

# Manuel de l'utilisateur

## Démonette version 1.2

### Présentation de la base

Nabil Hathout      Fiammetta Namer  
Université de Toulouse      Université de Lorraine  
& CLLE      & ATILF

2 décembre 2015

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Les champs</b>	<b>3</b>
2.1	Identité de l'entrée . . . . .	3
2.2	Typage de la relation (Mot <sub>1</sub> , Mot <sub>2</sub> ) . . . . .	3
2.2.1	Orientation . . . . .	3
2.2.2	Complexité dérivationnelle . . . . .	4
2.2.3	Synthèse . . . . .	5
2.3	Relation formelle (Mot <sub>1</sub> , Mot <sub>2</sub> ) . . . . .	6
2.3.1	Cas canonique . . . . .	6
2.3.2	Rétroformation . . . . .	7
2.3.3	Conversion . . . . .	8
2.4	Propriétés sémantiques de (Mot <sub>1</sub> , Mot <sub>2</sub> ) . . . . .	8
2.4.1	Définition concrète de Mot <sub>1</sub> . . . . .	8
2.4.2	Type sémantique de Mot <sub>1</sub> . . . . .	9
2.4.3	Définition abstraite de Mot <sub>1</sub> . . . . .	9
<b>3</b>	<b>Les définitions abstraites</b>	<b>10</b>
<b>4</b>	<b>Liste complète des champs pour le format csv</b>	<b>11</b>
<b>5</b>	<b>Exemple XML</b>	<b>13</b>

## 1 Introduction

La base Démonette est une ressource lexicale dérivationnelle destinée à :

- recevoir toute sorte de bases de données morphologiques munies d'informations diverses (catégorielles, sémantiques, définitoires, dérivationnelles, flexionnelles, phonétiques) provenant de sources variées,
- les représenter dans un format unifié, présenté dans ce qui suit,
- calculer pour certaines d'entre elles des informations morphosémantiques nouvelles, inférables à partir des propriétés fournies.

Démonette est un réseau composé de relations morphologiques dérivationnelles.

Une entrée de Démonette décrit ainsi une relation entre deux mots : Mot<sub>1</sub> et Mot<sub>2</sub>. Un mot (i.e. un **lexème** dans la terminologie morphologique standard) est caractérisé par les trois types d'informations fondamentales qui composent un lexème, à savoir des informations formelles, catégorielles et sémantiques. Plus précisément, un lexème Mot<sub>i</sub> est identifié de manière unique par un triplet (Graphie<sub>i</sub>, Cat<sub>i</sub>, Type<sub>i</sub>) où :

- Graphie<sub>i</sub> est la graphie de sa forme de citation (i.e. son lemme dans la terminologie du TAL),
- Cat<sub>i</sub> est l'étiquette morphosyntaxique de sa forme de citation (catégorie et traits flexionnels), et
- Type<sub>i</sub> est un type sémantique décrit plus en détail en section 2.4.2.

*Exemple :* Le tableau suivant présente quelques exemples. Les indices des lexèmes permettent de distinguer les deux types d'emplois des noms déverbaux en *-ion*, *-age*, *-ment*, etc. : noms d'action (*a*) des noms de résultat (*r*).

Lexème	Graphie	Catégorie	Type sémantique
BONIFICATION <sub>a</sub>	bonification	Ncmf	@ACT
BONIFICATION <sub>r</sub>	bonification	Ncmf	@RES
CARRELAGE <sub>a</sub>	carrelage	Ncms	@ACT
CARRELAGE <sub>r</sub>	carrelage	Ncms	@RES
DANSEUR	danseur	Ncms	@AGM
DISTINCTIF	distinctif	Afpms	@PRP

Une entrée dans Démonette est par conséquent un sextuplet (Graphie<sub>1</sub>, Cat<sub>1</sub>, Type<sub>1</sub>, Graphie<sub>2</sub>, Cat<sub>2</sub>, Type<sub>2</sub>).

Démonette est une ressource **redondante** qui fournit autant de descriptions morphologiques pour un mot qu'il y a de relations dans lesquelles ce mot intervient. Un mot Mot<sub>i</sub> = (Graphie<sub>i</sub>, Cat<sub>i</sub>, Type<sub>i</sub>) peut en effet intervenir dans plusieurs relations morphologiques avec les membres de sa famille constructionnelle. Ces relations peuvent être directes ou indirectes, ascendantes ou descendantes, simples, complexes ou lexicales. Une caractéristique originale de Démonette est qu'elle fournit à chaque Mot<sub>i</sub> une définition morphosémantique pour chacune des relations où il apparaît.

La conception de Démonette est **symétrique**. À chaque entrée reliant Mot<sub>1</sub> à Mot<sub>2</sub>, correspond une entrée reliant Mot<sub>2</sub> à Mot<sub>1</sub>, dont les propriétés sont définies symétriquement par rapport à (Mot<sub>1</sub>, Mot<sub>2</sub>).

La source d'**origine** de chaque information codée dans Démonette est systématiquement précisée : il s'agit soit d'une information native, i.e. en provenance de la source d'origine des données, soit d'une information produite lors de l'intégration de l'entrée dans Démonette.

La version distribuée (version 1.2) contient le résultat de la contribution de trois ressources et systèmes d'analyse morphologique : Morphonette, DériF et Verbaction, obéissant chacun à des traditions linguistiques différentes, et produisant des informations soit redondantes, soit complémentaires. Elle comporte 96 027 entrées qui décrivent des relations entre 61 822 couples différents de lexèmes. Les couples présents dans plusieurs ressources sont décrit plusieurs fois (7 816 sont décrites dans les 3 ressources, 18 573 dans 2 ressources et 35 433 dans une ressource).

Les relations morphologiques représentées sont les instances des dérivations déverbaux permettant de construire

1. des noms d'action en *-ade* (*bousculade*), *-age* (*abordage*), *-aille* (*retrouaille*), *-aire* \* (*commentaire*), *-aison* (*combinaison*), *-ance* (*accoutumance*), *-ande* \* (*offrande*), *-ange* \* (*louange*), *-ée* (*percée*), *-ence* (*adhérence*), *-erie* (*cajolerie*), *-et* (*ricochet*), *-ette* (*tremlette*), *-eur* \* (*erreur*), *-ice* \* (*exercice*), *-ie* (*garantie*), *-if* \* (*expectative*), *-ing* (*kidnapping*), *-ion* (*labialisation*), *-is* (*arrachis*), *-ise* (*chopardise*), *-isme* (*exorcisme*), *-ité* (*mendicité*), *-ment* (*miaulement*), *-oire* \* (*interrogatoire*), *-on* (*plongeon*) et *-ure* (*brisure*);<sup>1</sup>
2. de résultat en *-ment*, en *-age* et en *-ion*;<sup>2</sup>
3. les noms d'agent en *-eur* (*chialeur*), en *-euse* (*chipoteuse*) et en *-rice* (*clarificatrice*);
4. les adjectifs en *-if* (*cognitif*);
5. les verbes de base correspondants à ces dérivés et des verbes dérivés ou apparentés en *-iser* (*exorciser*), *-onner* (*sautillonner*) et *-oyer* (*guerroyer*).

