PROPOSITION DE NORME DES LEXIQUES POUR LE TRAITEMENT AUTOMATIQUE DU LANGAGE

Version-1.10 13 mai 2004

Gil Francopoulo
INRIA/LORIA-ACTION SYNTAXE

Préambule

Le présent travail est réalisé dans le cadre de l'action SYNTAXE de l'INRIA, du projet RNTL - Outilex, de l'action Normalangue et du groupe de travail AFNOR : « lexiques pour le TAL ». Il est en relation avec l'initiative américaine ISO-LMF (Lexical Markup Framework) ainsi qu'avec les travaux sur les catégories de données du TC37/SC4 de l'ISO.
Au sein de l'ISO, le travail est effectué dans le cadre du NWIP (New Work Item Proposal) LMF dont l'objectif est de définir la norme ISO-24613.

1- Introduction

Des ressources lexicales pour le traitement du langage (TAL) ont été créées, le plus souvent sans rapport avec le cadre des dictionnaires destinés à une édition imprimée. Il n'existe cependant pas de format standard pour les structures décrites et le choix des catégories de données varie d'une ressource à l'autre. La connaissance inscrite dans de telles ressources est à la fois extensive et onéreuse à maintenir. Le besoin de fusionner ces ressources est à la fois fréquent et onéreux. Une norme est nécessaire afin de permettre la création de telles ressources par des mécanismes de fusion ou par l'échange par simple interopérabilité.
Il s'agira ici de définir une norme de représentation des lexiques pour le traitement automatique du langage.

2- Périmètre de la spécification

L'objectif est de définir une norme de permettant la représentation des données lexicales pour en permettre la gestion générale et l'échange de dictionnaires.
Les données mentionnées ici sont exclusivement destinées aux applications du traitement automatique de la langue. Les dictionnaires éditoriaux, qui sont destinés à la lecture humaine, ne sont pas abordés car ils sont couverts par la norme ISO-1951, actuellement en cours de révision [Dérouin]. Parce qu'elles sont destinées à être exploitées par des programmes, les données lexicales nécessitent un travail de définition un peu plus précis que pour un dictionnaire éditorial. Le lexicographe d'un dictionnaire éditorial suppose que son lecteur aura assez de connaissances pour interpréter et étendre l'information trouvée dans le dictionnaire. Mais cette connaissance implicite fait défaut à l'utilisateur du lexique pour le TAL puisque celui-ci est un programme. Ainsi, chaque mot doit être précisément et complètement décrit, pour être d'un usage effectif par un programme de TAL.

Les applications du TAL que nous ciblons ne sont pas limitées. Il pourra s'agir de l'analyse, de la génération, de la traduction, de la correction, de l'extraction comme de l'indexation. Mais l'incidence du lexique ne se limite pas aux applications directes : dans la mesure où un certain nombre de terminologies d'entreprise sont couplées avec des applications de TAL. Il est en effet souhaitable que des passerelles soient définies entre la terminologie et le lexique. Pour éviter d'alourdir la description du modèle, la passerelle Lexique-TMF est présentée en fin de document.

Les lexiques pourront être monolingues ou bien multilingues.
Les langues ciblées ne sont pas limitées.

L'objet décrit ne se limite pas au mot simple mais est étendu aux mots multiples qui peuvent aller jusqu'à la quasi-phrase.

3- Normes référencées

ISO 639-2/T, Codes for the representation of names of languages.
ISO 8879:1986, (SGML) as extended by TC2 (ISO/IEC JTC 1/SC 34 N 029:1998-12-06) to allow for XML.
ISO 15924, Codes for the representation of names of scripts.
ISO 16642:2003, Computer applications in terminology – TMF (Terminological Markup Framework).
ISO 10646-1:2000, Unicode.
ISO CD12620-1, Model for description and procedures for maintenance of data category registries for language resources.
ISO NP12620-3, Data categories: electronic lexical terminological.

4- Crédits

La présente étude s'est inspirée des travaux suivants :

- Echanges de courriers électroniques de la liste de discussion « Lexiques pour le TAL » du site www.normalangue.org.
- La proposition de NWIP pour le TC37/SC4: Lexical Resource Markup Framework (LMF) par Monte George.
- De manière un peu plus générale, des travaux de l'équipe Papillon (notamment Christian Boitet et Mathieu Mangeot), d'Evelyne Jacquey, d'Eric Laporte, de Gérard Huet, de Jean Senellart, de Jean-François Delannoy, d'Alain Polguère, de Laurent Romary, de Sylvain Kahane, de Chantal Enguehard, de Jean Véronis, de Vanessa Combet, de Genelex et de mon expérience d'une douzaine d'années chez Erli-LexiQuest.

5- Définitions

Nous distinguons « lexicologie » qui est l'étude des mots, du terme « lexicographie » ou « dictionnairique » qui est l'activité de confection d'un dictionnaire. De ce fait, quand nous voulons désigner un utilisateur du modèle et dans la mesure où notre intérêt porte sur la confection de lexiques, nous parlons de lexicographe plutôt que de lexicologue. Nous n'utilisons pas le terme de « linguiste » qui n'est pas assez précis.

Nous ne faisons pas de distinction entre « dictionnaire » et « lexique ». Notons que certains auteurs [Gaudin] distinguent le dictionnaire comme étant un recueil où chaque mot possède une définition contrairement à un lexique dans lequel la définition n'est pas obligatoire.

Nous appelons un modèle de dictionnaire, l'organisation d'éléments destinés à représenter les mots d'une ou plusieurs langues. Le mot en tant qu'élément n'existe pas. Un mot est un groupement d'éléments.

Nous appelons un modèle de gestion, un modèle qui permet de créer, modifier ou supprimer des mots.
Nous appelons un modèle d'échange, un modèle qui permet d'exporter et d'importer des mots.

Les références au registre des catégories de données ISO 12620 seront exprimées entre deux slash. Ce sera par exemple : /gender/.

Nous appelons un trait morphologique, un axe de valeurs associées à la morphologie d'un mot. Nous appelons une valeur de trait, une constante particulière de ce trait. Ainsi, le trait morphologique de nombre de « maisons » aura pour valeur /plural/.
Les traits morphologiques et leurs valeurs sont définis par le « Data Category Registry » de la norme ISO 12620.

Le plus souvent nous ne parlerons pas du trait morphologique de manière isolée mais de la combinaison de différents traits morphologiques. Une telle combinaison sera par exemple : /number/ + /gender/. Nous parlerons aussi de combinaisons de valeurs de traits. Ce sera par exemple : /plural/ + /masculine/.

Nous appelons « forme lemmatisée », la chaîne de caractères représentative du mot. Pour un adjectif, ce sera le masculin singulier, pour un verbe, ce sera l'infinitif. Nous appelons « forme fléchie », la chaîne de caractère produite par combinaison de la forme lemmatisée avec une combinaison de traits morphologiques. Ainsi par exemple, « viendrait » est une forme fléchie issue de la forme lemmatisée « venir ».

Nous éviterons prudemment d'employer le terme « lexème » qui possède des sens sensiblement différents selon les auteurs. Par exemple dans la théorie Sens-Texte, un lexème est une sorte de « lexie » qui s'oppose à un « phrasème », ou locution. C'est le sens d'un mot simple. Au contraire, dans LMF il n'y a pas cette distinction, le lexème inclut le « phrasème ».

Un mot qui possède plusieurs sens sera dit polysémique. Par exemple, le mot « caillou » aura au moins deux sens : celui de « fragment de pierre » et le sens argotique qui signifie « tête ».

Toutes les chaînes de caractères sont exprimées en ISO10646-1:2000 (i.e. Unicode) donc nous ne préciserons pas le codage dans la description du modèle.

Dans les diagrammes, nous distinguerons les relations et les listes. Un élément A pourra être en relation avec un élément B, cela signifiera simplement que les éléments sont liés. Dans le diagramme, la représentation sera une flèche dotée d'une cardinalité. Dans la liste, nous avons la notion d'ordre, ainsi les éléments figurant dans la liste pourront être référencés par leur numéro d'ordre. La liste sera représentée par une boîte.

6- Etat de l'art

Depuis une vingtaine d'années un grand nombre de modèles de dictionnaires ont été définis et utilisés, avec plus ou moins de généralité. Un nombre plus restreint a été publié, et nous nous limiterons à ces derniers.
Nous pouvons schématiquement décrire ces modèles en fonction de leur filiation qui se résume à six familles généalogiques.

a) La famille de Princeton

Le modèle original est celui du lexique WordNet en version anglo-américaine créé par l'Université de Princeton. Les modèles dérivés sont EuroWordNet pour les langues de l'Europe de l'Ouest, ItalWordNet pour l'italien, IndoWordNet pour l'Asie et BalkaNet pour les langues de l'Europe de l'Est.
La version anglo-américaine de WordNet est certainement le dictionnaire le plus répandu car il est gratuit et porte sur la langue la plus répandue du domaine.

b) Les modèles Européens complexes

Le modèle original est Genelex [Antoni-Lay]. Les modèles dérivés sont Eagles, Parole, Simple, Isle et Mile. Le modèle Relex de l'IGM fait partie de cette famille.
Ces travaux ont fortement influencé la pratique du TAL en Europe. Ces modèles sont puissants, mais le principal reproche à leur faire est que ce sont des modèles complexes qui ne sont pas simplifiables. Ils sont le fait de consortiums qui, pour satisfaire tous les partenaires ont produit l'union des mécanismes de représentation. De ce fait, un grand nombre d'acteurs en Europe n'implémente qu'une petite partie de ces modèles sans prendre en compte la totalité des mécanismes de représentation.

c) Les modèles Européens simples
Ce sont BDLex, Celex, Multex et « Multex goes East ». A partir de 1994, Multext a été utilisé pour un grand nombre de langues européennes (de l'ouest et de l'est) avec des extensions aux langues asiatiques comme le vietnamien. Notons que ces modèles n'ont pas forcément de liens de filiation entre eux : ils partagent le fait qu'ils sont simples et traitent principalement de morpho-syntaxe.

d) Le modèle EDR
Il s'agit du modèle du consortium japonais EDR dont les travaux sont repris actuellement par le CRL. C'est un modèle bilingue spécifiquement destiné au couple japonais-anglais.

e) La famille des modèles de Mel'cuk
Le modèle original est le DEC (Dictionnaire explicatif et combinatoire [Mel'cuk]) qui est un peu particulier dans la mesure où le dictionnaire associé n'a pas été utilisé pour le TAL. En revanche, les modèles dérivés comme DiCo, Bdéf [Altman] ou Papillon [Mangeot] sont destinés au TAL.
La théorie sous-jacente, qui est la théorie Sens-Texte, fait de cette famille de dictionnaires un exemple significatif de travail lexicographique. La pratique en question se distingue par un jeu de critères méthodologiques particulièrement rigoureux.

f) La famille TEI
Ces modèles se fondent sur les directives TEI. Ce sont ALLEX (African languages lexicons) et CJKE (Chineese, japaneese, Korean and English). Notons que leur utilisation dans une perspective TAL n'est pas avérée.

7- Problématique
Du fait de la diversité des modèles, l'échange de données n'est pas très facile, surtout entre des lexiques de familles différentes. La fusion de dictionnaires reste une opération très complexe. Il n'est pas très facile non plus de faire cohabiter des programmes qui opèrent sur des modèles de lexiques différents.
Il existe au moins trois cas d'utilisation :

Cas-1 :		
	Situation :	un lexique multilingue en N langues
	Objectif :	ajouter une nouvelle langue
Cas-2 :		
	Situation :	un programme monolingue fonctionnant sur N langues
	Objectif :	utiliser le programme sur une nouvelle langue
Cas-3 :		
	Situation :	un lexique monolingue dans la langue L
	Objectif :	ajouter des mots de la langue L

De plus, pour certains modèles, la structure n'est pas très bien définie. Certains modèles ont quelques difficultés à représenter les informations linguistiques que **la langue nous force à décrire** : le phénomène est particulièrement criant pour la description des mots composés et les opérations de transfert verbal pour la traduction.

8- Critères imposés
L'orientation que nous adoptons est régie par les critères que nous nous imposons.

a) Critère de simplicité
Le modèle doit être simple pour un lexicographe qui désire la simplicité.
Le modèle doit être puissant pour un lexicographe qui veut de l'expressivité tout en admettant un peu de complexité.

b) Critère de représentativité
Le modèle doit être capable de représenter, dans la mesure du possible, les dictionnaires existants. Si tel n'est pas le cas, l'information problématique doit pouvoir être détectée et isolée.
c) Critère de distinction par rapport aux autres modèles
Le modèle ne doit pas constituer une septième famille mais doit à la fois s'inspirer de tous les modèles existants et en être une représentation pivot.

