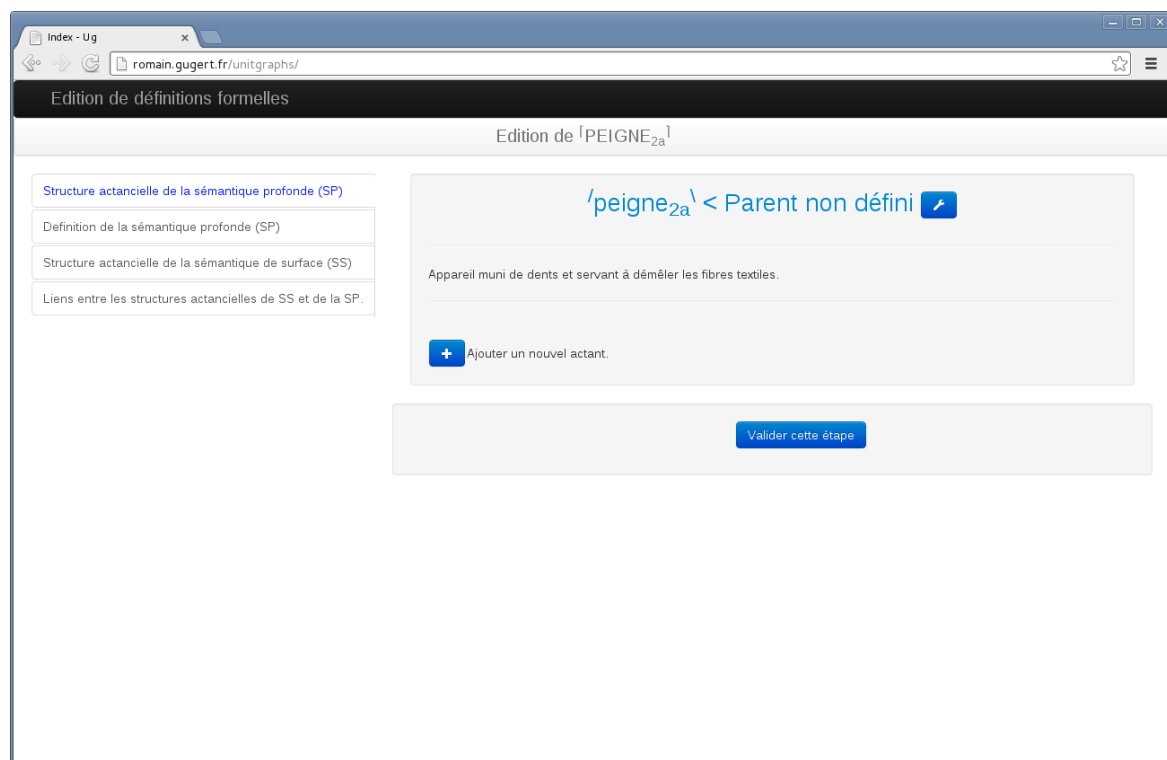


FIGURE C.1 – Point de départ de l'édition de la structure actancielle

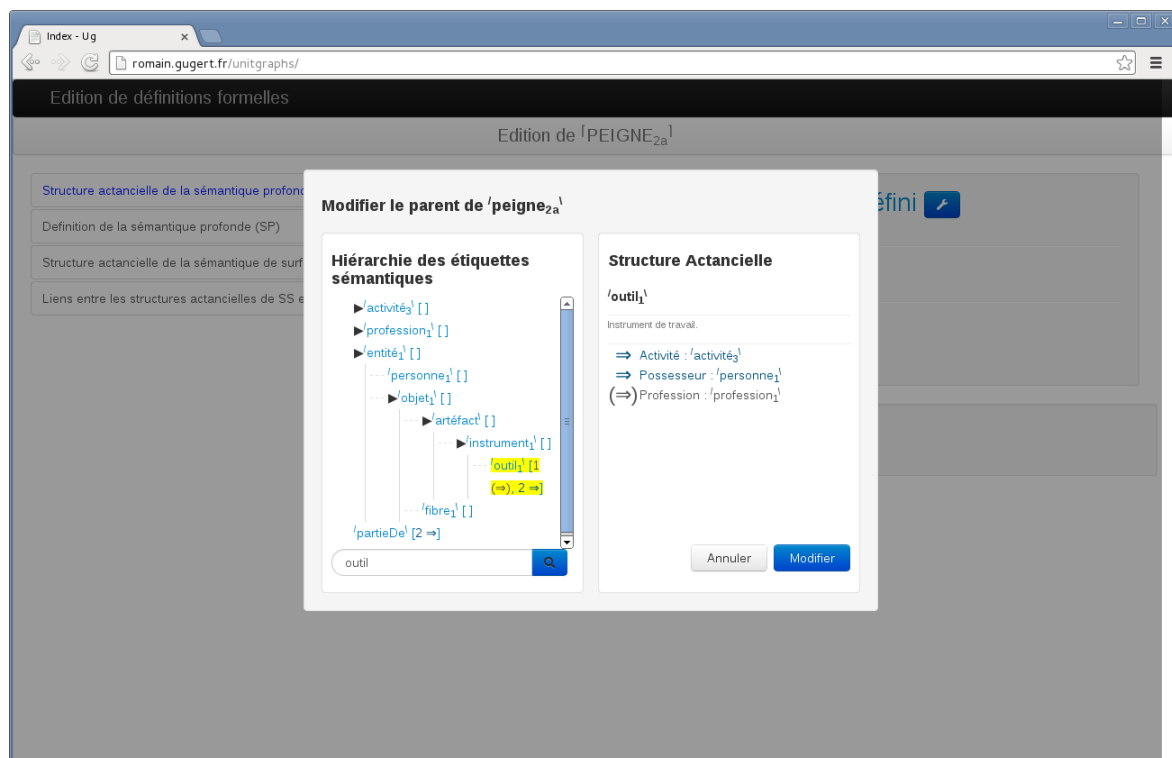


FIGURE C.2 – Sélection du parent hiérarchique

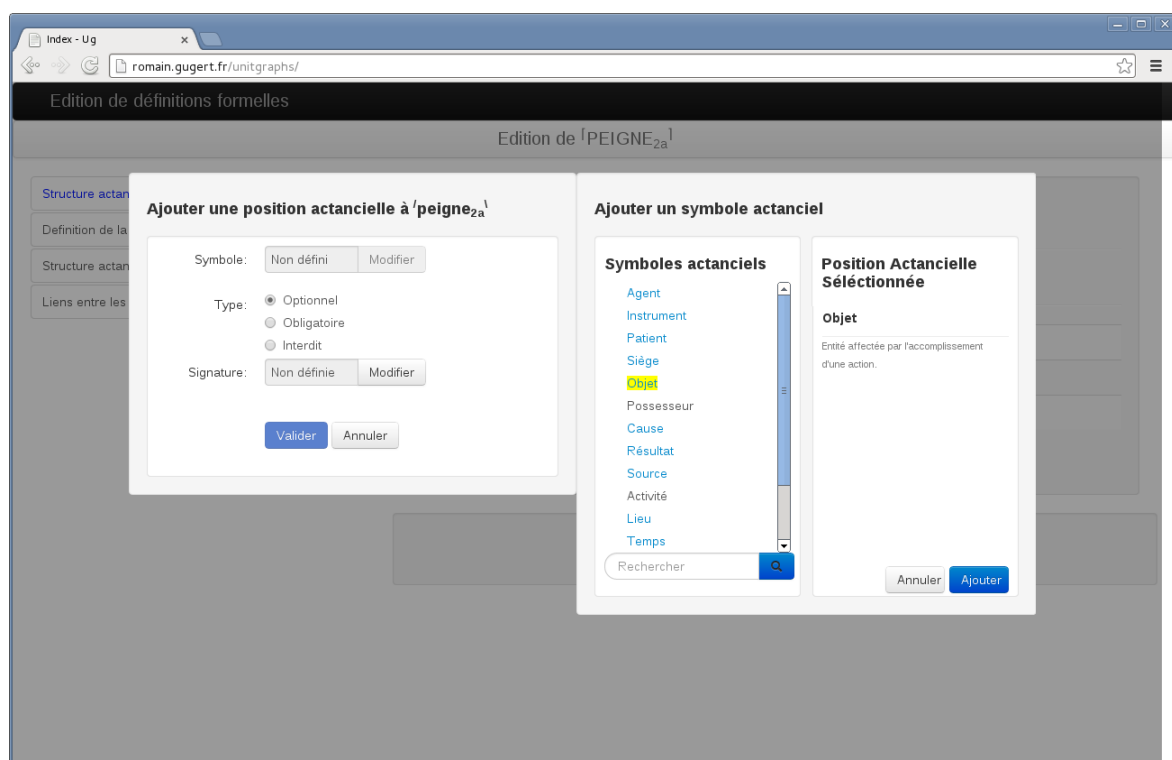


FIGURE C.3 – Ajout d'une position actancielle