1. Les suffixes suivis d'une étoile n'apparaissent que dans des relations lexicales.

2. Les noms issus de Verbaction n'ont été considérés que comme des noms d'action car ce lexique ne décrit que cet emploi. À l'inverse, les noms d'action issus de DériF et Morphonette sont à la fois noms d'action et noms de résultat.

## 2 Les champs

La version 1.2 de Démonette organise les propriétés de chaque entrée de la base en 31 champs, que l'on peut regrouper en 4 rubriques :

### 2.1 Identité de l'entrée

Cette rubrique comporte le sextuplet :

(Graphie <sub>1</sub> , Cat <sub>1</sub> , Type <sub>1</sub> , Graphie <sub>2</sub> , Cat <sub>2</sub> , Type <sub>2</sub> )
--

et décrit de façon univoque la relation morphologique entre Mot<sub>1</sub> et Mot<sub>2</sub> :

- Graphie<sub>1</sub> et Graphie<sub>2</sub> sont les lemmes de Mot<sub>1</sub> et Mot<sub>2</sub> (i.e. les formes de citation des deux lexèmes) ;
- Cat<sub>1</sub> et Cat<sub>2</sub> décrivent, respectivement, la catégorie grammaticale et les traits morphosyntaxiques (ou flexionnels) des lexèmes Mot<sub>1</sub> et Mot<sub>2</sub> au format Grace-Multext (Rajman *et al.*, 1997).
- Type<sub>1</sub> (resp. Type<sub>2</sub>) est le type sémantique de Mot<sub>1</sub> (resp. Mot<sub>2</sub>). Type<sub>1</sub> se déduit de la relation morphologique de Mot<sub>1</sub> relativement à Mot<sub>2</sub> (resp. Mot<sub>2</sub> relativement à Mot<sub>1</sub>).

<i>Exemple</i> : La relation entre les lexèmes ABOUTIR et ABOUTISSEMENT est identifiée par les informations suivantes (le symbole @ désigne le type sémantique 'prédicat', et le symbole @ACT, le type 'activité', cf. section 2.4.2) : (aboutir, Vmn---, @, aboutissement, Ncms, @ACT)
--

Nous utilisons des étiquettes morphosyntaxiques complètes pour conserver une compatibilité avec les lexiques TLFnome et GLÀFF.

- L'origine lexicographique de Graphie<sub>1</sub>, Graphie<sub>2</sub>, Cat<sub>1</sub> et Cat<sub>2</sub> est représentée par les champs Ori\_Graphie<sub>1</sub>, Ori\_Graphie<sub>2</sub>, Ori\_Cat<sub>1</sub>, Ori\_Cat<sub>2</sub> respectivement. Dans la version 1.2, les valeurs possibles sont tlfnome, verbaction (DériF et Morphonette fondent tous deux leurs analyses sur tlfnome ; Verbaction a été initialement construit à partir de TLFnome, mais contient également des mots issus du Web).
- Type<sub>1</sub> et Type<sub>2</sub> ayant été calculés lors de la création de Démonette, les valeurs des champs Ori\_Type<sub>1</sub> et Ori\_Type<sub>2</sub> sont demonette. Les types sémantiques sont présentés en section 2.4.2.
- Toutes les relations sont bi-orientées (i.e. décrites dans les 2 orientations), ce qui fait qu'au sextuplet (Graphie<sub>1</sub>, Cat<sub>1</sub>, Type<sub>1</sub>, Graphie<sub>2</sub>, Cat<sub>2</sub>, Type<sub>2</sub>) correspond systématiquement une autre entrée dans Démonette caractérisée par : (Graphie<sub>2</sub>, Cat<sub>2</sub>, Type<sub>2</sub>, Graphie<sub>1</sub>, Cat<sub>1</sub>, Type<sub>1</sub>).

### 2.2 Typage de la relation (Mot<sub>1</sub>, Mot<sub>2</sub>)

La relation morphologique de Mot<sub>1</sub> et Mot<sub>2</sub> est caractérisée par deux propriétés : son orientation et sa complexité dérivationnelle. Les champs dont il est question dans cette rubrique sont donc :

(Orientation, Complexite)
---------------------------

#### 2.2.1 Orientation

Démonette prévoit 3 types d'orientation (décrites dans le champ Orientation) : descendant, ascendant et indirect.

— Si Mot<sub>1</sub> est un descendant de Mot<sub>2</sub>, l'orientation de la relation (Mot<sub>1</sub>, Mot<sub>2</sub>) est descendant, et celle de la relation (Mot<sub>2</sub>, Mot<sub>1</sub>) est ascendant.

Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Orientation
aboutissement	Ncms	@ACT	aboutir	Vmn----	@	descendant
aboutir	Vmn----	@	aboutissement	Ncms	@ACT	ascendant

— Dans les cas où Mot<sub>1</sub> est construit à partir de Mot<sub>2</sub> par **rétroformation**, Mot<sub>2</sub> est décrit comme le porteur de l’affixe et Mot<sub>1</sub> comme le mot sans l’affixe. L’orientation de la relation (Mot<sub>1</sub>, Mot<sub>2</sub>) est **descendant**, et celle de la relation (Mot<sub>2</sub>, Mot<sub>1</sub>) est **ascendant**. Cette situation est inverse de la configuration canonique où le mot qui porte l’affixe descend de celui qui ne le porte pas.