9- Modèle proposé

a) Choix

a-1) nombre de couches
Concernant la structure monolingue du modèle, nous avons deux possibilités. La première consiste à définir un modèle avec deux couches : la morphologie et la sémantique. La seconde consiste à intercaler une information syntaxique entre la morphologie du mot et ses différents sens. Cela produit un modèle à trois couches : la morphologie, la syntaxe et la sémantique.
Par expérience, nous savons que les modèles à trois couches ne peuvent respecter le critère de simplicité car l'information syntaxique est un passage obligé entre la morphologie et la sémantique. Elle complexifie chaque entrées lexicale, même si l'on ne désire pas décrire spécialement le comportement syntaxique du mot. Pour cette raison, nous choisissons une modèle à deux couches. Si nous avons besoin de représenter les informations syntaxiques nous les projèterons sur les sens du mots, mais cette projection sera optionnelle.
Notons qu'il existe des modèles avec plus de trois couches comme le modèle Sens-Texte avec sept couches.

a-2) absence vs présence de la morphologie
Certains acteurs du domaine ne désirent pas faire figurer le calcul de la morphologie associée à chaque mot au prétexte que c'est soi-disant trop complexe ou bien qu'il n'y a pas consensus. L'argument qui milite contre cette position est que tous les dictionnaires pour le TAL possèdent l'information morphologique : il est en effet impossible d'utiliser un dictionnaire si celui-ci n'est pas capable d'associer une forme lemmatisée à une ou plusieurs formes fléchies, ceci est vrai pour les mots simples comme pour les mots composés.
Il faut considérer que l'information morphologique fait partie du dictionnaire au même titre que les entrées, si le modèle en tient compte, les possibilités d'échanges de données seront beaucoup plus importantes que si tel n'est pas le cas.
Imaginons le contexte suivant : supposons que nous ayons des logiciels et des lexiques qui fonctionnent en N langues et qu'il s'agisse d'ajouter une langue par acquisition via un format normalisé. Si la morphologie de la langue est relativement simple comme peut être l'anglais, le problème n'est très important. Mais si la morphologie de la langue en question est complexe comme le français (51 formes pour chaque verbe) ou le hongrois (238 formes pour chaque nom), il sera très appréciable de disposer de la morphologie avec les entrées plutôt que d'avoir à créer les paradigmes de flexion par ailleurs.

Il est vrai qu'il n'y a pas consensus à ce propos et l'on peut schématiser la situation en quatre orientations :

- les formes fléchies sont représentées explicitement,
- le code fait référence à un automate comme le pratique l'IGM,
- le code est une description symbolique comme Genelex ou Eagles,
- le code est une référence à un programme compilé opaque.

Pour ceux qui pratiquent la dernière orientation, il n'y aucun espoir de modélisation ou d'échange de données. Si l'orientation avec les formes fléchies est envisageable pour les langues à morphologie simple, elle n'est pas conseillée pour les langues à morphologie complexe.

Une autre façon de voir les choses est de considérer que le modèle est à la fois un modèle de gestion et un modèle d'échange. En tant que modèle de gestion, il est alors possible de gérer la morphologie via une description symbolique et de considérer que les formes fléchies ne sont que des informations engendrées qui seront produites à partir de la description symbolique, et ceci en considérant que le modèle sert à l'échange de données. L'intérêt d'un tel processus est double : d'une part, pour gérer les mots il est plus facile d'associer un code que de décrire les formes fléchies et d'autre part, le consommateur de données n'a pas besoin d'évaluer les descriptions symboliques pour connaître les formes fléchies : elles sont explicitement disponibles.

Nous prenons le parti suivant :

- il est possible de représenter les formes fléchies explicitement,
- il est possible de représenter la description sous forme symbolique et celle-ci peut être compilée sous forme d'automate.

Pour la morphologie des mots simples, la description symbolique repose sur une liste de radicaux qui est modifiable par une séquence de sept opérateurs linguistiques (c.f. détails dans le système de la morphologie).
Pour la morphologie des mots multiples, la description repose sur les combinateurs de traits morphologiques en s'inspirant du modèle Eagles tout en l'améliorant légèrement.

a-3) représentation graphique vs représentation phonétique
La forme lemmatisée graphique est obligatoire, en revanche la représentation phonétique de la forme lemmatisée est optionnelle. Si le lexicographe décide de décrire la morphologie graphique d'un mot, il n'est pas obligé de représenter la phonétique du mot. De même, si le lexicographe décide de représenter la morphologie phonétique du mot, il n'est pas obligé de représenter son équivalent graphique.

a-4) multilinguisme
Le multilinguisme est privilégié par rapport au bilinguisme. Le mécanisme du pivot de traduction est préféré à la définition d'un lien bilingue.

b) Structure
L'esprit général est le suivant :

Pour chaque mot, nous avons un squelette et des systèmes périphériques.
Le squelette est rigide et obligatoire : il est simple.
Les systèmes périphériques sont au contraire souples et optionnels : ils sont puissants.

Le squelette s'inspire du cœur de LMF et est composé :

- du système de l'entrée lexicale,
- du système du sens.

Le premier est pour le signifiant et le second pour le signifié. La cardinalité est d'un à N : un signifiant et éventuellement plusieurs sens afin de représenter la polysémie.

Les systèmes périphériques sont :

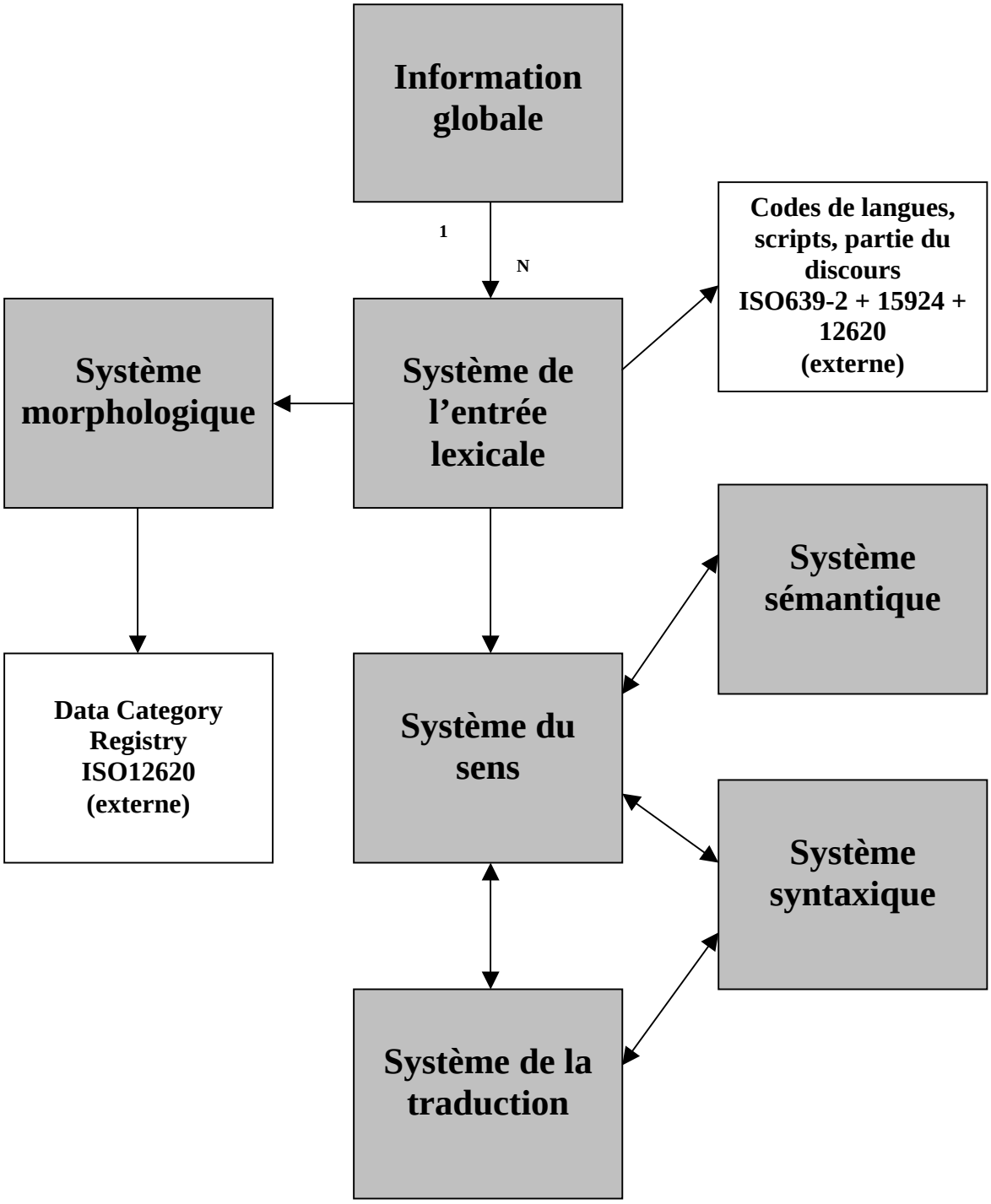
- la morphologie,
- la syntaxe,
- la sémantique,
- la traduction.

Afin de présenter la structure progressivement, nous commençons par présenter l'architecture générale puis, en détail, nous décrirons chaque sous-système.

c) Architecture générale

L'information globale est un élément **unique** qui permet de d'enregistrer, entre autres choses, le nom, la version, les mentions de copyright.

Par convention, dans le diagramme, seuls les éléments grisés appartiennent au modèle. Les informations dans des boîtes blanches sont prises dans d'autres normes.



d) Détail de chaque sous-système

d-1) Le système de l'entrée lexicale

L'entrée lexicale sert à représenter le mot en tant qu'entité morphologique, que l'on décrive ou non sa morphologie complète. C'est le 'Morphological Unit' du modèle Eagles. C'est le signifiant de Saussure.
Il y a autant d'entrées lexicales qu'il y a de mots dans le dictionnaire.

L'entrée contient les attributs obligatoires :

- /identifier/,
- le code de la langue qui est une valeur prise dans la norme ISO639-2/T,
- le code du système de script qui est une valeur prise dans la norme ISO15924.
- la catégorie grammaticale (partie du discours), qui est une valeur prise dans le registre de catégorie de données de la norme ISO12620. Ce sera par exemple : /nom commun/.
- la graphie lemmatisée qui est une chaîne de caractères.
- si le mot est (ou non) autonome.

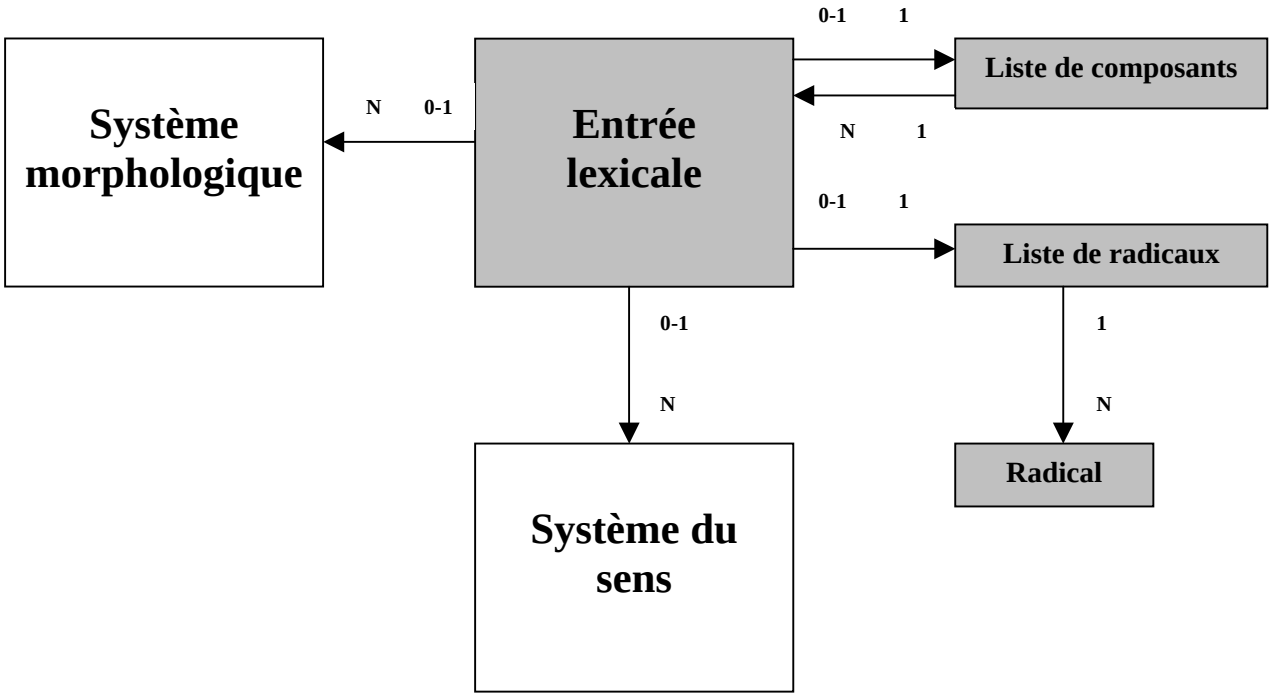
Les attributs optionnels sont les suivants :

- la représentation phonétique lemmatisée qui est une chaîne de caractères.
- la composition. C'est la liste des composants qui sont d'autres entrées lexicales. Quand la liste est vide, l'entrée lexicale est celle d'un mot simple.
- les radicaux. C'est la liste sur laquelle repose la morphologie flexionnelle (voir le système morphologique). Chaque radical porte une chaîne de caractères optionnelle pour la graphie et une chaîne optionnelle pour la phonétique. Au moins une des deux chaînes doit être renseignée. Un radical n'est pas partagé par plusieurs entrées lexicales.