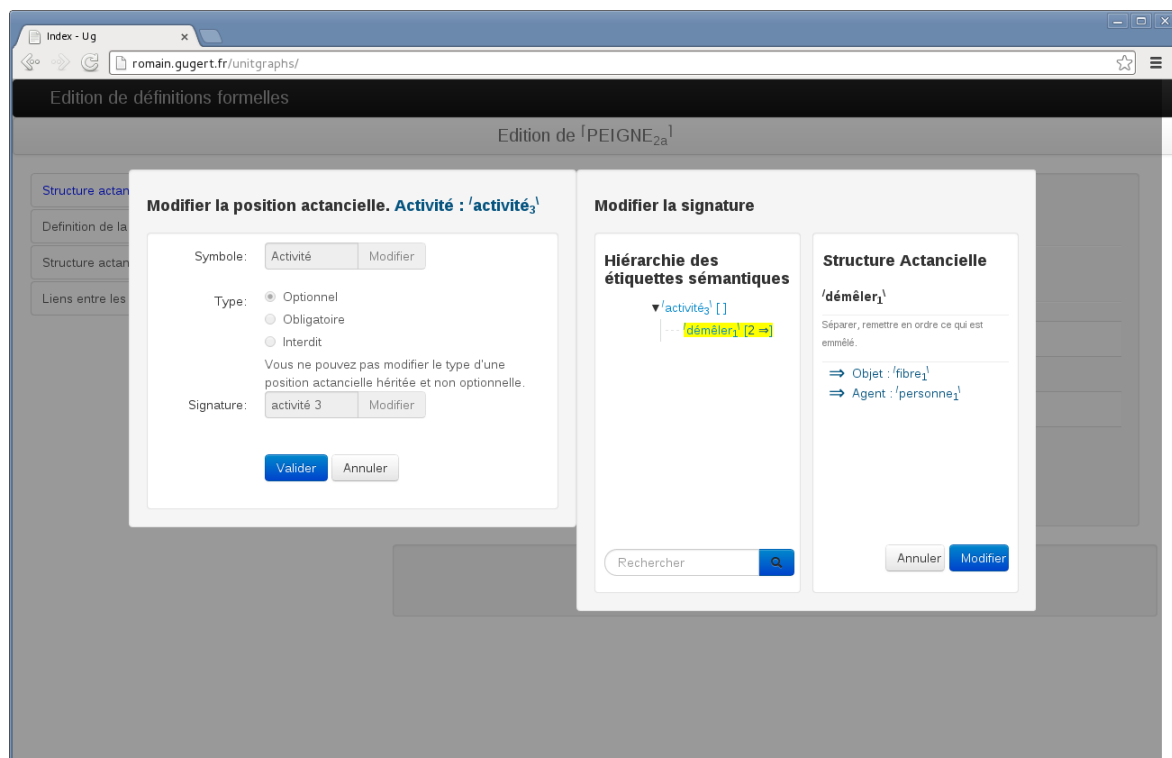


FIGURE C.4 – Edition d’une position actancielle

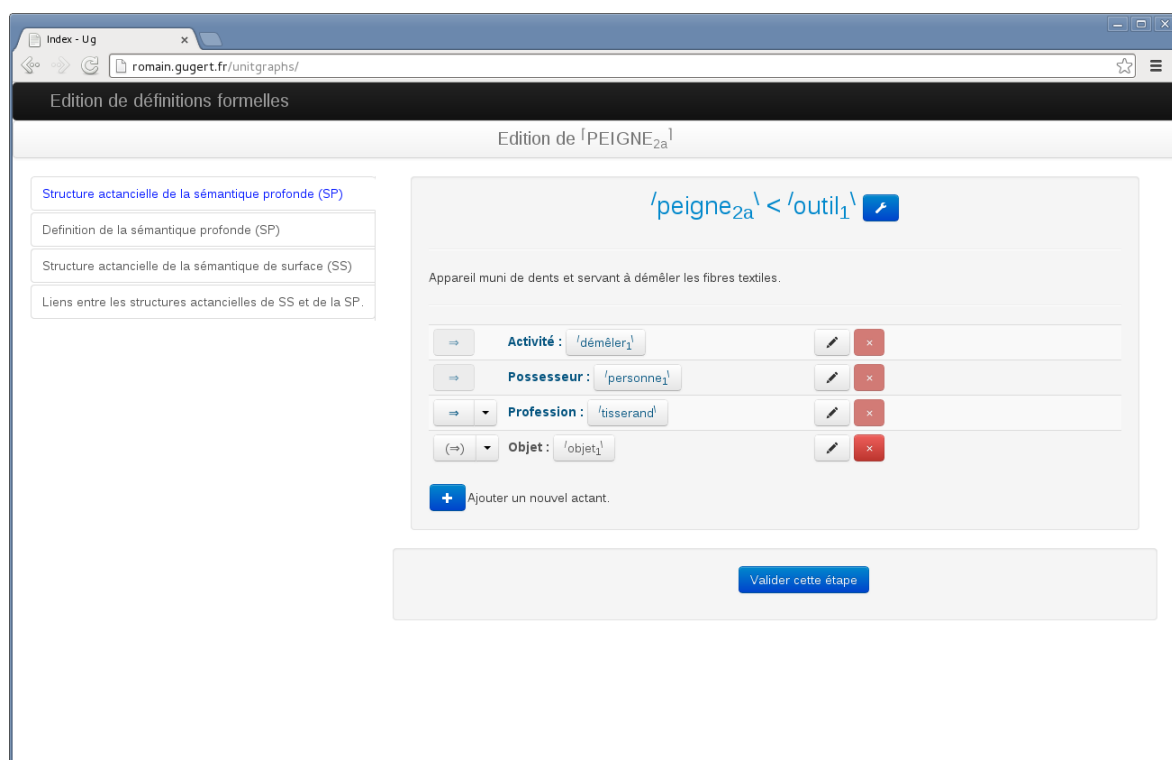


FIGURE C.5 – Structure actancielle finale de peigne2a



FIGURE C.6 – Infos bulles contenues dans l'éditeur de la structure actancielle