Exemple :<sup>3</sup>

#### Rétroformation

Graphie <sub>1</sub> /Cat <sub>1</sub> /Type <sub>1</sub>	Graphie <sub>2</sub> /Cat <sub>2</sub> /Type <sub>2</sub>	TypCst <sub>1</sub>	Exp <sub>1</sub>	TypCst <sub>2</sub>	Cst <sub>2</sub>	Orientation
hydroplaner/Vmn----/@	hydroplanage/Ncms/@ACT	–	–	suf	age	descendant
hydroplanage/Ncms/@ACT	hydroplaner/Vmn----/@	suf	age	–	–	ascendant

#### Situation canonique

Graphie <sub>1</sub> /Cat <sub>1</sub> /Type <sub>1</sub>	Graphie <sub>2</sub> /Cat <sub>2</sub> /Type <sub>2</sub>	TypCst <sub>1</sub>	Cst <sub>1</sub>	TypCst <sub>2</sub>	Cst <sub>2</sub>	Orientation
planage/Ncms/@ACT	planer/Vmn----/@	suf	age	–	–	descendant
planer/Vmn----/@	planage/Ncms/@ACT	–	–	suf	age	ascendant

— Si Mot<sub>1</sub> et Mot<sub>2</sub> appartiennent à la même famille morphologique mais ne descendent pas l’un de l’autre, l’orientation des relations (Mot<sub>1</sub>, Mot<sub>2</sub>) et (Mot<sub>2</sub>, Mot<sub>1</sub>) est **indirect**.

Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Orientation
aboyeur	Ncms	@AGM	aboyeuse	Ncfs	@AGF	indirect
aboyeuse	Ncfs	@AGF	aboyeur	Ncms	@AGM	indirect

## 2.2.2 Complexité dérivationnelle

L’autre caractéristique des relations morphologiques est leur complexité identifiée au nombre minimal d’opérations morphologiques élémentaires permettant de connecter Mot<sub>1</sub> à Mot<sub>2</sub> (et réciproquement) au sein de leur famille morphologique. Cette information est renseignée dans le champ **Complexite** dont les valeurs possibles sont : **simple**, **complexe** et **lexical**.

— La relation est **simple** dans les trois cas suivants :

1. Mot<sub>1</sub> a comme base Mot<sub>2</sub> ou Mot<sub>2</sub> a pour base Mot<sub>1</sub>. Dans ce cas, Mot<sub>1</sub> et Mot<sub>2</sub> sont reliés par une opération dérivationnelle élémentaire. L’orientation de la relation peut être **descendant** ou **ascendant**.

Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
chanteur	Ncms	@AGF	chanter	Vmn----	@	simple
chanter	Vmn----	@	chanteur	Ncms	@ACT	simple

2. Mot<sub>1</sub> et Mot<sub>2</sub> ont la même base. Dans ce cas, Mot<sub>1</sub> et Mot<sub>2</sub> sont reliés par une opération dérivationnelle élémentaire à un même lexème. L’orientation de la relation est dans ce cas **indirect**. Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
chanteur	Ncms	@AGM	chanteuse	Ncfs	@AGF	simple

3. Mot<sub>1</sub> et Mot<sub>2</sub> appartiennent à des séries dont les mots sont connectés par des relations indirectes simples, comme la série des noms d’action en *-ion* et des adjectifs en *-if* ou des noms d’action en *-ion* et des noms d’agent en *-eur*. La différence avec le cas précédent réside en l’absence en synchronie d’une base commune à Mot<sub>1</sub> et Mot<sub>2</sub> :

Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
relation	Ncmf	@ACT	relatif	Afpms	@PRP	simple
prédateur	Ncmf	@AGM	prédation	Ncfs	@ACT	simple

3. Notation condensée : Exp est une abréviation de Exosant et TypCst de Type\_Constr.

— Les autres relations régulières qui s'établissent au sein des familles dérivationnelles sont **complexe**, notamment dans les trois cas suivants :

1. Mot<sub>1</sub> a comme ascendant non immédiat Mot<sub>2</sub> ou Mot<sub>2</sub> a comme ascendant non immédiat Mot<sub>1</sub>. L'orientation de la relation (Mot<sub>1</sub>, Mot<sub>2</sub>) peut donc être **descendant** ou **ascendant**. Dans ce cas, Mot<sub>1</sub> et Mot<sub>2</sub> sont reliés par une séquence d'opérations dérivationnelles élémentaires qui sont toutes descendantes ou toutes ascendantes.

*Exemple :*

Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
progressivité	Ncfs	@PRP	progresser	Vmn----	@	complexe
progresser	Vmn----	@	progressivité	Ncfs	@PRP	complexe

2. Les bases respectives de Mot<sub>1</sub> et Mot<sub>2</sub> sont deux mots qui entretiennent entre eux une relation **simple**, dont l'orientation a une valeur **indirect**, **ascendant** ou **descendant** :
  - c'est le cas de l'exemple (1), où **progressif**, base de **progressivité**, est le **descendant simple** de **progresser**, base de **progression**.
  - c'est le cas naturellement de l'exemple symétrique (2), où **progresser**, base de **progression**, est l'**ascendant simple** de **progressif**, base de **progressivité**.
  - c'est le cas de l'exemple (3), où **progressif**, base de **progressivité**, est en relation **indirect simple** avec **progression**, base de **non-progression**.

*Exemple :*

	Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
(1)	progressivité	Ncfs	@PRP	progression	Ncfs	@ACT	complexe
(2)	progression	Ncfs	@ACT	progressivité	Ncfs	@PRP	complexe
(3)	progressivité	Ncfs	@PRP	non-progression	Ncfs	@ACT	complexe

3. Les ascendants respectifs de Mot<sub>1</sub> et Mot<sub>2</sub>, entretiennent l'un avec l'autre une relation **complexe**

— La relation est **lexical** lorsque Mot<sub>1</sub> et Mot<sub>2</sub> sont morphologiquement apparentés mais qu'ils ne sont pas reliés par une relation régulière. Une relation est analysée comme **lexical** si elle est exceptionnelle. Dans les exemples (3) et (4) ci-dessous, les Mot<sub>1</sub> sont analysés comme étant suffixés en *-oire* et *-ande* respectivement.

*Exemple :*

	Graphie <sub>1</sub>	Cat <sub>1</sub>	Type <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Type <sub>2</sub>	Complexite
(1)	mensonge	Ncms	@ACT	mentir	Vmn----	@	lexical
(2)	parole	Ncfs	@ACT	parler	Vmn----	@	lexical
(3)	interrogatoire	Ncms	@ACT	interroger	Vmn----	@	lexical
(4)	offrande	Ncfs	@ACT	offrir	Vmn----	@	lexical

### 2.2.3 Synthèse

L'identification de la relation entre Mot<sub>1</sub> et Mot<sub>2</sub> résulte du croisement de la valeur des deux champs : Orientation (**descendant**, **ascendant**, **indirect**) et Complexite (**simple**, **complexe**, **lexical**). Le tableau ci-dessous illustre l'ensemble des combinaisons que l'on obtient en croisant ces deux propriétés.