Dans ses relations avec les systèmes voisins :

- l'entrée lexicale possède zéro ou un lien vers une description morphologie. Concernant l'absence de lien, ce n'est pas que la description n'existe pas en langue, c'est que l'on n'impose pas au lexicographe.
- l'entrée possède zéro à N sens.

Par convention, seuls les éléments grisés appartiennent au système de l'entrée lexicale.



d-2) Le système du sens

C'est le système du modèle LMF. Dans la théorie Sens-Texte, l'élément s'appelle « lexie ». Pour Eagles, l'élément s'appelle « Semantic Unit ». Pour Saussure, c'est le signifié. En français, c'est une acception.

Un mot est défini par une entrée lexicale et un ou plusieurs sens. Si nous avons plusieurs sens, le mot sera dit polysémique.
Un sens n'est relié qu'à une seule entrée lexicale. Les sens ne sont pas partagés afin de rendre compte d'une synonymie. **Pour exprimer un tel phénomène, il faut utiliser une relation entre deux sens**. Nous aurions pu permettre un lien 1 à N entre le sens et l'entrée lexicale afin de représenter une configuration avec deux entrées lexicales qui référencent une même unité de sens. Ce n'est pas ce que nous avons fait, car cela complexifie inutilement le modèle pour des situations rarissimes. En effet, les études linguistiques montrent que les véritables synonymes n'existent quasiment pas. Même les exemples classiques comme 'automobile' et 'voiture' (le sens associé à l'objet qui roule sur la route et non sur un chemin de fer) sont discutables. Le niveau de langue est sensiblement différent ; 'automobile' est plus soutenu que 'voiture'.

Le sens contient l'attribut obligatoire :

- /identifier/,
- la KeyForm pour la représentation externe, par exemple : 'souris (animal)', exprimée sous forme d'une chaîne de caractères.

Le sens contient l'attribut optionnel :

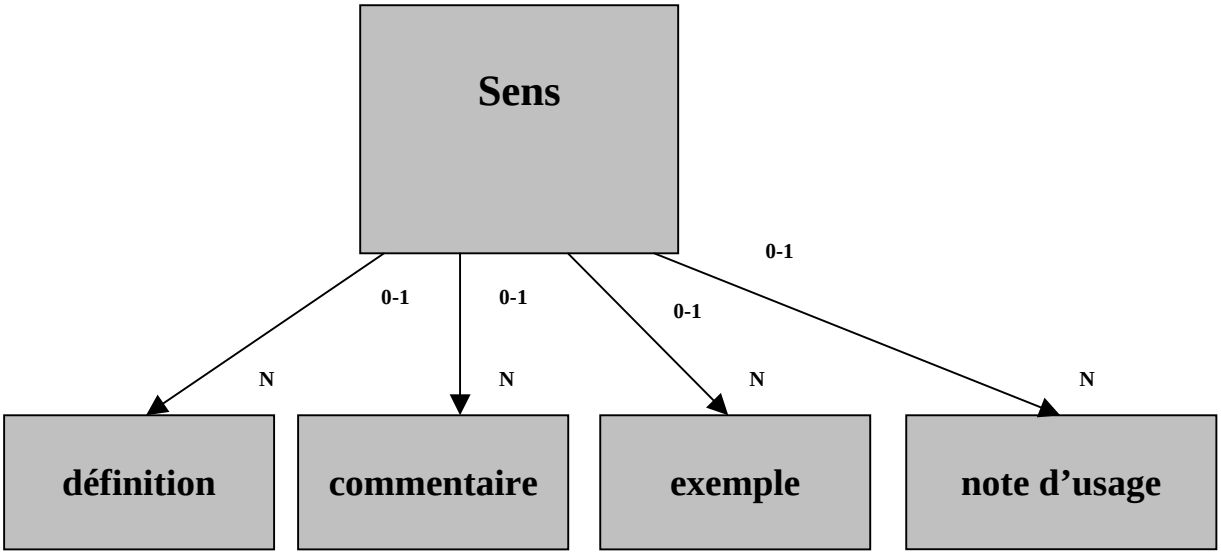
- Les marques d'usage définies dans l'ISO 12620.

Les éléments additionnels sont :

- la définition sous forme d'un texte,
- le commentaire,
- l'exemple,
- la note d'usage.

Ces éléments peuvent être multiples et sont optionnels.
Chacun d'eux porte deux attributs :

- le contenu, sous forme d'une chaîne de caractères.
- la langue d'expression qui n'est pas nécessairement la langue du mot. La valeur est prise dans la norme ISO 639-2.



d-3) Le système morphologique

d-3.1 Introduction
La fonction du système est de produire toutes les paires : combinaison de valeurs de traits morphologiques / formes fléchies. Par exemple, pour l'entrée 'go', une de ces paires sera : (/third person/ + /singular/ + /present/) / 'goes'.

Le système traite les mots simples et multiples.
Le système fonctionne en génération aussi bien qu'en analyse.
Le système traite l'information graphique et phonétique.

d-3.2 Le mot simple
Le système est celui d'Eagles qui a été modifié pour être rendu plus puissant.

Le mécanisme pour décrire la morphologie d'un mot simple est le suivant :

- on fait référence à la forme lemmatisée ou à une liste de radicaux qui est attachée à l'entrée lexicale. La référence zéro signifie la forme lemmatisée. Un entier supérieur à un est le rang du radical.
- on spécifie des opérateurs que l'on applique à la chaîne obtenue au point précédent. Il y a entre zéro et sept opérateurs qui sont appliqués dans l'ordre dans lequel ils sont présentés dans la séquence suivante :
  - opérateur-1 : retirer N caractères en début.
  - opérateur-2 : ajouter une chaîne en début.
  - opérateur-3 : enlever N caractères en fin.
  - opérateur-4 : ajouter une chaîne en fin.
  - opérateur-5 : remplacer N caractères à la position X par une chaîne.
  - opérateur-6 : déplacer N caractères depuis la position X jusqu'à la position Y.
  - opérateur-7 : dupliquer N caractères depuis la position X jusqu'à la position Y.

  Chaque opérateur ne s'applique qu'une seule fois. Les positions s'expriment par des entiers positifs pour signifier que l'on compte à partir du début du mot et par des entiers négatifs pour signifier que l'on compte depuis la fin du mot.

Le paradigme de flexion porte les attributs suivants :

- /identifier/.
- exemple.
- partie du discours auquel il est destiné. C'est une référence à un registre de catégories de données.
- si le paradigme s'applique à la représentation graphique.
- si le paradigme s'applique à la représentation phonétique.

Le combinateur de traits morphologique ne porte aucun attribut propre, seulement des liens. C'est un objet non-linguistique qui ne sert qu'à relier des éléments.

Le composeur porte les traits suivants :

- s'il est simple ou composé.
- le rang.

Le calculateur de formes fléchies porte un double jeu d'opérateurs : un jeu pour le calcul graphique et un jeu pour le calcul phonétique.
Ses attributs sont les suivants :

- Le radical. C'est un entier qui fait référence aux radicaux attachés à l'entrée lexicale. La valeur zéro désigne la forme lemmatisée.
- graphiqueEnleverAvant. C'est un entier.
- graphiqueAjouterAvant. C'est une chaîne.
- graphiqueEnleverAprès. C'est un entier.
- graphiqueAjouterAprès. C'est une chaîne.
- graphiqueRemplacer. C'est un entier.
- graphiqueRemplacerEn. C'est un entier.
- graphiqueRemplacerPar. C'est une chaîne.
- graphiqueDéplacer. C'est un entier.
- graphiqueDéplacerDepuis. C'est un entier.
- graphiqueDéplacerJusquA. C'est un entier.
- graphiqueDupliquer. C'est un entier.
- graphiqueDupliquerDepuis. C'est un entier.
- graphiqueDupliquerJusquA. C'est un entier.
- Idem pour la phonétique.
- contexteVariation. C'est un code libre qui permet d'enregistrer des variations dues aux voisins du mot. On pourra, en français, par exemple coder les phénomènes très rares de variation de forme quand le mot est suivi d'une voyelle ou d'un H muet. On l'appliquera à l'adjectif « beau », qui devant un tel mot s'écrit « bel » au lieu de « beau » quand il est au masculin singulier.

Le système est certainement plus facile à comprendre en consultant le chapitre des exemples de mots.

d-3.3 Le mot multiple

Le système retenu est une combinaison des mécanismes de composition morphologique et syntaxique de Genelex. En fait, on est plus ou moins obligé de faire intervenir des éléments de syntaxe même si celle-ci reste locale.

On peut définir le mot multiple selon deux vues :
1) L'aspect intrinsèque :
Si le mécanisme des mots simples combinent des chaînes pour former un mot, celui des mots multiples combine des mots pour en former un autre.
2) L'aspect extrinsèque
Le mot multiple est une séquence de mots qui se comporte comme une unité simple à un certain niveau de l'analyse linguistique [Calzolari].

Un mot multiple est soit :

- un mot composé continu comme : « pomme de terre ». Le mot n'est pas nécessairement un nom, il peut être de n'importe quelle catégorie grammaticale.
- un mot agglutiné.
- un mot discontinu comme « passer en revue ». On appelle un mot discontinu, un mot qui peut admettre une insertion sans que celle-ci soit obligatoire. On pourra former : « passer N1 en revue » tout aussi bien que « passer en revue N1 » sans qu'il soit nécessaire d'enregistrer deux entrées lexicales.

Un mot multiple peut être une construction à verbe support comme « faire une acquisition ».

Un mot multiple est constitué de composants qui sont ou non autonomes. Ainsi, dans « au fur et à mesure », le composant « fur » n'a pas d'existence en tant que mot isolé, on l'appelle alors un mot non-autonome.

Le mécanisme s'applique pour former les agglutinés. En fait, on considère qu'un mot agglutiné est un mot composé qui ne comporte aucun séparateur graphique.

Le mécanisme s'applique récursivement : un mot multiple peut être constitué de composants qui sont eux-même des mots multiples. Par exemple : l'adjectif composé « à haute tension » peut former le nom composé « ligne à haute tension ». De même, un mot multiple peut comporter des composants qui sont des agglutinés.

Le système combine trois types de descriptions :

- une information qui porte sur plus d'un mot. C'est un codage qui peut porter sur la totalité du mot : par exemple, si le composé admet ou non un ordre libre de ses composants.
- les insertions possibles en précisant leur type et position. Ce sera utile pour décrire « passer N1 en revue ».
- une information qui porte sur chacun des composants en précisant :
  - la combinaison de traits morphologiques. C'est ce qui va déterminer le calcul des formes fléchies à partie des composants.
  - la référence au lemme ou radical. C'est un entier.
  - le séparateur graphique, par exemple, un tiret, un espace ou la possibilité d'avoir les deux. La valeur « jointure » permet de représenter l'agglutination. Par convention, le séparateur s'applique après le composant.
  - un code libre, afin par exemple de spécifier un changement de casse. Dans certaines langues agglutinantes, le composant autonome commence par une majuscule qui devient minuscule en agglutination.

Notons que nous ne représentons pas deux types d'information :

- la désignation de la tête.
- l'accord entre une sous-partie des composants. En fait, on l'exprime plus ou moins : on n'exprime pas explicitement qu'un composant s'accorde avec un autre composant, mais l'on spécifie que le composant s'accorde de telle ou telle manière avec le composé.

En français, pour les mots composés continus, nous avons les structures régulières qui sont au nombre de 9 : Adj-N, N-Adj, NàN, NàGN, NdeN, NdeGN, N-N , V-N, Prép-N [Silberztein], [Gaudin]. En définissant ces neuf paradigmes de flexion composée, nous couvrons l'écrasante majorité des mots composés du français. La tâche n'est pas finie pour autant car il reste un certain nombre de mots qui sont particuliers et qui nécessitent une description spécifique.

Le système est certainement plus facile à comprendre en consultant le chapitre des fragments d'exemples de mots.

d-3.4 Critères d'éclatement

De par sa structure, le modèle définit quand le lexicographe doit faire deux éléments plutôt que deux et inversement. Le système morphologique permet de décrire quels sont les traits morphologiques comme le genre qui sont affectés à l'entrée lexicale. Mais les traits morphologiques ne sont pas directement associés à l'entrée lexicale ; ils sont décrits dans le combinateur de traits morphologiques. Par exemple, en français, pour la graphie 'moule', nous avons deux entrées lexicales, l'une pour 'un moule' et l'autre pour 'une moule'. Pour un lexicographe qui a choisi la stratégie du paradigme de flexion, il aura deux paradigmes ; l'un masculin et l'autre féminin.
Notons qu'un lexicographe qui décrit le mot 'moule' sans se préoccuper de la morphologie, n'a pas lieu de faire deux éléments : un seul suffit.