C.2 Définition de la sémantique profonde

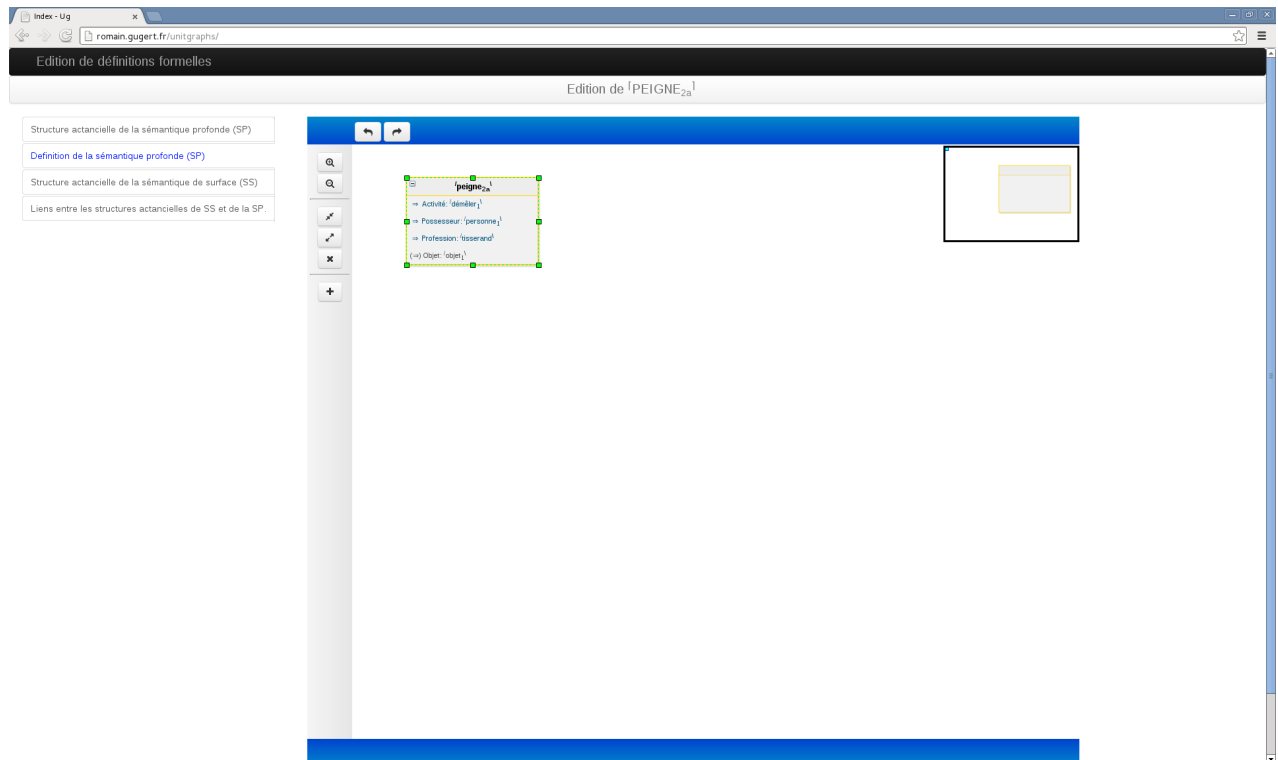


FIGURE C.7 – Point de départ de l’édition de la définition

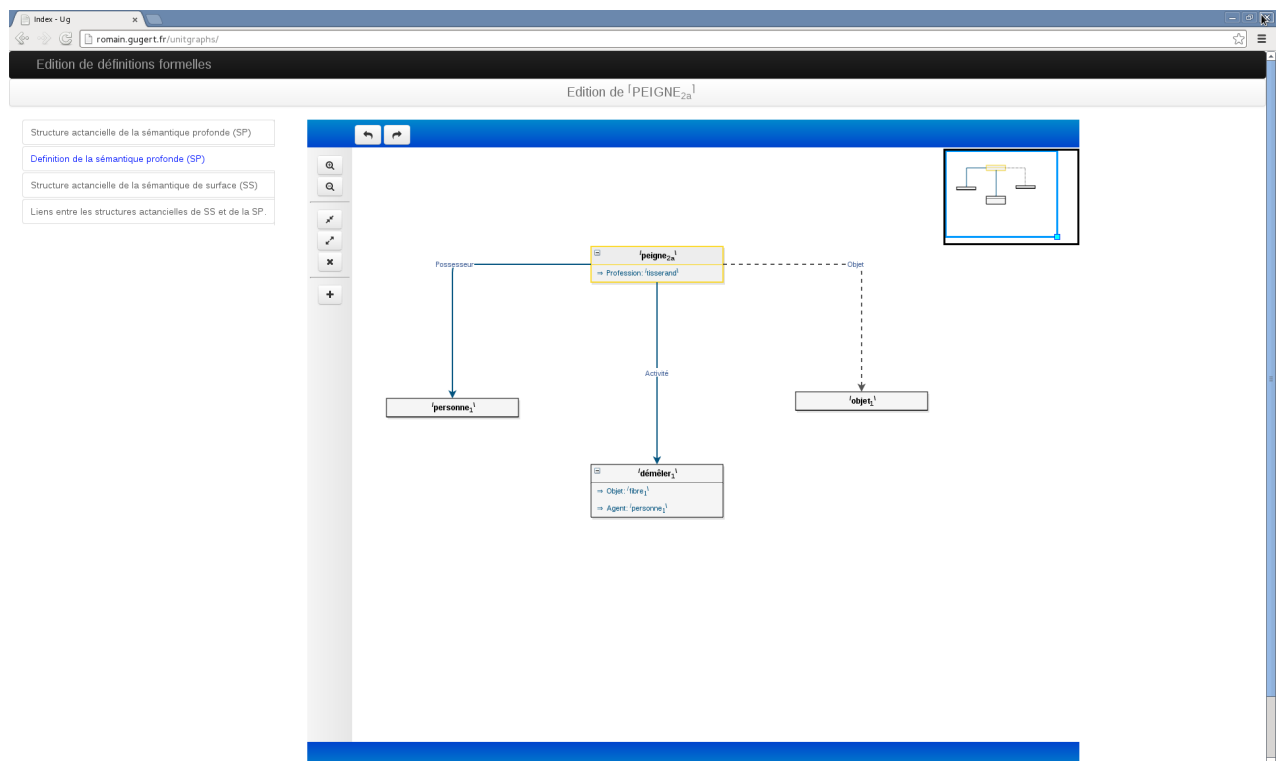


FIGURE C.8 – Sortir les postions actancielle intéressantes de la définition