*Exemple :*

Complexite	Orientation			
	descendant	ascendant	indirect	–
simple	( <i>danseur,</i> <i>danser</i> )	( <i>danser,</i> <i>danseur</i> )	( <i>perfectif,</i> <i>perfection</i> )	( <i>abandon,</i> <i>abandonner</i> )
complexe	( <i>indiscutable,</i> <i>discuter</i> )	( <i>discuter,</i> <i>indiscutable</i> )	( <i>scolarité,</i> <i>scolarisable</i> )	
lexical				( <i>exercer,</i> <i>exercice</i> )

*Remarques :*

- L'origine des valeurs des champs Orientation et Complexite est représentée par les champs Ori\_Orientation et Ori\_Complexite, respectivement. Actuellement, les valeurs possibles de ces champs sont : **deriv**, **morphonette**, **verbaction**.

- Les données intégrées dans la version 1.2 de Démonette n’instancient que la valeur `simple` du champ `Complexite`.

## 2.3 Relation formelle (Mot<sub>1</sub>, Mot<sub>2</sub>)

Dans chaque entrée de Démonette, Mot<sub>1</sub> est morphologiquement défini relativement à Mot<sub>2</sub>. D’un point de vue formel, les propriétés morphologiques de la relation entre Mot<sub>1</sub> et Mot<sub>2</sub> sont décrites dans six champs :

(Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub>, Type\_Constr<sub>2</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub>)

- Type\_Constr<sub>*i*</sub> décrit le mode de formation de Mot<sub>*i*</sub>. Les valeurs sont `suf` pour suffixation, `pref` pour préfixation et `conv` pour conversion.
- Exosant<sub>*i*</sub> fournit la valeur de l’exosant (l’affixe) utilisé. Ce champ est non instancié pour les conversions.
- Theme\_Graph<sub>*i*</sub> fournit le thème dérivationnel (i.e. le radical) utilisé pour construire la forme de Mot<sub>1</sub> et / ou de Mot<sub>2</sub>.

L’instanciation de ces six champs dépend du type de la relation (Mot<sub>1</sub>, Mot<sub>2</sub>)

**suffixation canonique** Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub> sont instanciés ;  
 Type\_Constr<sub>2</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont non instanciés ;  
 Orientation vaut `descendant`.

**suffixation canonique** Type\_Constr<sub>2</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont instanciés ;  
 Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub> sont non instanciés ;  
 Orientation vaut `ascendant`.

**suffixation canonique** Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Type\_Constr<sub>2</sub>, Exosant<sub>2</sub> sont instanciés ;  
 Theme\_Graph<sub>1</sub> et Theme\_Graph<sub>2</sub> sont non instanciés ;  
 Orientation vaut `indirect`.

**rétroformation** Type\_Constr<sub>1</sub>, Exosant<sub>1</sub> sont instanciés ;  
 Theme\_Graph<sub>1</sub>, Type\_Constr<sub>2</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont non instanciés ;  
 Orientation vaut `ascendant`.

**rétroformation** Type\_Constr<sub>2</sub>, Exosant<sub>2</sub> sont instanciés ;  
 Theme\_Graph<sub>2</sub>, Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub> sont non instanciés ;  
 Orientation vaut `descendant`.

**conversion** Type\_Constr<sub>1</sub> et Type\_Constr<sub>2</sub> valent `conv` ;  
 Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont non instanciés ;  
 Orientation n’est pas instancié.

**conversion** Type\_Constr<sub>1</sub> vaut `conv` et Type\_Constr<sub>2</sub>, Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont instanciés ;  
 Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub> sont non instanciés ;  
 Orientation vaut `indirect`.

**conversion** Type\_Constr<sub>2</sub> vaut `conv` et Type\_Constr<sub>1</sub>, Exosant<sub>1</sub>, Theme\_Graph<sub>1</sub> sont instanciés ;  
 Exosant<sub>2</sub>, Theme\_Graph<sub>2</sub> sont non instanciés ;  
 Orientation vaut `indirect`.

### 2.3.1 Cas canonique

Le tableau ci-dessous donne 4 exemples de la manière dont est représentée l’analyse morphologique formelle de Mot<sub>1</sub>, relativement à Mot<sub>2</sub> dans le cas canonique.

**Colonne 1 :** *conceptrice* est analysé relativement à sa base *concevoir*, verbe avec lequel ce nom entretient une relation descendante (cf. dernière ligne : Orientation = `descendant`). Le nom est construit au moyen du `suf(fixe)` *rice*, à partir du radical (dit « savant ») de *concevoir* (valeur du champ Theme\_Graph<sub>1</sub>).

**Colonne 2 :** La même relation ci-dessus est vue cette fois-ci en inversant les rôles de *conceptrice* et *concevoir* (Orientation = `ascendant`). En terme d’analyse formelle, cette inversion revient à échanger les valeurs des champs d’indices 1 et 2 de la colonne précédente.

**Colonnes 3 et 4 :** Dans la relation caractérisée par **Orientation = indirect**,  $Mot_1$  et  $Mot_2$  sont des descendants d'un même prédicat qui est soit attesté (colonne 3) soit non accessible en français contemporain (colonne 4) : en effet, alors que *créatrice* et *créateur* ont pour base le verbe *créer*, les noms *prédateur* et *prédation* ont perdu, en synchronie, la base verbale à laquelle ils ont été reliés dans un état antérieur de la langue. Néanmoins, les seconds, comme les premiers, conservent une marque suffixale qui les intègre dans une série morphologique identifiée : celle des noms masculins d'agent, pour *prédateur*, celle des noms d'action, pour *prédation*.

Exemple :

	1	2	3	4
Graphie <sub>1</sub>	conceptrice	concevoir	créatrice	prédateur
Cat <sub>1</sub>	Ncfs	Vmn----	Ncfs	Ncms
Type <sub>1</sub>	@AGF	@	@AGF	@AGM
Graphie <sub>2</sub>	concevoir	conceptrice	créateur	prédation
Cat <sub>2</sub>	Vmn----	Ncfs	Ncms	Ncfs
Type <sub>2</sub>	@	@AGF	@AGM	@ACT
Type_Constr <sub>1</sub>	suf	–	suf	suf
Exposant <sub>1</sub>	rice	–	rice	eur
Theme_Graph <sub>1</sub>	concept	–	–	prédat
Type_Constr <sub>2</sub>	–	suf	suf	suf
Exposant <sub>2</sub>	–	rice	eur	ion
Theme_Graph <sub>2</sub>	–	concept	–	prédat
Orientation	descendant	ascendant	indirect	indirect

Remarques :

- Les sources d'origine des informations contenues dans les champs `Type_Constr1`, `Type_Constr2`, `Exposant1`, `Exposant2`, `Theme_Graph1` et `Theme_Graph2` sont données, respectivement, par les champs `Ori_Type_Constr1`, `Ori_Type_Constr2`, `Ori_Constr1`, `Ori_Constr2`, `Ori_Theme_Graph1` et `Ori_Theme_Graph2`.
- Actuellement, les valeurs possibles pour ces champs sont : **dérif**, **morphonette** et **verbaction**.
- Dans la colonne 4, `Theme_Graph1` est renseigné (**prédat**) mais le champ est vide dans la colonne 3. Cela est dû à la différence de traitement entre DériF (origine du couple (*prédateur*, *prédation*)) et Morphonette (origine du couple (*créatrice*, *créateur*)).