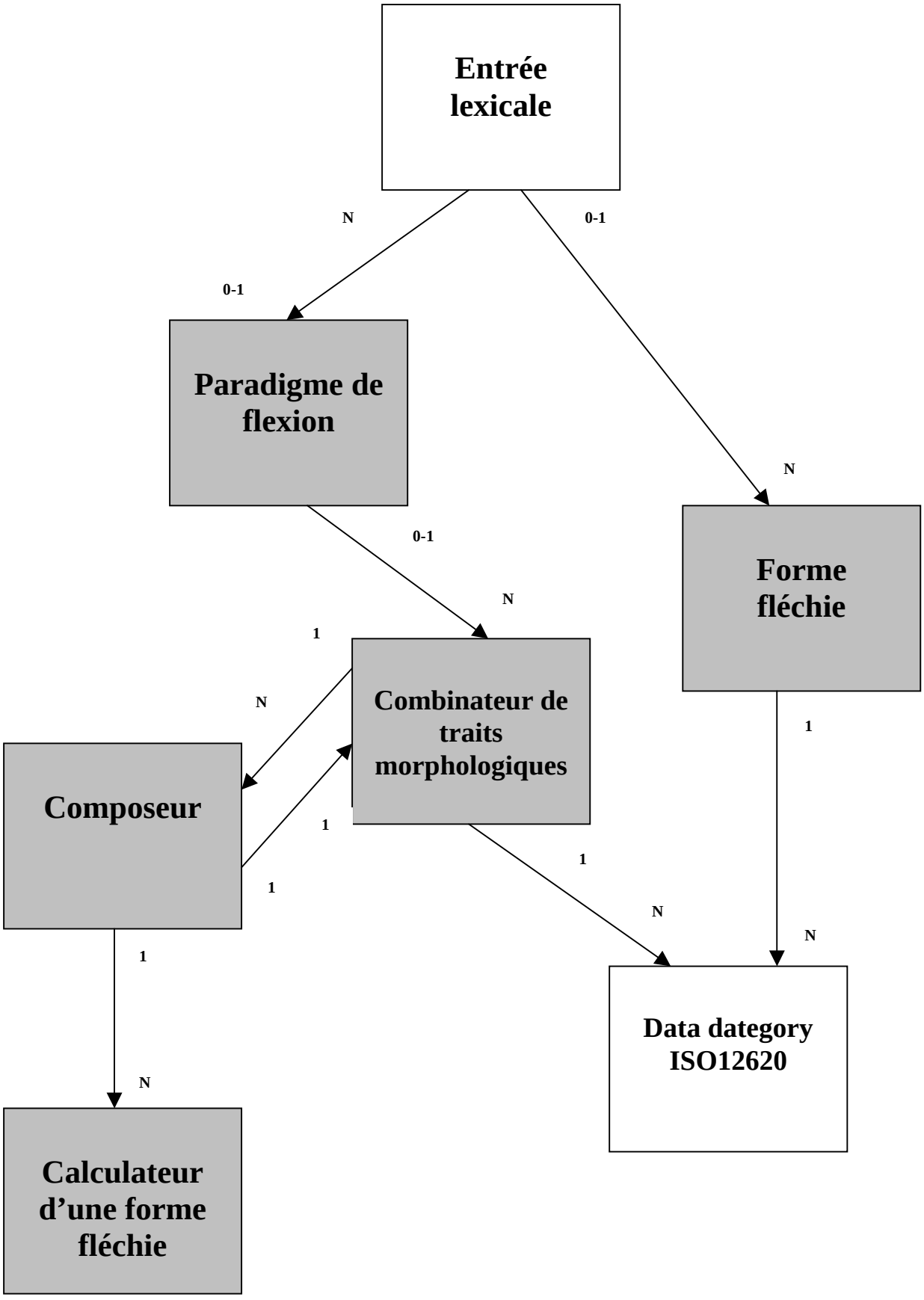
d-3.5 Description à profondeur variable

On observe que l'entrée lexicale peut être plus ou moins décrite en ce qui concerne son comportement morphologique. En plus du double système paradigme vs forme fléchies, une certaine liberté est laissée quand à la précision de la description.
Il est possible de manière graduelle :

- d'indiquer que le mot se fléchit selon un certain paradigme P, sans pour autant définir l'élément P de manière analytique. En d'autres termes, cela consiste simplement à associer une étiquette à l'entrée lexicale. Le lexicographe doit dans ce cas définir un paradigme de flexion qui porte l'identifiant P et y faire référence dans les entrées lexicales, et c'est tout.
- de décrire sommairement la morphologie des mots en précisant seulement les traits morphologiques. Il suffit de définir les paradigmes et les combinateurs sans les associer aux composeurs. Pour reprendre l'exemple de la chaîne graphie 'moule', si le lexicographe désire simplement indiquer qu'un mot est masculin et que l'autre est féminin, tout en ayant des pluriels et sans définir la production des formes fléchies, il faut deux entrées lexicales, deux paradigmes. Le premier paradigme possède un combinateur associé au masculin. Et le second paradigme possède un combinateur associé au féminin.
- de décrire complètement les paradigmes, combinateurs et composeurs.

En ce qui concerne l'utilisation d'un système morphologique partiellement décrit, le modèle le permet, mais c'est plus à envisager comme une propriété qui permet de faire vivre le lexique tout en respectant le modèle, car évidemment, **seule une description complète sera opérationnelle dans un analyseur ou un générateur morphologique**. Cette remarque ne s'applique pas aux acteurs qui n'enregistrent volontairement qu'une partie de l'information dans le dictionnaire et utilisent des routines externes opaques pour réaliser les traitements, ce sera typiquement un paradigme qui portera le nom d'un programme compilé spécialement pour réaliser l'analyse ou la génération du mot.

Par convention, les éléments du système de la morphologie sont grisés.



d-4) Le système sémantique

Il s'agit de représenter les relations sémantiques et les traits sémantiques.

Une relation permet de qualifier le lien qu'entretient deux sens. Ces derniers doivent appartenir à la même langue, en effet, les mécanismes de liaison d'une langue à l'autre sont couverts par un autre système : le système de la traduction.

Un trait est simplement une information attachée à un sens. Un trait est un cas particulier de la relation : c'est une relation qui n'a pas de cible.

La relation permet de représenter la dérivation sémantique, la synonymie, la parasynonymie des dictionnaire éditoriaux. Il peut tout aussi bien représenter les fonctions lexicales du DEC. Les liens de synonymie permettent aussi de décrire les synsets (synonyms in a set) de WordNet.

Le trait permet de représenter par exemple, le mécanisme du champ sémantique du DEC.

Plus précisément, le système comprend deux éléments : la relation qui est l'instance qui relie les sens et le type de relation qui permet de qualifier de manière factorisée la relation. Les types de relation peuvent entretenir entre eux des liens d'héritage.

Le type de relation dispose des attributs suivants :

- /identifier/,
- nom qui est une chaîne,
- commentaire qui est une chaîne,
- exemple qui est une chaîne,
- lien d'héritage.

Le bloc définitionnel et la proposition permettent de décrire de manière analytique le sens. Notons qu'un exemple d'usage est donné dans le chapitre : fragments d'exemples.

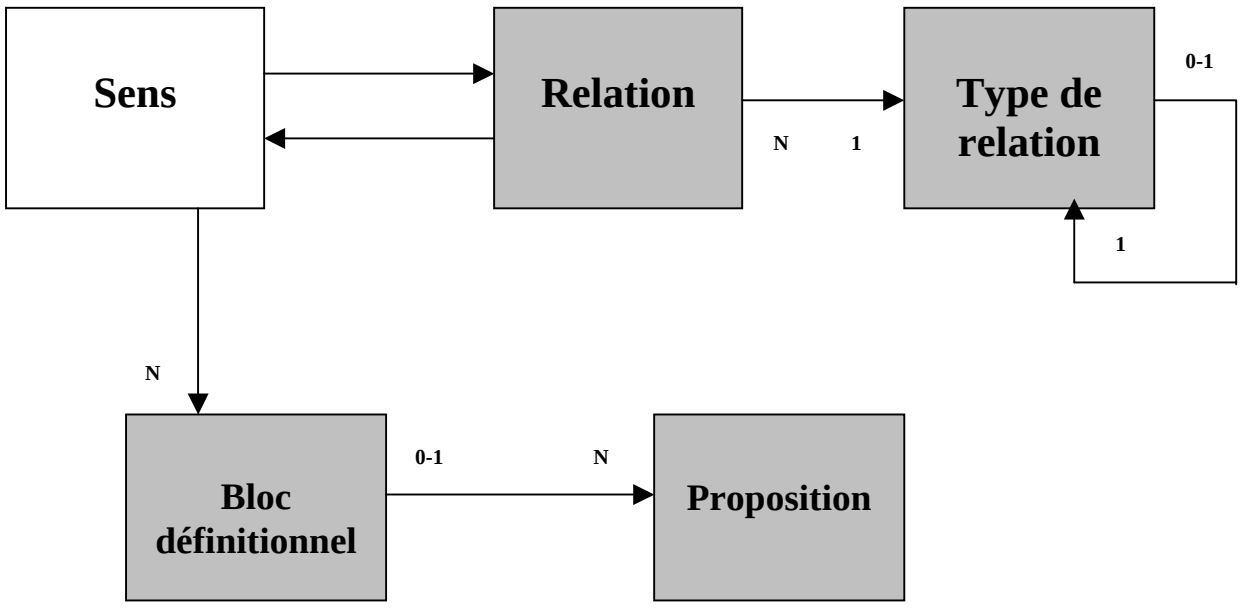
Le bloc définitionnel porte l'attribut :

- nom qui est une chaîne.

La proposition porte les attributs :

- nom qui est une chaîne,
- type qui est une chaîne,
- texte qui est une chaîne.

Par convention, les éléments du système sémantique sont grisés.



d-5) Le système syntaxique

Le système syntaxique est inspiré du mécanisme écrit par A Sanfilippo pour Eagles. Se fondant sur le fait que l'on ne peut pas faire de syntaxe sans sémantique, ce n'est pas un système purement syntaxique, c'est plutôt un système syntaxico-sémantique.

La description syntaxique est attachée au sens. Ainsi pour un mot polysémique, nous pouvons avoir autant de descriptions syntaxiques que nous avons de sens. La description syntaxique est optionnelle.

On peut répartir les propriétés syntaxiques en deux groupes : celles qui caractérisent le mot comme dépendant syntaxique et celles qui concernent la façon dont le mot détermine le comportement de ses dépendants [Mel'cuk 95]. Les propriétés du premier groupe concernent les capacités du mot d'entrer dans certaines constructions syntaxiques ; elles sont plus grammaticales que lexicales, et nous les négligerons. Quant aux propriétés syntaxiques du deuxième groupe, elles visent avant tout les actants syntaxiques et sont liées au lexique. Elles nous concernent donc.

On appelle un actant syntaxique du sens S, un syntagme qui dépend de S et qui en exprime un actant sémantique.

Un sens peut avoir plusieurs comportements syntaxiques. L'élément « comportement syntaxique » est simplement un connecteur entre des éléments : il ne porte aucun attribut.
Le cadre porte les attributs suivants :

- /identifier/
- commentaire

Le slot porte l'attribut suivant :

- fonction

L'actant syntaxique porte les attributs suivants :

- définition
- commentaire,
- introducteur. Ce sera par exemple la préposition qui débute le groupe syntaxique.
- groupe syntaxique
- restriction

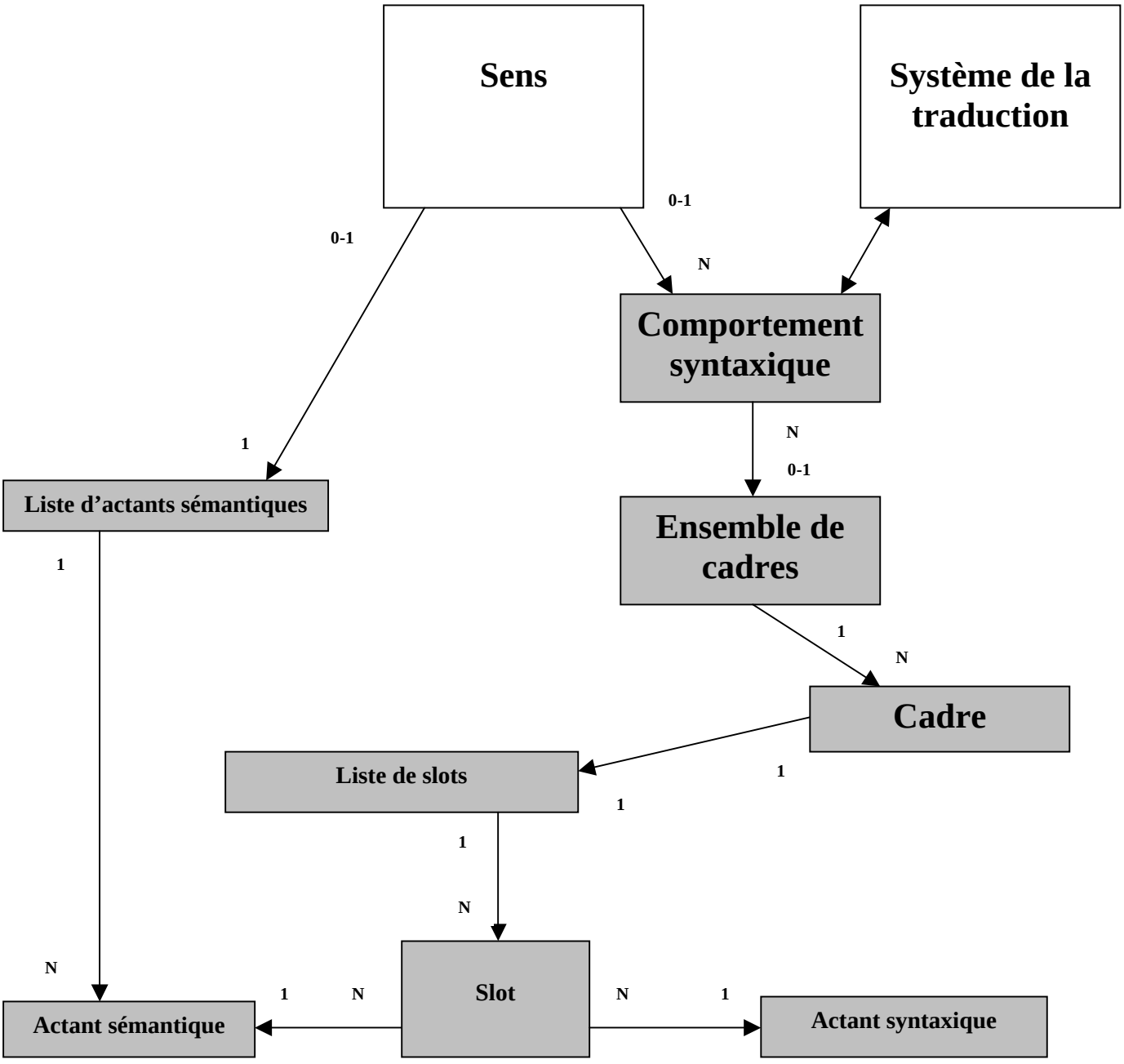
L'actant sémantique porte les attributs :

- nom
- restriction

Le système est surtout destiné aux verbes, aux noms prédicatifs et aux adjectifs.