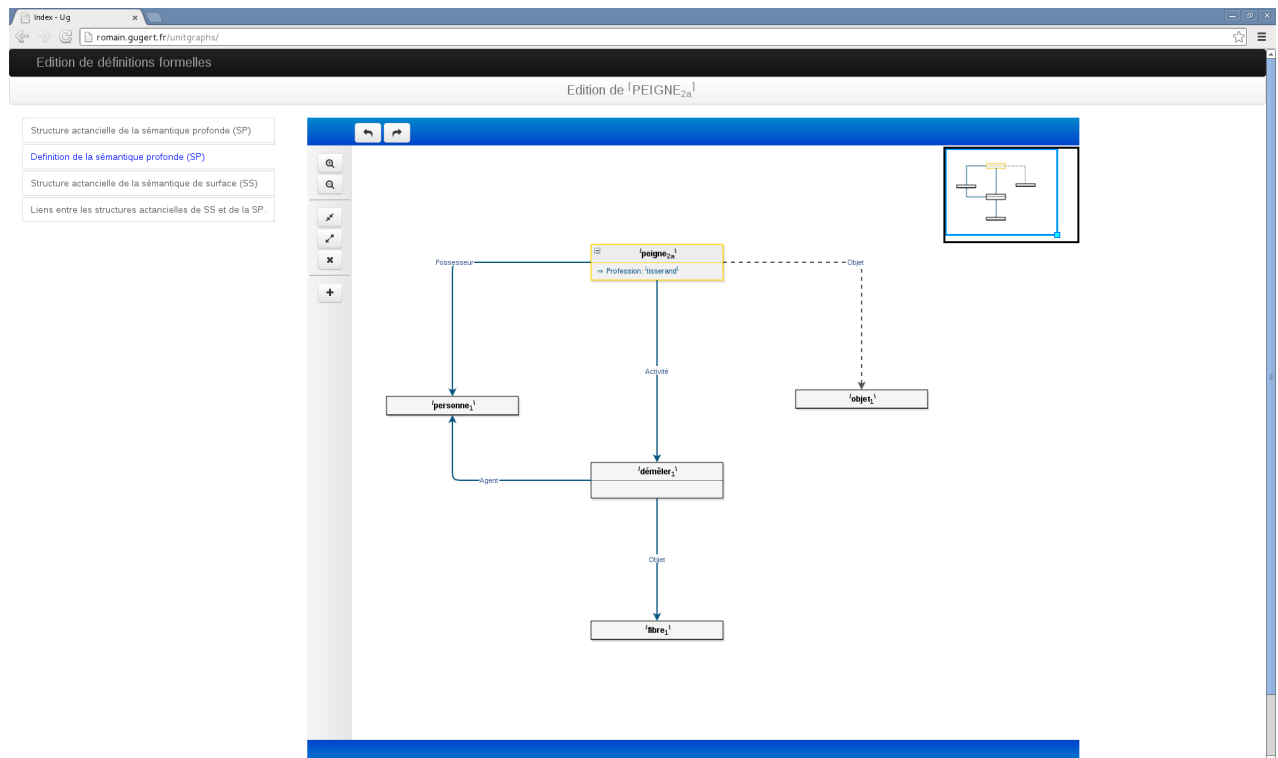


FIGURE C.9 – Fusion des positions actanciels

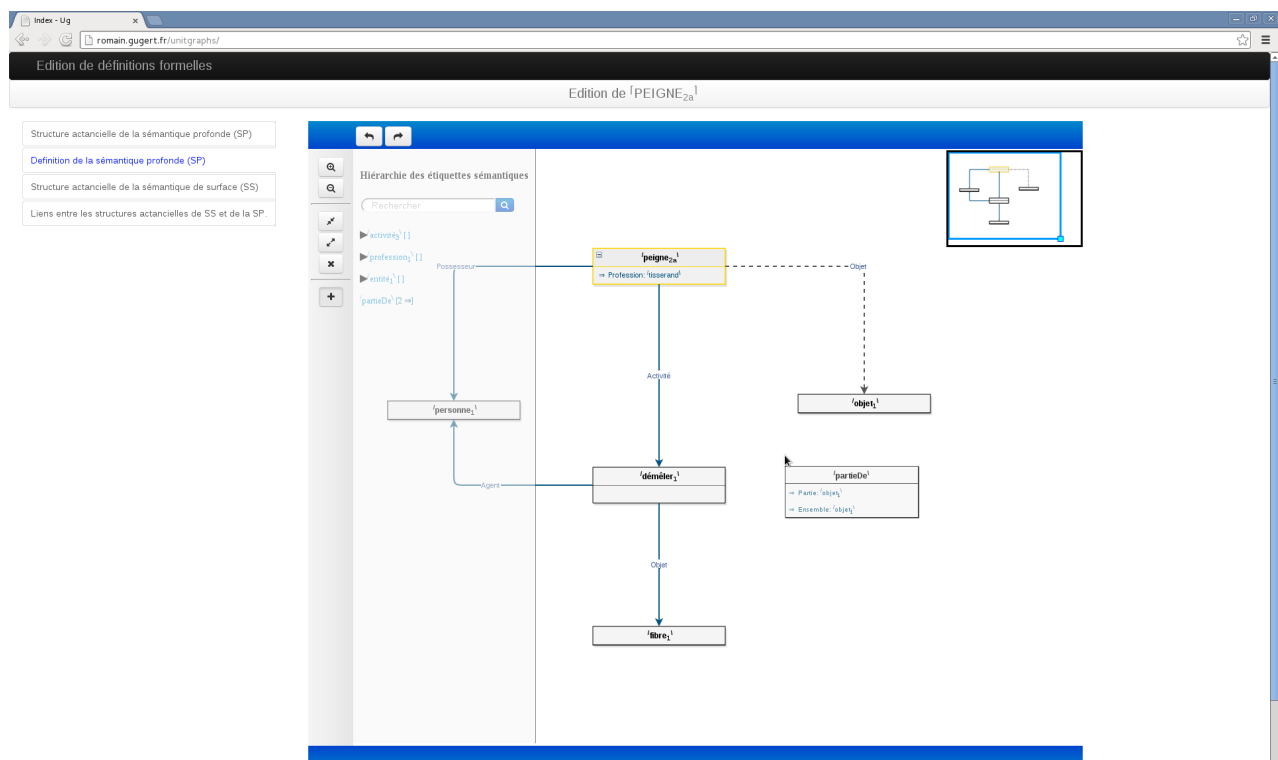


FIGURE C.10 – Ajout d'un noeud dans la définition

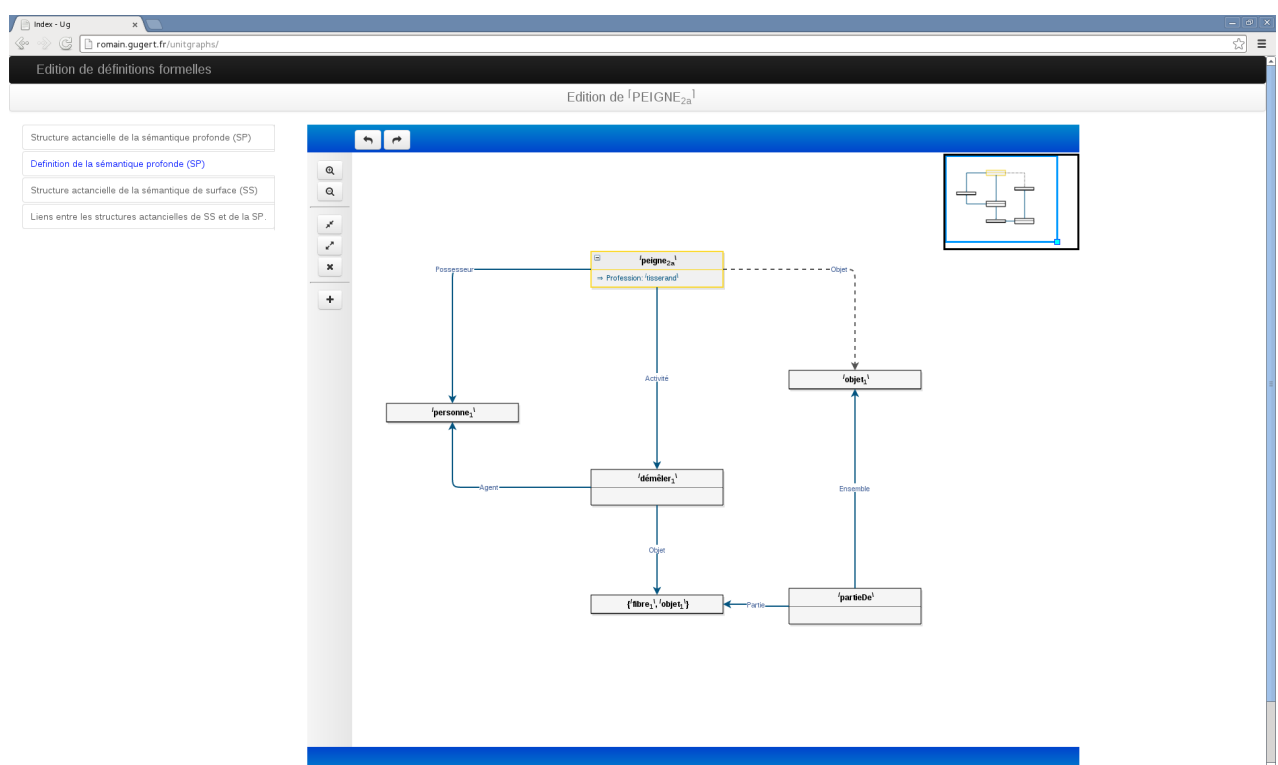


FIGURE C.11 – Structure finale de la définition