### 2.3.2 Rétroformation

Le tableau ci-dessous donne 2 exemples de la manière dont est représentée l'analyse morphologique formelle de  $Mot_1$ , relativement à  $Mot_2$  dans un cas de rétroformation.

**Colonne 1 :** *hydroplaner* est analysé relativement à sa base *hydroplanage*, nom avec lequel ce verbe entretient une relation descendante (cf. dernière ligne : **Orientation = descendant**). Le verbe est construit en supprimant le **suf**(fixe) **age**.

**Colonne 2 :** La même relation ci-dessus est vue en inversant les rôles de *hydroplaner* et *hydroplanage* (**Orientation = ascendant**). En terme d'analyse formelle, cette inversion revient à échanger les valeurs des champs d'indices 1 et 2 de la colonne précédente.

Exemple :

	1	2
Graphie <sub>1</sub>	hydroplaner	hydroplanage
Cat <sub>1</sub>	Vmn----	Ncfs
Type <sub>1</sub>	@	@ACT
Graphie <sub>2</sub>	hydroplanage	hydroplaner
Cat <sub>2</sub>	Ncfs	Vmn----
Type <sub>2</sub>	@ACT	@
Type_Constr <sub>1</sub>	–	suf
Exposant <sub>1</sub>	–	age
Theme_Graph <sub>1</sub>	–	–
Type_Constr <sub>2</sub>	suf	–
Exposant <sub>2</sub>	age	–
Theme_Graph <sub>2</sub>	–	–
Orientation	descendant	ascendant

### 2.3.3 Conversion

Le tableau ci-dessous donne 5 exemples de la manière dont est représentée l'analyse morphologique formelle de  $Mot_1$ , relativement à  $Mot_2$  dans un cas de rétroformation.

**Colonne 1 :** *additionner* est analysé relativement à sa base *addition*, nom avec lequel ce verbe entretient une relation descendante (cf. dernière ligne : **Orientation = descendant**). Le verbe est construit par conv(ersion).

**Colonne 2 :** La même relation ci-dessus est vue en inversant les rôles de *additionner* et *addition* (**Orientation = ascendant**). En terme d'analyse formelle, cette inversion revient à échanger les valeurs des champs d'indices 1 et 2 de la colonne précédente.

**Colonnes 3 et 4 :** *analyser* et *analyse* sont dans une relation de conversion, mais le programme utilisé pour analyser Verbaction n'est pas en mesure de déterminer son orientation.

**Colonne 5 :** *chromer* et *chromisation* descendent du nom *chrome*, l'un par conversion et l'autre par suffixation en *-ion* via *chromiser*. **Orientation = indirect**.

Exemple :

	1	2	3	4	5
Graphie <sub>1</sub>	additionner	addition	analyser	analyse	chromer
Cat <sub>1</sub>	Vmn----	Ncfs	Vmn----	Ncfs	Vmn----
Type <sub>1</sub>	@	@ACT	@	@ACT	@
Graphie <sub>2</sub>	addition	additionner	analyse	analyser	chromisation
Cat <sub>2</sub>	Ncfs	Vmn----	Ncfs	Vmn----	Ncfs
Type <sub>2</sub>	@ACT	@	@ACT	@	@ACT
Type_Constr <sub>1</sub>	conv	–	conv	conv	conv
Exposant <sub>1</sub>	–	–	–	–	–
Theme_Graph <sub>1</sub>	–	–	–	–	–
Type_Constr <sub>2</sub>	–	conv	conv	conv	suf
Exposant <sub>2</sub>	–	–	–	–	ion
Theme_Graph <sub>2</sub>	–	–	–	–	–
Orientation	descendant	ascendant	–	–	indirect

## 2.4 Propriétés sémantiques de ( $Mot_1$ , $Mot_2$ )

Trois types d'informations sémantiques sont codées dans Démonette pour la relation de  $Mot_1$  avec  $Mot_2$  :

- la définition de  $Mot_1$  (appelée **définition concrète**), relativement à sa relation avec  $Mot_2$  ;
- son **type sémantique** ;
- le type sémantique de cette relation (appelé **définition abstraite**).

Les champs décrits ci-dessous sont les suivants :

(Def_Conc, Type_1, Type_2, Def_Abs)
-------------------------------------

### 2.4.1 Définition concrète de $Mot_1$

Chaque entrée de Démonette comporte une définition, décrivant le sens de  $Mot_1$  par rapport à celui de  $Mot_2$ . Cette définition est contenue dans le champ Def\_Conc. Le champ Ori\_Def\_Conc indique, quant à lui, l'origine de cette information.

Exemple :

Graphie <sub>1</sub>	Cat <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>	Def_Conc
danseuse	Ncfs	danser	Vmn----	(agent féminin OR instrument) de danser
danseuse	Ncfs	danseur	Ncms	celle qui a pour correspondant masculin danseur

Remarque :

- Dans la distribution courante, la définition concrète de  $Mot_1$ , qui instancie le champ Def\_Conc est définie soit dans DériF, soit dans Démonette (pour les entrées issue de Morphonette ou de Verbaaction), comme le précise la valeur du champ Ori\_Def\_Conc.

- Le champ `Def_Conc` des relations où `Complexite = lexical` et des conversions issues de `Morphonette` n'est pas renseigné : les propriétés formelles ne permettent de calculer cette information.