Notons qu'un exemple d'usage est présenté dans le chapitre : «fragments d'exemples ».

Par convention, les éléments du système syntaxique sont grisés.



d-6) Le système de la traduction
Le mécanisme s'inspire du modèle Papillon.

Dans un dictionnaire bilingue, nous avons besoin d'un lien pour traduire un sens en un autre et on pourrait imaginer naïvement qu'il suffit d'un simple lien entre deux sens.

En fait, il y a deux types de problèmes :
a) **Premier problème**
Dans certains cas, cela ne fonctionne pas très bien parce que la finesse de la langue source n'est pas la même que celle de la langue cible. Ainsi, pour traduire le français « fleuve » (rivière qui se jette dans la mer) en anglais, nous arrivons pas à être aussi précis parce que le mot (donc le sens) n'existe pas dans la langue cible. Une solution consiste à créer un objet intermédiaire (que certains appellent une « cheville ») et d'indiquer une spécialisation par rapport au lien de traduction « rivière » vers l'anglais « river ».
b) **Deuxième problème**
Si cette stratégie est viable pour deux langues, elle est intenable pour un nombre de langues plus important comme quatre ou cinq tout bonnement parce que le nombre de liens explose. Si la stratégie du lien bilingue était possible il y a une quinzaine d'années quand les dictionnaires étaient petits et le nombre de langues restreint, cette stratégie n'est plus possible actuellement. Prenons l'exemple de l'Europe qui est une mosaïque de langues et où la plupart des acteurs du domaine opèrent en un minimum de cinq langues avec des dictionnaires qui atteignent des centaines de milliers de sens. Le nombre de liens est de l'ordre du carré du nombre de sens, autrement dit quadratique. La formule exact est : (N * (N-1)) / 2. Pour cinq langues, le nombre de liens avec un dictionnaire de 100 000 sens va imposer la gestion d'un million de liens. Et il faut considérer cinq comme étant un minimum, certains acteurs gèrent une douzaine de langues.

Pour éviter ces problèmes, nous représentons les traductions via un objet intermédiaire que l'on appelle « axie ». C'est une structure pivot qui met en relation des éléments appartenant à des dictionnaires de langues différentes.
C'est une structure qui n'a pas lieu d'exister dans un dictionnaire monolingue.

D'autre part, en ce qui concerne les TAL multilingues, il y a deux écoles : celle du transfert et celle du pivot. Le transfert opère en syntaxe et le pivot en sémantique. Le transfert consiste à traduire en se fondant sur l'information syntaxique d'une langue pour aboutir à une structure syntaxique de la langue cible. L'approche via un pivot consiste à retrouver un élément qui ne dépend pas de la langue (aka pivot interlingua) et ensuite à engendrer la phrase dans la langue cible via un programme de génération. Notons que la majorité des outils de traduction (ou d'aide à la traduction) commerciaux se fondent sur l'approche transfert. Mais, on ne peut négliger les partisans de l'approche pivot qui se focalisent sur des niches techniques ou qui sont associés à des mécanismes de représentation interlingua. De plus, la traduction n'est pas la seule application d'un lexique multilingue : la recherche trans-linguistique d'information (i.e. Cross-Lingual Information Retrieval) est un domaine important, même si il est moins visible [Peters]. Et dans ce domaine, la répartition transfert vs pivot est moins tranchée.

De ce fait, le modèle de lexique doit permettre de pratiquer les deux approches. Notons qu'il est techniquement envisageable de combiner les deux approches, même si la complexité accrue serait problématique.

On distingue :

1) L'axie de sens qui relie plusieurs sens de langues différentes.
Cette axie sert à implémenter l'approche pivot. Elle peut servir aussi pour l'approche par transfert, dans les situations où l'on dispose de la traduction sans avoir quoi que ce soit à exprimer sur la syntaxe du sens.

L’axie permet de traduire des mots qui n’ont pas nécessairement le même statut d’une langue à l’autre. Ainsi, dans la langue source, nous pourrons avoir un mot simple et qui se traduira par un mot composé dans la langue cible.
Les axies entre elles peuvent être décrites les unes par rapport aux autres via une relation d’axie.
La relation d’axie porte trois attributs optionnels :

- un intitulé qui est une chaîne de caractères.
- le nom d’un système descriptif externe.
- la référence dans le système descriptif externe.

L’intitulé permet de coder des relations interlingua simples comme le raffinement de « fleuve » comparativement à « rivière » et « river ». Mais il n’a pas pour objectif de coder un système complexe de représentation de connaissances : pour cela, il est préférable d’utiliser un système cohérent, complet et surtout conçu pour cela. Un bon candidat est UNL, éventuellement associé à des fonctions lexicales pour représenter des choses un peu délicates comme les collocations à verbe support [Boguslavsky] dans un système pivot.

2) **L’axie de transfert** qui permet de réaliser le transfert.
La liaison entre plusieurs langues se situe au niveau syntaxique.
On peut rendre compte des phénomènes d’inversion d’actants syntaxiques comme :

« FR :Elle me manque » => « EN : I miss her »

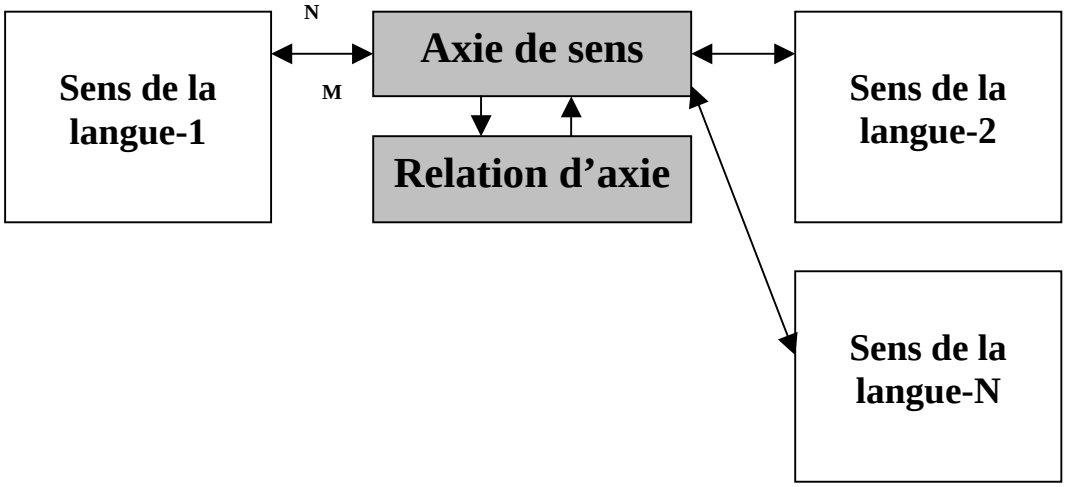
Dans la mesure où une entrée lexicale peut être une construction à verbe support, on peut rendre compte des traductions qui font passer d’un verbe plein (dans la langue-1) à un verbe support (dans la langue-2) comme :

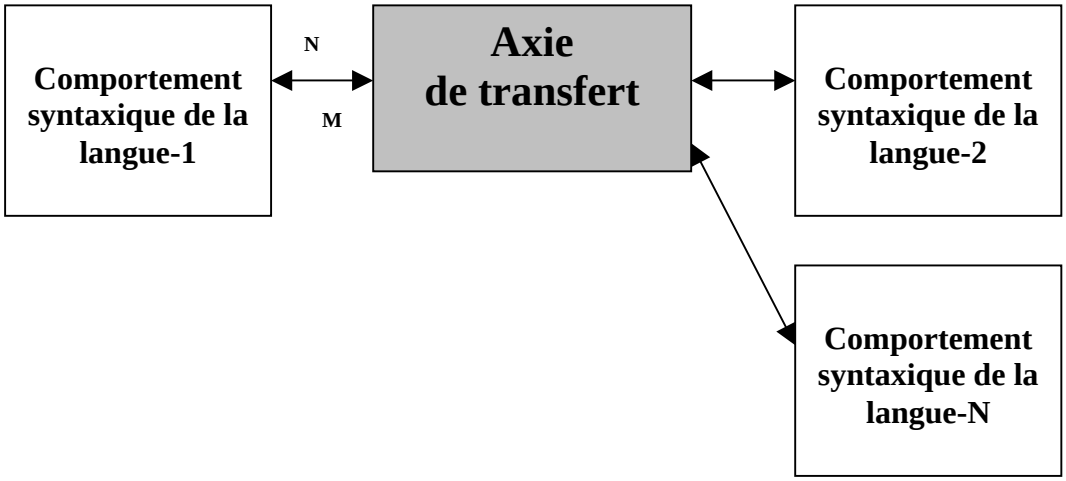
"FR: Marie rêve" => "JP: Marie wa yume wo miru".

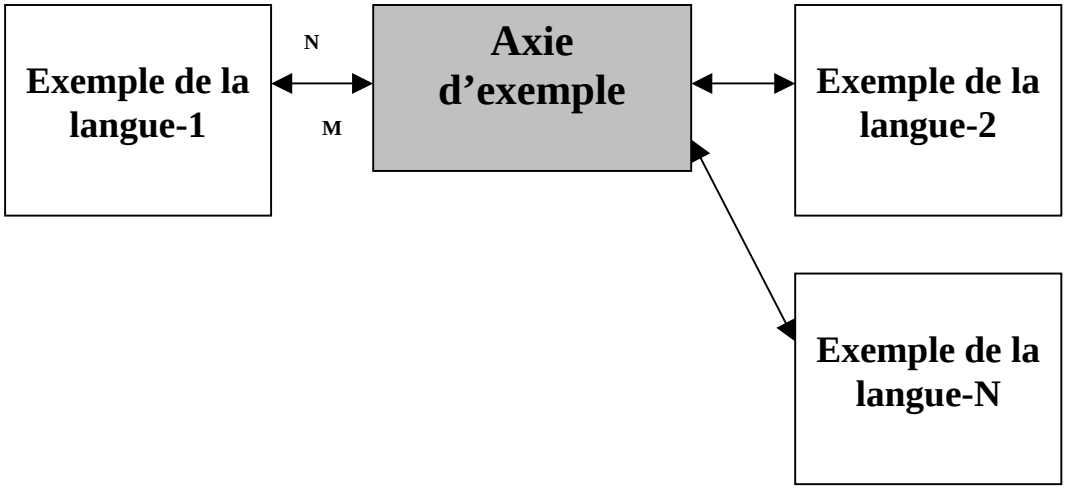
Cette axie ne porte aucun attribut.

3) **L’axie d’exemple** qui permet de documenter la traduction des exemples. Les exemples de sens peuvent être traduits mais ce n’est pas une obligation. Dans la mesure où on peut avoir plusieurs exemples pour un seul sens, nous avons besoin d’un mécanisme de d’association d’un exemple d’une langue en un exemple d’une autre langue. Quelques fois les exemples font référence à des objets culturels et il est nécessaire de transposer les référence d’une culture à l’autre. En adaptant l’exemple de [Boitet], en français, on dirait : « Pour mes voyages, je fais confiance à l’automobile club de France », alors qu’un américain dirait : « For travelling, I trust the American Automobile Association ».
Cette axie ne porte aucun attribut.

Pour bien insister sur le fait que le système ne s'applique pas seulement à une traduction bilingue, nous représentons un nombre de traduction supérieur à deux. Le fait qu'une langue soit située dans la partie gauche ou dans la partie droite n'a aucune signification particulière. On peut parcourir la structure dans n'importe quel sens.
Par convention, les éléments du système de la traduction sont grisés.