C.3 Structure actancielle de la sémantique de surface

Index - Ug
romain.gugert.fr/unitgraphs/

Edition de définitions formelles

Edition de 'PEIGNE_{2a}'

Structure actancielle de la sémantique profonde (SP)

Définition de la sémantique profonde (SP)

Structure actancielle de la sémantique de surface (SS)

Liens entre les structures actancielle de SS et de la SP.

(peigne_{2a})

Appareil muni de dents et servant à démêler les fibres textiles.

Nombre d'actants: 2 + -

Actant 1	X
Actant 2	Y

Valider cette étape

FIGURE C.12 – Interface d'édition de la structure actancielle de la sémantique profonde dans le prototype

C.4 Correspondances entre les deux niveaux de sémantique

Index - Ug
romain.gugert.fr/unitgraphs/

Edition de définitions formelles

Edition de 'PEIGNE_{2a}'

Sémantique profonde

(peigne_{2a})

- ⇒ Activité : /démêler₁
- ⇒ Possesseur : /personne₁
- ⇒ Profession : /tisserand
- (⇒) Objet : /objet₁

Sémantique de surface

(peigne_{2a})

- ⇒ Actant 1 X
- ⇒ Actant 2 Y

Valider cette étape

FIGURE C.13 – Interface d'édition des correspondances entre les deux niveau de sémantique dans le prototype