## 2.4.2 Type sémantique de `Mot1`

Le type sémantique complète la graphie et la catégorie pour garantir une identité univoque de `Moti`. Ce type sémantique est calculé à partir des propriétés morphologiques de `Moti` (cf. section 2.3). Dans la notation employée pour typer `Moti`, le symbole '@' représente un prédicat (le plus souvent) verbal qui est généralement la base morphologique de `Moti`, à une exception notable près : les conversions où le prédicat peut être le convers (i.e. le descendant). Les autres types indiquent le rôle sémantique joué par `Moti` dans sa relation avec `Motj`. Les types actuellement instanciés dans `Démonette` sont les suivants :

**@ :** désigne un prédicat (verbal).

**@ACT :** désigne une action. C'est l'un des types par défaut des noms dérivés en *-age*, *-ment* et *-ion*.

**@RES :** désigne le résultat d'une action. C'est l'un des types par défaut des noms dérivés en *-age*, *-ment* et *-ion*.

**@AGM :** désigne un agent masculin (ou un instrument). C'est l'un des types par défaut des noms dérivés en *-eur*.

**@AGF :** désigne un agent féminin (ou un instrument). C'est l'un des types par défaut des noms dérivés en *-euse* et *-rice*.

**@PRP :** désigne une propriété. C'est l'un des types par défaut des adjectifs dérivés en *-if*.

L'exemple ci-dessous résume et illustre les combinaisons de types que l'on trouve dans `Démonette` (version 1.2). La valeur du suffixe ayant servi à la formation de `Mot1` et/ou `Mot2` est indiquée en gras dans les colonnes `Graphiei`. Les combinaisons sont symétriques : à chaque entrée (`Mot1`, `Mot2`) qui met en jeu l'un des couples (`Type1`, `Type2`) ci-dessous, correspond une entrée inverse (`Mot2`, `Mot1`), caractérisée par le couple (`Type2`, `Type1`) (cf. section 1).

Exemple :

Type <sub>1</sub>	Type <sub>2</sub>	Exemple :			
		Graphie <sub>1</sub>	Cat <sub>1</sub>	Graphie <sub>2</sub>	Cat <sub>2</sub>
@ACT	@	déambul <b>age</b>	Ncms	déambuler	Vmn----
@ACT	@AGF	aboie <b>ment</b>	Ncms	aboye <b>use</b>	Ncfs
@ACT	@AGM	débouch <b>age</b>	Ncms	débouch <b>eur</b>	Ncms
@ACT	@PRP	déduct <b>ion</b>	Ncfs	déduct <b>if</b>	Afpms
@RES	@	déduct <b>ion</b>	Ncfs	déduire	Vmn----
@AGF	@	destructr <b>ice</b>	Ncfs	détruire	Vmn----
@AGF	@RES	délatr <b>ice</b>	Ncfs	délat <b>ion</b>	Ncfs
@AGF	@AGM	danse <b>use</b>	Ncfs	danse <b>ur</b>	Ncms
@AGF	@PRP	auditr <b>ice</b>	Ncfs	audit <b>if</b>	Afpms
@AGM	@	chante <b>ur</b>	Ncms	chanter	Vmn----
@AGM	@PRP	dépresse <b>ur</b>	Ncms	dépress <b>if</b>	Afpms
@AGM	@RES	déserte <b>ur</b>	Ncms	désert <b>ion</b>	Ncfs
@PRP	@RES	correct <b>if</b>	Afpms	correct <b>ion</b>	Ncfs
@AGM	@AGM	sauve <b>ur</b>	Ncms	sauvete <b>ur</b>	Ncms
@AGF	@AGF	donne <b>use</b>	Ncfs	donatr <b>ice</b>	Ncfs
@ACT	@ACT	lav <b>age</b>	Ncms	lavem <b>ent</b>	Ncms

Remarque :

- Dans la distribution courante, les types sémantiques décrits dans les champs `Type1` et `Type2`, ont tous été calculés lors de la création de `Démonette` comme indiqué dans les champs `Ori_Type1` et `Ori_Type2`.

## 2.4.3 Définition abstraite de `Mot1`

Dans une entrée (`Mot1`, `Mot2`), où `Mot1` a pour type sémantique `Type1`, et `Mot2`, `Type2`, la définition abstraite de `Mot1` s'obtient en substituant `Type2` à `Graphie2` dans la définition concrète qui se trouve dans le champ `Def_Conc`. De cette manière, on obtient une glose (contenue dans le champ `Def_Abs`), qui sert, non pas à définir une unité du lexique, relativement eu sens d'un membre de sa famille morphologique (ce que fait le contenu du champ `Def_Conc`), mais à caractériser l'instruction sémantique associée à la règle

morphologique qui relie  $Mot_1$  à  $Mot_2$ , et à identifier l'ensemble des couples qui instancient cette règle. En d'autres termes, cette glose abstraite permet de retrouver l'ensemble des  $(Mot_i, Mot_j)$  qui se comportent morphologiquement comme une paire  $(Mot_1, Mot_2)$  donnée, c'est-à-dire où  $Mot_i$  possède la même catégorie  $Cat_1$  et les mêmes caractéristiques morphosémantiques que  $Mot_1$ , relativement à  $Mot_j$  qui, lui, possède la même catégorie  $Cat_2$  et les mêmes caractéristiques morphosémantiques que  $Mot_2$ .

Nous reprenons dans l'exemple ci-dessous les définitions concrètes qui ont servi à illustrer la section 2.4.1, pour montrer comment on passe de ces définitions aux définitions abstraites contenues dans les champs `Def_Abs`. Le détail des gloses ou définitions abstraites présentes dans *Démonette* (version 1.2) est donné dans la section 3.

*Exemple :*

Graphie <sub>1</sub> , Cat <sub>1</sub> , Type <sub>1</sub>	danseuse, Ncfs, @AGF
Graphie <sub>2</sub> , Cat <sub>2</sub> , Type <sub>2</sub>	danser, Vmn----, @
Def_Conc	(agent féminin OR instrument) de danser
Def_Abs	(agent féminin OR instrument) de @
Graphie <sub>1</sub> , Cat <sub>1</sub> , Type <sub>1</sub>	danseuse, Ncfs, @AGF
Graphie <sub>2</sub> , Cat <sub>2</sub> , Type <sub>2</sub>	danseur, Ncms, @AGM
Def_Conc	celle qui a pour correspondant masculin danseur
Def_Abs	celle qui a pour correspondant masculin @AGM

*Remarque :*

- Dans la distribution courante, les définitions abstraites, qui instancient les champs `Def_Abs`, ont toutes été calculées lors de la création de *Démonette*, comme le précise le champ `Ori_Def_Abs`.
- Les entrées dont la complexité est `lexical` issues de *Verbaction* et les conversions issues de *Morphonette* qui n'ont pas de définition concrète n'ont pas non plus de définition abstraite.