10- Fragments d’exemples de mots

a) Présentation

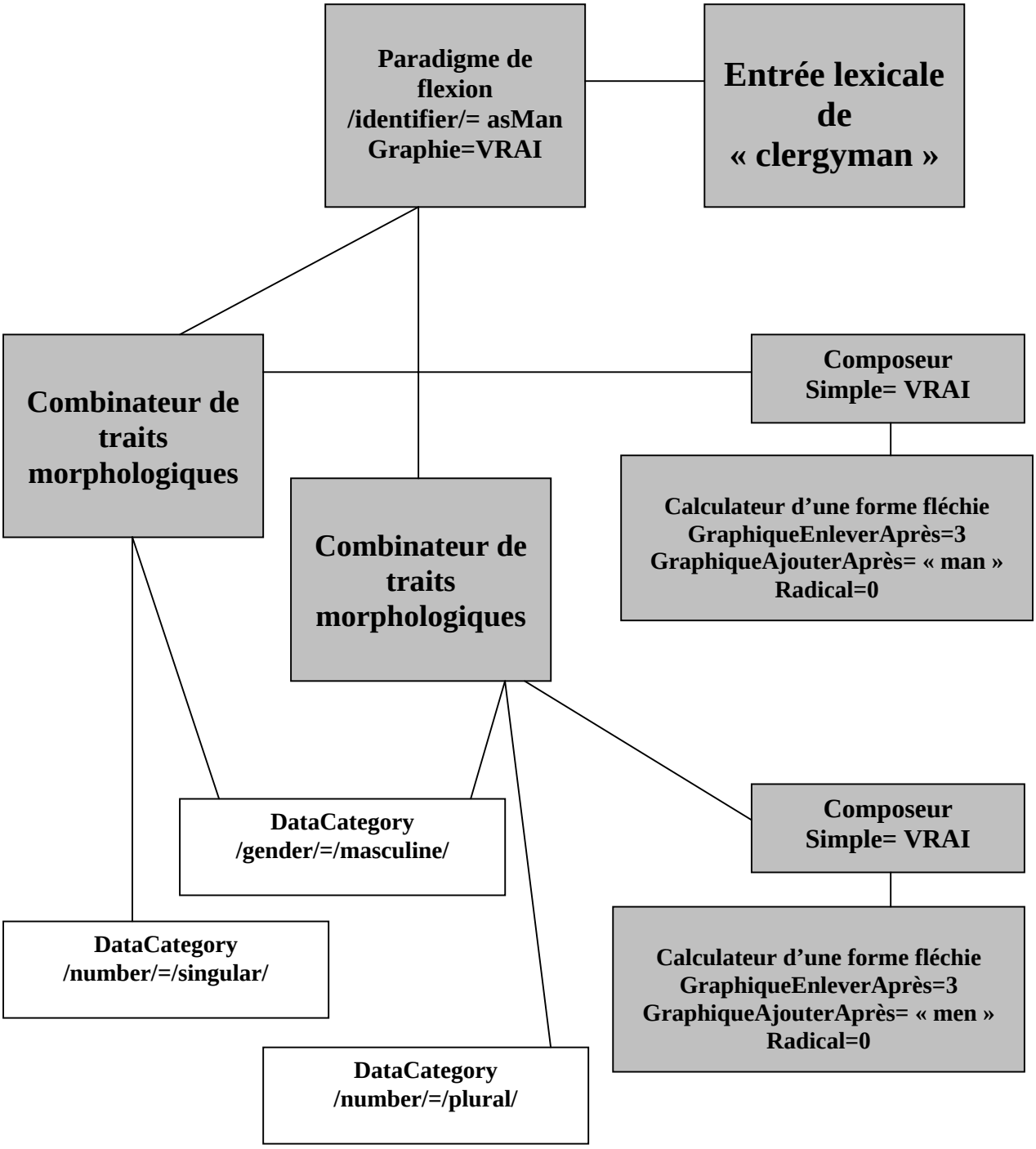
La présentation formelle qui précède n’est pas toujours très facile à comprendre, surtout si vous n’êtes pas familier de la modélisation entités-relations. Pour éclairer l’usage des entités, nous présentons des fragments de descriptions appliquées à des mots connus.

Il s’agit d’une part de vous montrer comment employer les éléments du modèle pour coder des entrées et d’autre part, de vous montrer le pouvoir d’expression du modèle.

Nous prenons la convention que les éléments grisés font partie du modèle et que les éléments blancs sont externes. D’autre part, pour gagner de la place sur les diagrammes, les attributs non informatifs pour l’exemple ne sont pas représentés.

b) La morphologie du mot anglais « clergyman »

C'est un exemple d'application du paradigme à un mot simple. Le singulier est « clergyman » et le pluriel est « clergymen ». Le mode de flexion s'appelle « asMan ». Comme la morphologie de la langue anglaise est relativement simple, nous prenons le parti de rester simple. Donc, nous ne gérons aucun radical et procédons par référence à la forme lemmatisée. La valeur de l'attribut « radical » vaut donc zéro.



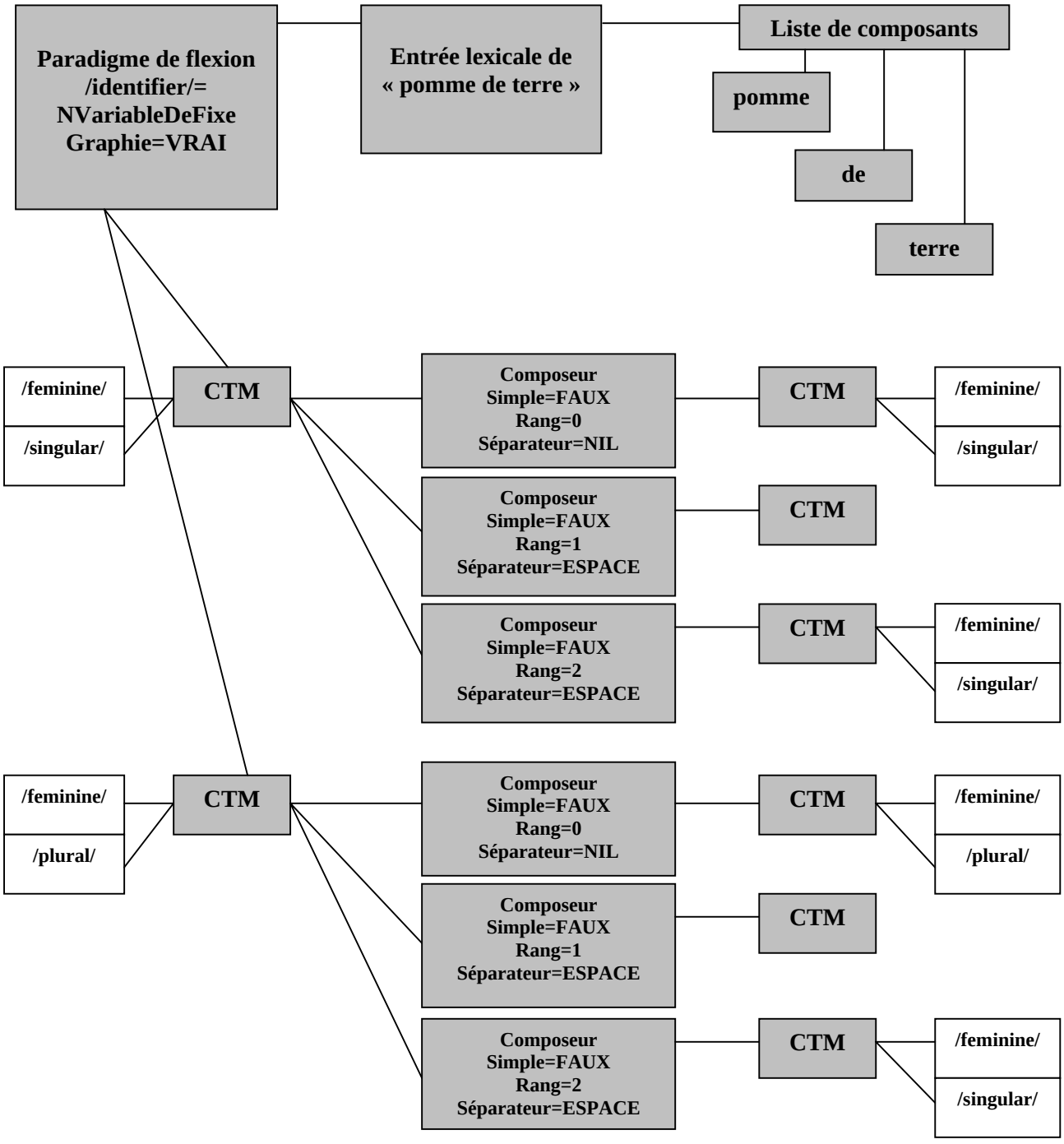
c) La morphologie du mot français « pomme de terre »

Les formes fléchies sont calculées depuis les composants en faisant référence aux combinaisons des traits morphologiques de chacun des composants.

Par convention, et pour gagner de la place sur le diagramme, « Combinateur de traits morphologiques » est abrégé en CTM.

Le singulier de « pomme de terre » est « pomme de terre ». Le pluriel est « pommes de terre ». C'est un comportement très commun en français pour une forme NdeN que d'avoir une variation sur la seule tête du mot composé avec un figement sur la partie modifiante.

Les CTM sur la partie gauche représentent le nombre et le genre du composé. Les CTM de la partie droite représentent le nombre et le genre des composants. La préposition « de » n'est pas liée à une DataDategory car elle n'a aucun trait morphologique.



d) La définition analytique du sens de « défier » à la manière de la BDéf

A titre d'exemple, nous allons représenter le sens de « défierI.1 » tel qu'il est décrit dans [Altman]. Ce sens est issu du DEC-4 et correspond à : « Marcel a défié ce pédant en duel à l'épée ».

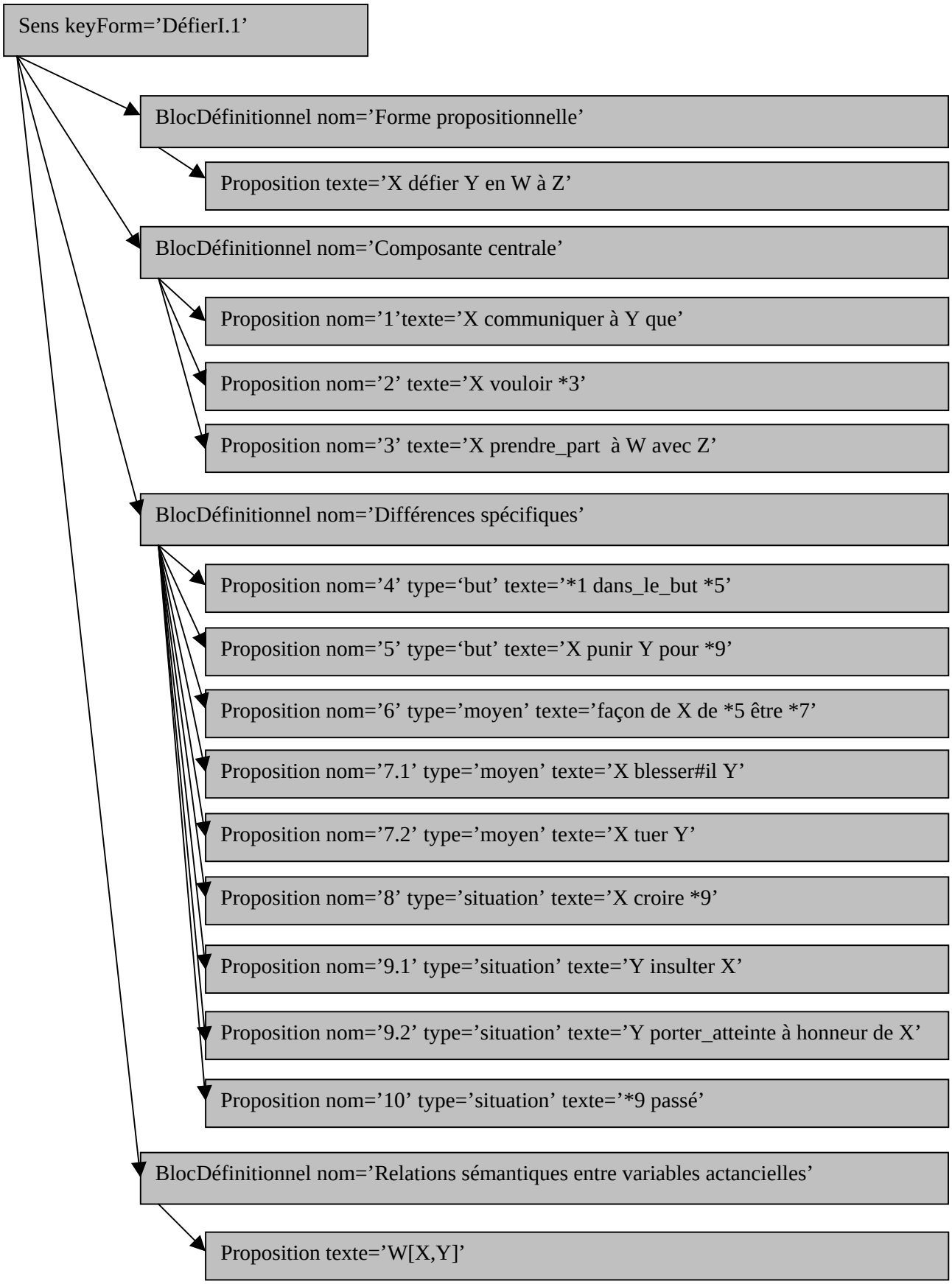
- Forme propositionnelle
  - X défier Y en W à Z
- Composante centrale
  - 1 : X communiquer à Y que *2
  - 2 : X vouloir *3
  - 3 : Y prendre_part à Y avec Z
- Différences spécifiques

  /*but*/
  - 4 : *1 dans_le_but *5
  - 5 : X punir Y pour *9

  /*moyen*/
  - 6 : façon de X de *5 être *7
  - 7.1 : X blesser#il Y
  - 7.2 : X tuer Y

  /*situation*/
  - 8 : X croire *9
  - 9.1 : Y insulter X
  - 9.2 : Y porter_atteinte à honneur de X
  - 10 : *9 passé
- Typage des variables
  - X : individu
  - Y : individu
  - Z : arme
  - W : combat
- Relations sémantiques entre actants
  - W[X,Y]

Au lieu d'utiliser un bloc pour représenter le typage des variables, nous allons utiliser la liste des actants sémantiques. Donc au lieu d'avoir cinq blocs, nous n'aurons que quatre blocs qui seront structurés de la façon suivante :


Sens keyForm=’DéfierI.1’
BlocDéfinitionnel nom=’Forme propositionnelle’
Proposition texte=’X défier Y en W à Z’
BlocDéfinitionnel nom=’Composante centrale’
Proposition nom=’1’texte=’X communiquer à Y que’
Proposition nom=’2’ texte=’X vouloir *3’
Proposition nom=’3’ texte=’X prendre_part à W avec Z’
BlocDéfinitionnel nom=’Différences spécifiques’
Proposition nom=’4’ type=‘but’ texte=’*1 dans_le_but *5’
Proposition nom=’5’ type=‘but’ texte=’X punir Y pour *9’
Proposition nom=’6’ type=’moyen’ texte=’façon de X de *5 être *7’
Proposition nom=’7.1’ type=’moyen’ texte=’X blesser#il Y’
Proposition nom=’7.2’ type=’moyen’ texte=’X tuer Y’
Proposition nom=’8’ type=’situation’ texte=’X croire *9’
Proposition nom=’9.1’ type=’situation’ texte=’Y insulter X’
Proposition nom=’9.2’ type=’situation’ texte=’Y porter_atteinte à honneur de X’
Proposition nom=’10’ type=’situation’ texte=’*9 passé’
BlocDéfinitionnel nom=’Relations sémantiques entre variables actancielles’
Proposition texte=’W[X,Y]’

e) La syntaxe du sens « Aider1 » à la manière du DEC

L'exemple est issu du chapitre : « zone de combinatoire syntaxique » de [Mel'cuk 95].

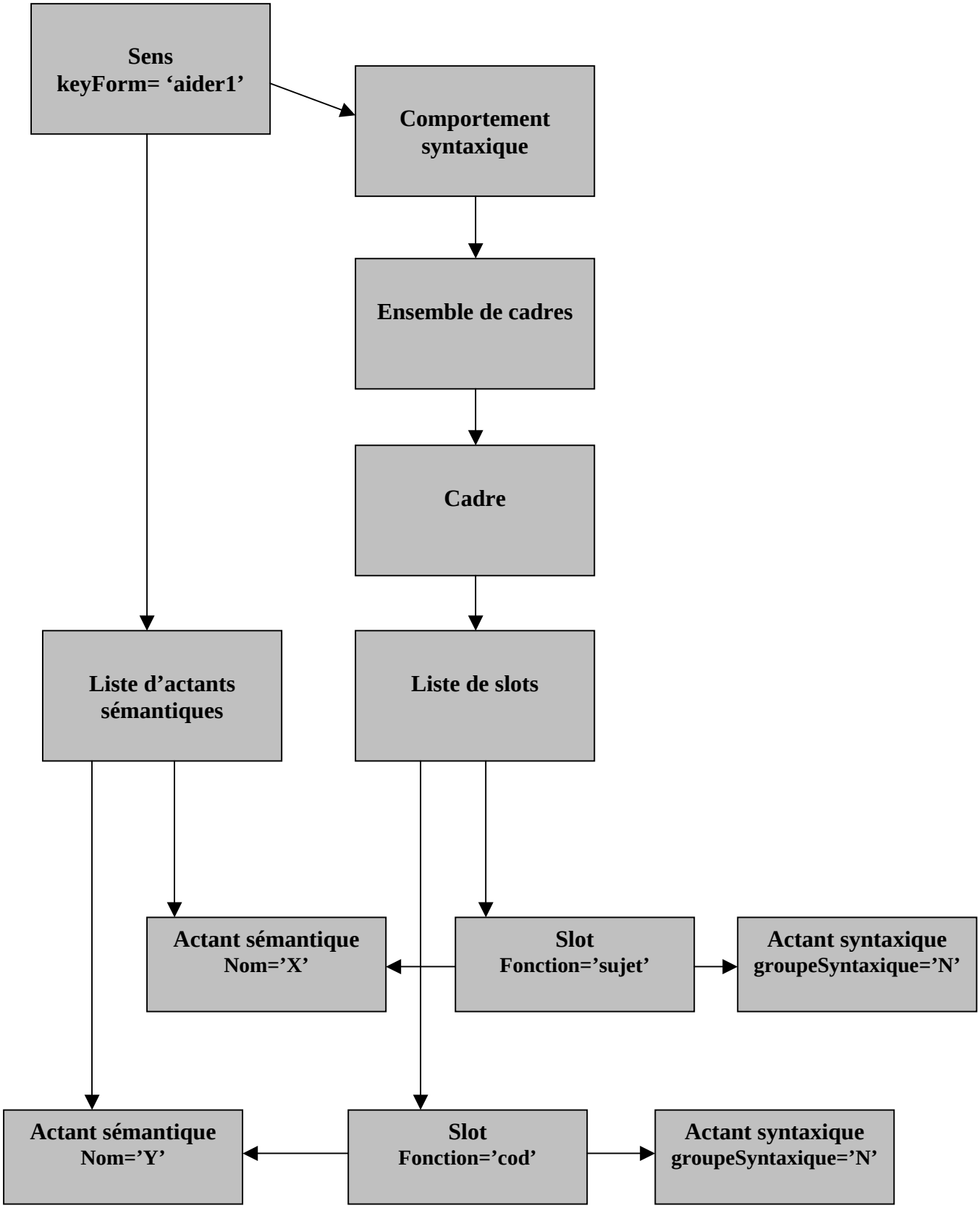
Le DEC décompose la syntaxe en deux parties : la syntaxe de surface et la syntaxe profonde. Les informations de la syntaxe de surface sont très simples, ce sont les marques des fonctions comme le sujet ou le complément d'objet. Ces informations peuvent être représentées dans l'attribut « fonction » du slot. La syntaxe profonde est un peu plus complexe, elle s'articule autour de la notion de « régime ». Un régime est une table dans lequel chaque colonne correspond à un actant sémantique. La notion de ligne dans le tableau n'a aucune importance. Ce qui importe, c'est la cellule. Chaque cellule décrit la réalisation possible d'un groupe syntaxique avec les indications concernant la préposition introductive et le type de groupe syntaxique. Ainsi, nous aurons : « pour Vinf » afin d'indiquer que la groupe infinif commencera par la proposition « pour ». A côté du tableau, le DEC définit les combinaisons possibles entre les cellules.

Dans notre modèle, le régime est un ensemble de cadres. Un cadre est une combinaison possible de cellules. Un slot est une cellule entrant dans une combinaison autorisée.

Pour en revenir au sens « Aider1 », nous aurons quatre actants sémantiques : « X aide Y à Z-er par W » comme dans : « il vous aidera par son intervention à surmonter cette épreuve ».

Nous aurons huit cadres :
Cadre-1 : La Grande-Bretagne aide ses voisins.
Cadre-2 : La Grande-Bretagne a aidé à la création de l'ONU.
Cadre-3 : La Grande-Bretagne a aidé dans toutes les activités de l'ONU.
Cadre-4 : La Grande-Bretagne a aidé l'ONU pour la réussite de ce projet.
Cadre-5 : La Grande-Bretagne a aidé l'ONU pour la réussite de ce projet.
Cadre-6 : La Grande Bretagne a aidé l'ONU avec ses activités.
Cadre-7 : La Grande-Bretagne a aidé à la création de l'ONU.
Cadre-8 : La Grande-Bretagne a aidé les autres pays à créer l'ONU.

Nous n’allons représenter que le cadre-1 : « La Grande-Bretagne aide ses voisins ».



11- Synthèse

La lexicographie éditoriale a pour tradition de considérer qu'un dictionnaire se définit selon trois critères [Rey-Debove] :

- la nomenclature : quels sont les mots du dictionnaire ?
- la microstructure : l'organisation d'un article qui se répète de façon systématique,
- la macrostructure : l'organisation de l'ensemble du dictionnaire.

Pour notre modèle, c'est un peu différent car nous avons un axe supplémentaire qui est celui des constantes linguistiques. Ce sont des éléments qui sont caractéristiques d'une langue et qui sont utilisés par un grand nombre d'éléments.
La microstructure est constituée de l'entrée lexicale et du sens.
La macrostructure est constituée des systèmes sémantiques (sauf type de relation), syntaxiques et des traductions.
Les constantes linguistiques sont les éléments de la morphologie, le type de relation ainsi que l'information globale.

12- Complétude de la description d'un mot

Le travail sur un dictionnaire n'est jamais fini : il y a toujours certaines zones en chantier. Le dictionnaire peut être dans un état plus ou moins stable, ou bien plus ou moins présentable.
Le modèle présenté ici n'est pas destiné à un dictionnaire idéal qui n'existe pas, mais au contraire à un dictionnaire réel. Et ceci que ce soit pour la gestion des entrées ou bien pour l'échange de données.

Les cardinalités entre éléments indiquent que le degré de renseignement d'un mot est très souple. Un mot peut être plus ou moins décrit. Il peut être en cours de description tout en étant conforme au modèle.
Néanmoins, pour distinguer un mot partiellement décrit d'un mot complètement décrit, nous posons la définition suivante :
Un mot est complètement décrit si et seulement si :

- l'entrée lexicale a les champs renseignés,
- l'entrée lexicale a une morphologie,
- l'entrée lexicale est reliée à au moins un sens,
- chacun de ces sens a au moins un comportement syntaxique,
- chacun de ces comportements syntaxiques est décrit par un régime,
- pour chacune des langues du système, chaque sens du mot a une axie,
- chacune de ces axies n'est reliée qu'à un seul sens.

13- Connexion avec TMF

TMF est un méta-modèle dédié à la terminologie et décrit par la norme ISO-16642 [Romary].
Il est souhaitable qu'un utilisateur de TMF puisse associer des outils de TAL avec son thésaurus. La passerelle ne peut être une bijection dans la mesure où le thésaurus ne comporte que des noms, et seulement les noms du domaine. En fait, seule une sous-partie du dictionnaire général pourra avoir un correspondant dans TMF. La correspondance pourra se faire entre « sens » et « TermSection ».
[à creuser]

14- Retroconversion d'OLIF-2

Notre modèle est englobant par rapport à OLIF-2.
Il est possible d'exprimer OLIF-2 dans notre modèle via une feuille de style XSLT.
[à faire : écrire la feuille de style pour le prouver]

15- Traduction des éléments de français en anglais

Français	Anglais
actant sémantique	semantic actant
actant syntaxique	syntactic actant
axie de sens	senseAxie
axie de transfert	transferAxie
axie d'exemple	exampleAxie
bloc définitionnel	definition block
cadre	frame
calculateur d'une forme fléchie	inflected form calculator
combinateur de traits morphologiques	morphological features combiner
commentaire	comment
comportement syntaxique	syntactic behavior
composeur	composer
définition	definition
exemple	example
ensemble de cadres	frame set
entrée lexicale	lexical entry
forme fléchie	inflected form
information globale	global information
liste d'actants sémantiques	list of semantic actant
liste de composants	list of components
liste de radicaux	list of stems
liste de slots	list of slots
note d'usage	usage note
paradigme de flexion	inflectional category
proposition	proposition
radical	stem
relation	relation
relation d'axie	axie relation
sens	sense
slot	slot
type de relation	relationType

16- DTD XML du modèle

[à discuter :

- au lieu de mettre la langue sur chaque lexicalEntry, on peut la mettre dans globalInformation, (c'est ce que j'ai fait pour l'instant)
- idem pour script
- question : inflectedForm serait peut-être plus à sa place avec lexicalEntry ?]