### 3 Les définitions abstraites

Les définitions abstraites sont l'une des caractéristiques originales de *Démonette*. Ces définitions sont obtenues par abstraction de `Graphie2` dans les définitions concrètes. Par défaut, la définition abstraite de  $Mot_1$  est directement déterminée par les `Type1` et `Type2`. Dans certains cas, cette définition dépend également de la `Cati`. Le tableau suivant donne l'ensemble de ces définitions pour chaque couple dont les membres sont formés d'un type et de la valeur de la catégorie. Par souci de lisibilité, et quand le genre est une information non pertinente, le symbole `Nc.s` remplace `Ncms` ou `Ncfs` dans l'indication de la catégorie des noms communs féminins ou masculins. Les couples absents de ce tableau n'apparaissent pas dans *Démonette*.

Type <sub>1</sub> /Cat <sub>1</sub>	Type <sub>2</sub> /Cat <sub>2</sub>	Def_Abs	(Mot <sub>1</sub> , Mot <sub>2</sub> )
@ACT/Nc.s	@/Vmn----	action de @	( <i>abordage / aborder</i> )
@ACT/Nc.s	@/Nc.s	action en lien avec le @	( <i>gastrulation / gastrula</i> )
@AGF/Ncfs	@/Vmn----	(agent féminin OR instrument) de @	( <i>danseuse / danser</i> )
@AGM/Ncms	@/Vmn----	(agent masculin OR instrument) de @	( <i>accordeur / accorder</i> )
@PRP/Afpms	@/Vmn----	en rapport avec l'acte de @	( <i>admiratif / admirer</i> )
@RES/Nc.s	@/Vmn----	résultat de @	( <i>abstention / abstenir</i> )
@RES/Nc.s	@/Nc.s	résultat en lien avec le @	( <i>ioduration / iodure</i> )
@/Nc.s	@ACT/Nc.s	entité servant au @ACT	( <i>azoture / azoturation</i> )
@/Vmn----	@ACT/Nc.s	réaliser le @ACT	( <i>laver / lavage</i> )
@ACT/Nc.s	@ACT/Nc.s	@ACT	( <i>abattage / abattement</i> )
@AGF/Ncfs	@ACT/Nc.s	(agent féminin OR instrument) du @ACT	( <i>agricultrice / agriculture</i> )
@AGM/Ncms	@ACT/Nc.s	(agent masculin OR instrument) du @ACT	( <i>footballeur / football</i> )
@PRP/Afpms	@ACT/Nc.s	qui résulte de @ACT	( <i>adhésif / adhésion</i> )
@/Vmn----	@AGF/Ncfs	réaliser l'action dont (l'agent féminin OR l'instrument) est @AGF	( <i>adoucir / adoucisseuse</i> )
@ACT/Nc.s	@AGF/Ncfs	action dont (l'agent féminin OR l'instrument) est @AGF	( <i>imprécation / imprécatrice</i> )

@AGF/Ncfs	@AGF/Ncfs	@AGF	( <i>calculouse/ calculatrice</i> )
@AGM/Ncms	@AGF/Ncfs	celui qui a pour correspondant féminin @AGF	( <i>agriculteur/ agricultrice</i> )
@PRP/Afpms	@AGF/Ncfs	qui caractérise l'action pratiquée par @AGF	( <i>décoratif/ décoratrice</i> )
@RES/Nc.s	@AGF/Ncfs	résultat de l'action pratiquée par @AGF	( <i>compétition/ compétitrice</i> )
@/Vmn----	@AGM/Ncms	réaliser l'action dont (l'agent masculin OR l'instrument) est @AGM	( <i>administrer/ administrateur</i> )
@ACT/Nc.s	@AGM/Ncms	action dont (l'agent masculin OR l'instrument) est @AGM	( <i>agriculture/ agriculteur</i> )
@AGF/Ncfs	@AGM/Ncms	celle qui a pour correspondant masculin @AGM	( <i>aguicheuse/ aguicheur</i> )
@AGM/Ncms	@AGM/Ncms	@AGM	( <i>acteur/activateur</i> )
@PRP/Afpms	@AGM/Ncms	qui caractérise l'action pratiquée par @AGM	( <i>compétitif/ compétiteur</i> )
@RES/Nc.s	@AGM/Ncms	résultat de l'action pratiquée par @AGM	( <i>croquis/ croquiseur</i> )
@/Vmn----	@PRP/Afpms	manifeste le fait d'être @PRP	( <i>agresser/agressif</i> )
@ACT/Nc.s	@PRP/Afpms	action dont le résultat a la propriété @PRP	( <i>alternation/ alternatif</i> )
@AGF/Ncfs	@PRP/Afpms	celle dont l'action a un résultat qui a la propriété @PRP	( <i>coopératrice/ coopératif</i> )
@AGM/Ncms	@PRP/Afpms	celui dont l'action a un résultat qui a la propriété @PRP	( <i>comparateur/ comparatif</i> )
@RES/Nc.s	@PRP/Afpms	résultat ayant la propriété @PRP	( <i>description/ descriptif</i> )
@/Vmn----	@RES/Nc.s	réaliser l'action dont le résultat est un @RES	( <i>abrutir/ abrutissement</i> )
@/Nc.s	@RES/Nc.s	entité servant à l'action dont le résultat est le @RES	( <i>iodure/ ioduration</i> )
@AGF/Ncfs	@RES/Nc.s	(agent féminin OR instrument) de l'action dont le résultat est un @RES	( <i>médiatrice/ médiation</i> )
@AGM/Ncms	@RES/Nc.s	(agent masculin OR instrument) de l'action dont le résultat est un @RES	( <i>inquisiteur/ inquisition</i> )
@PRP/Afpms	@RES/Nc.s	qui caractérise l'action dont le résultat est @RES	( <i>amplificatif/ amplification</i> )
@RES/Nc.s	@RES/Nc.s	@RES	( <i>agglutination/ agglutinement</i> )

## 4 Liste complète des champs pour le format csv

Démonette-1.2 est distribuée dans un format texte tabulé (csv) et au format XML. On trouvera ci-dessous les champs décrits ci-dessus, dans l'ordre dans lequel ils apparaissent dans Démonette-1.2 au format **tabulé**. Pour chaque champ, on trouvera, entre parenthèses, le nom de la balise et/ou de l'attribut qui lui correspond dans le format **XML** de Démonette-1.2:

1. Graphie<sub>1</sub> : graphie du mot cible  
XML: balise <targetWord><writtenForm>
2. Ori\_Graphie<sub>1</sub> : origine de la graphie du mot cible  
XML: attribut **origin** de la balise <targetWord><writtenForm>
3. Graphie<sub>2</sub> : graphie du mot source  
XML: balise <sourceWord><writtenForm>
4. Ori\_Graphie<sub>2</sub> : origine de la graphie du mot source  
XML: attribut **origin** de la balise <sourceWord><writtenForm>
5. Cat<sub>1</sub> : étiquette morphosyntaxique du mot cible  
XML: balise <targetWord><morphoSyntacticTag>
6. Ori\_Cat<sub>1</sub> : origine de l'étiquette morphosyntaxique du mot cible  
XML: attribut **origin** de la balise <targetWord><morphoSyntacticTag>

7. *Cat<sub>2</sub>* : étiquette morphosyntaxique du mot source  
XML: balise <sourceWord> <morphoSyntacticTag>
8. *Ori\_Cat<sub>2</sub>* : origine de l'étiquette morphosyntaxique du mot source  
XML: attribut *origin* de la balise <sourceWord><morphoSyntacticTag>
9. *Complexite* : complexité de la relation (*simple* / *complexe* / *lexical*)  
XML: balise <relationType><complexity>
10. *Ori\_Complexite* : origine de l'information de complexité de la relation  
XML: attribut *origin* de la balise <relationType>
11. *Procede<sub>1</sub>* : procédé dérivationnel utilisé pour construire le mot cible (*pre* / *suf* / *conv*)  
XML: balise <targetFormConstruction><constructionalProcess>
12. *Exposant<sub>1</sub>* : exposant dérivationnel du mot cible  
XML: balise <targetFormConstruction><constructionalExponent>
13. *Ori\_Constr<sub>1</sub>* : origine des informations concernant le procédé et l'exposant dérivationnel du mot cible  
XML: attribut *origin* de la balise <targetFormConstruction><constructionalProcess> et <targetFormConstruction><constructionalExponent>
14. *Procede<sub>2</sub>* : procédé dérivationnel utilisé pour construire le mot source (*pre* / *suf* / *conv*)  
XML: balise <sourceFormConstruction><constructionalProcess>
15. *Exposant<sub>2</sub>* : exposant dérivationnel du mot source  
XML: balise <sourceFormConstruction><constructionalExponent>
16. *Ori\_Constr<sub>2</sub>* : origine des informations concernant le procédé et l'exposant dérivationnel du mot source  
XML: attribut *origin* des balises <sourceFormConstruction><constructionalProcess> et <sourceFormConstruction><constructionalExponent>
17. *Type<sub>1</sub>* : type morphosémantique du mot cible  
XML: balise <targetWord><morphoSemanticType>
18. *Ori\_Type<sub>1</sub>* : origine du type morphosémantique du mot cible  
XML: attribut *origin* de la balise <targetWord><morphoSemanticType>
19. *Type<sub>2</sub>* : type morphosémantique du mot source  
XML: balise <sourceWord><morphoSemanticType>
20. *Ori\_Type<sub>2</sub>* : origine du type morphosémantique du mot source  
XML: attribut *origin* de la balise <sourceWord><morphoSemanticType>
21. *Def\_concrete* : définition concrète du sens construit du mot cible  
XML: balise <targetMeaningConstruction><concreteDefinition>
22. *Ori\_Def\_concrete* : origine de la définition concrète du sens construit du mot cible  
XML: attribut *origin* de la balise <targetMeaningConstruction><concreteDefinition>
23. *Def\_abstraite* : définition abstraite du sens construit du mot cible  
XML: balise <targetMeaningConstruction><abstractDefinition>
24. *Ori\_Def\_abstraite* : origine de la définition abstraite du sens construit du mot cible  
XML: attribut *origin* de la balise <targetMeaningConstruction><abstractDefinition>
25. *Direction* : orientation de la relation dérivationnelle (*descendant* / *ascendant* / *indirect*)  
XML: balise <relationType><direction>
26. *Ori\_Direction* : origine de l'orientation de la relation dérivationnelle  
XML: attribut *origin* de la balise <relationType>
27. *Theme<sub>1</sub>* : thème dérivationnel du mot cible  
XML: balise <targetFormConstruction><constructionalTheme>
28. *Ori\_Theme<sub>1</sub>* : origine du thème dérivationnel du mot cible  
XML: attribut *origin* de la balise <targetFormConstruction><constructionalTheme>
29. *Theme<sub>2</sub>* : thème dérivationnel du mot source  
XML: balise <sourceFormConstruction><constructionalTheme>

- 30. Ori\_Theme<sub>2</sub> : origine du thème dérivationnel du mot source  
XML: attribut `origin` de la balise  
`<targetFormConstruction><constructionalTheme>`
- 31. Ori\_Couple: origine du couple  
XML: attribut `origin` de la balise `<morphologicalRelation>`

## 5 Exemple XML

Extrait de la version XML de Démonette qui correspond à la relation *enregistreur:enregistrer* issue de DériF.

```
<morphologicalRelation origin="derif">
  <targetWord>
    <writtenForm origin="tlfnome">enregistreur</writtenForm>
    <morphoSyntacticTag origin="tlfnome">Ncms</morphoSyntacticTag>
    <morphoSemanticType origin="demonette">@AGM</morphoSemanticType>
  </targetWord>
  <sourceWord>
    <writtenForm origin="tlfnome">enregistrer</writtenForm>
    <morphoSyntacticTag origin="tlfnome">Vmn----</morphoSyntacticTag>
    <morphoSemanticType origin="demonette">@</morphoSemanticType>
  </sourceWord>
  <relationType origin="derif">
    <direction>descendant</direction>
    <complexity>simple</complexity>
  </relationType>
  <targetFormConstruction>
    <constructionalProcess origin="derif">suf</constructionalProcess>
    <constructionalExponent origin="derif">eur</constructionalExponent>
    <constructionalTheme origin="derif">enregistr</constructionalTheme>
  </targetFormConstruction>
  <sourceFormConstruction></sourceFormConstruction>
  <targetMeaningConstruction>
    <concreteDefinition origin="derif">(agent masculin OR instrument) de enregistrer</concreteDefinition>
    <abstractDefinition origin="demonette">(agent masculin OR instrument) de @</abstractDefinition>
  </targetMeaningConstruction>
</morphologicalRelation>
```

## Références

RAJMAN, M., LECOMTE, J. et PAROUBEK, P. (1997). Format de description lexicale pour le français. Partie 2: Description morpho-syntaxique. Rapport technique, EPFL & INaLF. GRACE GTR-3-2.1.