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!-- a lexical data base may contains zero or several
    monolingual lexicons and possibly a translation system -->
<!ELEMENT lexicalDataBase (monolingual*, translationSystem?) >
<!ELEMENT monolingual (globalInformation,
                       lexicalEntrySystem?,
                       morphologicalSystem?,
                       syntacticSystem?,
                       semanticSystem?) >
<!ELEMENT globalInformation EMPTY>
<!ATTLIST globalInformation
   id                    ID              #IMPLIED
```

```
   dtdVersion             CDATA           #FIXED       '1.2'
   name                   CDATA           #REQUIRED
   title                  CDATA           #IMPLIED
   subTitle               CDATA           #IMPLIED
   comment                CDATA           #IMPLIED
   language               CDATA           #REQUIRED
   script                 CDATA           #IMPLIED
   dataVersion            CDATA           #IMPLIED
   creator                CDATA           #IMPLIED
   creationDate           CDATA           #IMPLIED
   modificationDate       CDATA           #IMPLIED
   copyright              CDATA           #IMPLIED>
<!--=================================================-->
<!ELEMENT lexicalEntrySystem (lexicalEntry*)>
<!ELEMENT lexicalEntry (radical*)>
<!ATTLIST lexicalEntry
   id                     ID              #REQUIRED
   pos                    CDATA           #REQUIRED
   lemmatizedForm         CDATA           #REQUIRED
   autonomy               (yes|no)        "yes"
   phoneticForm           CDATA           #IMPLIED
   components             IDREFS          #IMPLIED
   inflectionalCategory   IDREF           #IMPLIED
   inflectedForms         IDREFS          #IMPLIED
   senses                 IDREFS          #IMPLIED>
<!ELEMENT radical EMPTY>
<!ATTLIST radical
   id                     ID              #IMPLIED
   string                 CDATA           #REQUIRED>
<!--=================================================-->
<!ELEMENT sense (definition*, comment*, example*, usageNote*)>
<!ATTLIST sense
   id                     ID              #REQUIRED
   keyForm                CDATA           #REQUIRED
   usageMarks             CDATA           #IMPLIED
   syntacticBehaviors     IDREFS          #IMPLIED
   semanticActants        IDREFS          #IMPLIED
   relations              IDREFS          #IMPLIED
   definitionBlocks       IDREFS          #IMPLIED
   senseAxies             IDREFS          #IMPLIED>
<!ELEMENT definition EMPTY>
<!ATTLIST definition
   id                     ID              #IMPLIED
   text                   CDATA           #REQUIRED
   lang                   CDATA           #REQUIRED>
<!ELEMENT comment EMPTY>
<!ATTLIST comment
   id                     ID              #IMPLIED
   text                   CDATA           #REQUIRED
   lang                   CDATA           #REQUIRED>
<!ELEMENT example EMPTY>
<!ATTLIST example
   id                     ID              #IMPLIED
   text                   CDATA           #REQUIRED
   lang                   CDATA           #REQUIRED
   exampleAxies           IDREFS          #IMPLIED>
<!ELEMENT usageNote EMPTY>
<!ATTLIST usageNote
   id                     ID              #IMPLIED
   text                   CDATA           #REQUIRED
   lang                   CDATA           #REQUIRED>
```

```
<!--=====================================================-->
<!ELEMENT morphologicalSystem (inflectionalCategory*,
                               inflectedForm*)>
<!ELEMENT inflectionalCategory (morphologicalFeaturesCombiner*)        >
<!ATTLIST inflectionalCategory
   id                     ID              #REQUIRED
   example                CDATA           #IMPLIED
   comment                CDATA           #IMPLIED
   pos                    CDATA           #REQUIRED
   graphical              (yes|no)        "yes"
   phonetical             (yes|no)        "no">
<!ELEMENT morphologicalFeaturesCombiner (composer*)>
<!ATTLIST morphologicalFeaturesCombiner
   id                     ID              #IMPLIED
   features               CDATA           #REQUIRED>
<!ELEMENT composer (inflectedFormCalculator*,
                    morphologicalFeaturesCombiner*)>
<!ATTLIST composer
   id                     ID              #IMPLIED
   type                   (simple|compound) "simple"
   rank                   CDATA           #IMPLIED
   separator              CDATA           #IMPLIED>
<!ELEMENT inflectedFormCalculator EMPTY>
<!ATTLIST inflectedFormCalculator
   id                     ID              #IMPLIED
   radical                CDATA           #REQUIRED
   graphicRemoveBefore    CDATA           #IMPLIED
   graphicAddBefore       CDATA           #IMPLIED
   graphicRemoveAfter     CDATA           #IMPLIED
   graphicAddAfter        CDATA           #IMPLIED
   graphicSubstitute      CDATA           #IMPLIED
   graphicSubstituteIn    CDATA           #IMPLIED
   graphicSubstituteBy    CDATA           #IMPLIED
   graphicMove            CDATA           #IMPLIED
   graphicMoveFrom        CDATA           #IMPLIED
   graphicMoveUntil       CDATA           #IMPLIED
   graphicCopy            CDATA           #IMPLIED
   graphicCopyFrom        CDATA           #IMPLIED
   graphicCopyUntil       CDATA           #IMPLIED
   phoneticRemoveBefore   CDATA           #IMPLIED
   phoneticAddBefore      CDATA           #IMPLIED
   phoneticRemoveAfter    CDATA           #IMPLIED
   phoneticAddAfter       CDATA           #IMPLIED
   phoneticlSubstitute    CDATA           #IMPLIED
   phoneticSubstituteIn   CDATA           #IMPLIED
   phoneticSubstituteBy   CDATA           #IMPLIED
   phoneticMove           CDATA           #IMPLIED
   phoneticMoveFrom       CDATA           #IMPLIED
   phoneticMoveUntil      CDATA           #IMPLIED
   phoneticCopy           CDATA           #IMPLIED
   phoneticCopyFrom       CDATA           #IMPLIED
   phoneticCopyUntil      CDATA           #IMPLIED
   contextVariation       CDATA           #IMPLIED>
<!ELEMENT inflectedForm EMPTY>
<!ATTLIST inflectedForm
   id                     ID              #REQUIRED
   form                   CDATA           #REQUIRED
   features               CDATA           #REQUIRED>
<!--=====================================================-->
<!ELEMENT syntacticSystem (syntacticBehavior*, frameSet*, frame*,
          slot*, syntacticActant*, semanticActant*)>
```

```
<!ELEMENT syntacticBehavior EMPTY>
<!ATTLIST syntacticBehavior
   id                    ID             #IMPLIED
   frameSet              IDREF          #REQUIRED
   transferAxies         IDREFS         #IMPLIED>
<!ELEMENT frameSet  EMPTY>
<!ATTLIST frameSet
   id                    ID             #REQUIRED
   frames                IDREFS         #REQUIRED>
<!ELEMENT frame EMPTY>
<!ATTLIST frame
   id                    ID             #REQUIRED
   comment               CDATA          #IMPLIED
   slots                 IDREFS         #REQUIRED>
<!ELEMENT slot EMPTY>
<!ATTLIST slot
   id                    ID             #REQUIRED
   function              CDATA          #IMPLIED
   semanticActant        IDREF          #IMPLIED
   syntacticActant       IDREF          #REQUIRED>
<!ELEMENT syntacticActant EMPTY>
<!ATTLIST syntacticActant
   id                    ID             #REQUIRED
   definition            CDATA          #IMPLIED
   comment               CDATA          #IMPLIED
   introducer            CDATA          #IMPLIED
   syntacticGroup        CDATA          #IMPLIED
   restriction           CDATA          #IMPLIED>
<!ELEMENT semanticActant EMPTY>
<!ATTLIST semanticActant
   id                    ID             #REQUIRED
   name                  CDATA          #IMPLIED
   restriction           CDATA          #IMPLIED>
<!--=================================================-->
<!ELEMENT semanticSystem (relation*, relationType*,
                          definitionBlock*)>
<!ELEMENT relation EMPTY>
<!ATTLIST relation
   id                    ID             #REQUIRED
   target                IDREF          #IMPLIED
   relationType          IDREF          #REQUIRED>
<!ELEMENT relationType EMPTY>
<!ATTLIST relationType
   id                    ID             #REQUIRED
   name                  CDATA          #REQUIRED
   comment               CDATA          #IMPLIED
   example               CDATA          #IMPLIED
   isA                   IDREF          #IMPLIED>
<!ELEMENT definitionBlock (proposition*)>
<!ATTLIST definitionBlock
   id                    ID             #REQUIRED
   name                  CDATA          #IMPLIED>
<!ELEMENT proposition EMPTY>
<!ATTLIST proposition
   name                  CDATA          #REQUIRED
   type                  CDATA          #IMPLIED
   text                  CDATA          #IMPLIED>
<!--=================================================-->
<!ELEMENT translationSystem (senseAxie*, transferAxie*,
                             exampleAxie*)>
<!ELEMENT senseAxie (axieRelation?)>
```

```
<!ATTLIST senseAxie
   id                    ID              #REQUIRED>
<!ELEMENT axieRelation EMPTY>
<!ATTLIST axieRelation
   id                    ID              #IMPLIED
   title                 CDATA           #REQUIRED
   externalSystem        CDATA           #IMPLIED
   externalRef           CDATA           #IMPLIED>
<!ELEMENT transferAxie EMPTY>
<!ATTLIST transferAxie
   id                    ID              #REQUIRED>
<!ELEMENT exampleAxie EMPTY>
<!ATTLIST exampleAxie
   id                    ID              #REQUIRED>
```

17- Schéma XML du modèle

Le schéma XML est plus précis que la DTD dans la mesure où l'on peut typer les liens et ainsi rendre compte plus fidèlement du modèle conceptuel. En revanche, il est plus verbeux et le nombre de logiciels capable d'en faire usage est très limité.
[à faire quand le diagramme sera stabilisé, voir si on lui préfère un descriptif Relax-NG]

18- Références bibliographiques d'articles

Altman J., Polguère A. 2003
La Bdéf : base de définitions dérivée du dictionnaire explicatif et combinatoire. MTT Paris.
Antoni-Lay MH., Francopoulo G., Zaysser L. 1994
A generic model for reusable lexicons : The Genelex project. Literary and Linguistic Computing.
Boguslavsky I. 2002
Some lexical issues of UNL. LREC.
Boitet C. 2002
The translation of examples, citations, definitions and glosses in the Papillon project. Journées Papillon. Tokyo.
Calzolari N., Fillmore C., Grishman R., Ide N., Lenci A., MacLeod C., Zampolli A. 2002
Towards best practice for Multiword Expressions in Computational Lexicons. LREC.
Dérouin MJ., Le Meur A. 2002.
Report on the revision of the lexicographical Standard ISO 1951. LREC.
Fellbaum C. 1998
WordNet : an electronic lexical database. MIT Press Cambridge, Mass
Fradin B. 2003
Nouvelles approches en morphologie. PUF. Paris.
Gaudin L., Guespin F. 2000
Initiation à la lexicologie française : de la néologie aux dictionnaires. Duculot. Bruxelles.
Genelex (consortium)
Rapport sur la couche morphologique 1991, syntaxique 1993, sémantique 1994.
Mangeot M. 2001
Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Thèse de nouveau doctorat. Grenoble I.
Mangeot M., Sérasset G. 2001
Papillon lexical databases project : monolingual dictionaries and interlingual links. NLPRS. Tokyo.
Mel'cuk & al. 1984, 1988, 1992, 1999
Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques. Volumes 1, 2, 3,4. Presses de l'université de Montréal.
Mel'cuk I. 2000
Cours de morphologie générale. Volume-5. Presses de l'université de Montréal.

Mel'cuk I., Clas A., Polguère A. 1995
Introduction à la lexicologie explicative et combinatoire. Duculot. Bruxelles.
Peters C., Braschler M., Gonzalo J., Kluck M. 2002
Advances in Cross-Language Information Retrieval. Springer. Berlin.
Polguère A. 2000
Towards a theorically motivated general public dictionary of semantic derivation and collocation for French. Euralex 2000.
Pruvost J. 2000
Dictionnaires et nouvelles technologies. PUF. Paris.
Rey-Debove J. 1971
Etude linguistique et sémiotique des dictionnaires français contemporains. Mouton. The Hague.
Romary L. 2001
Towards an Abstract Representation of Terminological Data Collections – the TMF model. TAMA. Antwerp.
Saussure (de) F. 1974
Cours de linguistique générale : édition critique de Tullio de Mauro. Payot. Paris.
Silberztein M. 1993
Dictionnaires électroniques et analyse automatique de textes. Masson. Paris.

19- Références de sites comportant de multiples articles

EAGLES	www.ilc.cnr.it/EAGLES96/home.html
EDR/CRL	www2.crl.go.jp/kk/e416/EDR
Papillon	www.papillon-dictionary.org
Relax NG	www.oasis-open.org/committees/tc_home.php?wg_abbrev=relax-ng
Relex	www-igm.univ-mlv.fr/~unitex/linguistic_data.html
TEI	www.tei-c.org/P4X
UNL	www.unl.ias.unu.edu
WordNet	www.globalwordnet.org

20- Conclusion

Nous n'avons pas réinventé la roue. Nous avons pris les meilleures idées parmi les modèles que nous connaissions. Ces mécanismes descriptifs ont fait leurs preuves car ils sont mis en pratique depuis plusieurs années dans des équipes lexicographiques opérationnelles. Il ne faut pas considérer le présent travail comme une activité de recherche : il faut plutôt le voir comme un retour d'expérience et de pratique des grands projets lexicographiques des années 90. Nous pensons que la lexicographie pour le TAL a suffisamment mûri pour que des pratiques consensuelles se dégagent.
De plus, chaque fois que c'était possible, nous avons utilisé les normes existantes comme les registres de catégories de données ISO 12620 ou Unicode.
Mais surtout, nous avons résisté à la tentation de la complexité inutile. Et, en effet, **tout en étant puissant, le modèle reste simple**.