

## **Aide à la création lexicale en langue de minorité : problèmes et perspectives de l'application de la ZStation en langue yaqui**

Albert ALVAREZ GONZALEZ

*Université de Sonora, Mexique*

### **0. Introduction**

L'objectif de cet article consiste à présenter mais également à évaluer d'une certaine manière les possibilités d'aide à la création lexicale en langue de minorité que permet le programme ZStation, formalisme informatique développé au sein du Laboratoire d'Ingénierie Linguistique et de Linguistique Appliquée (LILLA) de l'Université de Nice-Sophia Antipolis par le professeur Henri Zinglé<sup>1</sup> (1994, 1996). Ce travail s'inscrit dans un projet de recherche en linguistique appliquée qui, en 2002, sous l'impulsion de feu M. Zinglé, a été mis en place conjointement entre l'Université de Sonora et l'Université de Nice-Sophia Antipolis avec pour but, celui d'associer la linguistique descriptive et indigéniste mexicaine avec l'ingénierie linguistique, en combinant une étude descriptive de morphologie et sémantique lexicale dans une langue indigène du Mexique (la langue yaqui) avec un système informatique de création et analyse lexicales (le programme ZStation). Ce projet qui a déjà été présenté dans sa phase initiale (Zinglé & Alvarez 2003), a

---

<sup>1</sup> Cet article veut être également un hommage à M. Zinglé, notre regretté professeur à l'origine de ce projet. Assurément un grand homme qui nous a quittés trop tôt.

permis de développer dans un premier temps une analyse linguistique pour passer dans une seconde étape à son traitement automatique. L'analyse formelle et sémantique des mécanismes combinatoires de création de mots en langue yaqui<sup>2</sup> a ainsi pu servir de base à l'analyse et à la génération automatiques de mots et de néologismes dans cette langue.

En premier lieu, il s'agira donc de présenter les applications de la ZStation qui ont été développées en langue yaqui. Nous montrerons alors comment, à partir d'une analyse morphosémantique des suffixes et des mots composés du yaqui et de la modélisation informatique de la description linguistique permise par la ZStation, on peut arriver à un système de génération et d'analyse automatiques de néologismes combinés yaquis. Pour ce faire, nous présenterons brièvement dans un premier chapitre une description morphologique et sémantique des mécanismes lexicogénétiques de combinaison que possède la langue yaqui (suffixation et composition), analyse qui servira dans un second chapitre pour la modélisation et la représentation des connaissances linguistiques à travers le programme ZStation, formalisme informatique qui fonctionne en utilisant des bases de connaissances orientées à objets. Après avoir fait état des présupposés linguistiques et des principes de formalisation de la ZStation, nous tenterons d'illustrer et d'expliquer, à partir d'un corpus de mots suffixés et composés de la langue yaqui, les différentes étapes de la réalisation informatique (élaboration du dictionnaire d'unités morphologiques avec la prise en compte des caractéristiques catégorielles, sémantiques et formatives des formants impliqués, élaboration des ontologies indispensables pour la prise en compte des restrictions sémantiques) ainsi que les deux applications permises par le programme, à savoir l'analyse et la génération automatiques de mots.

Dans un dernier chapitre, nous essaierons d'évaluer cet outil d'analyse automatique et d'aide à la création lexicale en langue de minorité, en considérant des aspects liés aussi bien à l'outil informatique en soi qu'à sa fonction au sein de la communauté. Le travail se penchera alors sur les problèmes et les perspectives entrouvertes par ce genre d'étude, tant aux niveaux linguistique et informatique (description, modélisation) qu'aux niveaux social et politique (accueil et usage de cet outil dans la communauté, rôle dans la politique et planification linguistique).

---

<sup>2</sup> La langue yaqui appartient à la famille uto-aztèque. Elle est parlée dans l'État de Sonora, dans le nord-ouest du Mexique par une communauté d'environ 20 000 locuteurs. Le livre *Parlons yaqui* (Estrada & Alvarez 2008) présente la langue ainsi que les aspects historiques et culturels les plus importants de cette communauté.

## 1. La lexicogénèse combinatoire de la langue yaqui

Comme nous l'avons déjà mentionné ailleurs (Alvarez 1998a, 1998b, 2002), les langues naturelles ont principalement à leur disposition deux grands types de mécanismes de formation de mots selon que la nouveauté lexicogénétique se situe dans la forme ou exclusivement dans le sens de l'unité lexicale formée. Ainsi, si nous considérons le signe linguistique dans sa définition saussurienne, c'est-à-dire comme la combinaison d'un signifiant ("une image acoustique" : la forme) et d'un signifié ("le concept" : le sens), on peut alors distinguer une lexicogénèse de forme représentée par tous les mécanismes qui permettent de former de nouveaux signifiants et une lexicogénèse de sens représentée par tous les mécanismes qui permettent d'associer de nouveaux signifiés à des mots déjà existants dans le lexicon de la langue. La lexicogénèse formelle, contrairement à la lexicogénèse sémantique, se caractérise par conséquent par la génération de nouveaux mots et elle apparaît à travers l'apparition d'un signifiant non enregistré encore dans le lexicon de la langue.

L'introduction de ces unités lexicales dans le système lexico-sémantique de la langue obéit à différentes stratégies et on peut ainsi distinguer différents types de lexicogénèse formelle selon que l'élément lexical formé est le résultat d'une combinaison, d'un abrègement ou d'une réduction d'éléments déjà existants dans la langue ou bien encore le résultat d'une onomatopée ou d'un emprunt (Alvarez 1998b: 15).

De ces divers processus lexicogénétiques, la lexicogénèse combinatoire représente, sans aucun doute, un des mécanismes morphologiques les plus communs et les plus productifs que possèdent les langues naturelles pour former des mots. Celle-ci inclut les phénomènes de dérivation et de composition, où l'on combine des morphèmes dépendants ou des mots autonomes avec des mots déjà existants.

### 1.1. La suffixation yaqui

Des différents mécanismes dérivatifs que possèdent les langues pour former des mots (préfixation, infixation, suffixation et les possibles combinaisons), la langue yaqui utilise uniquement la suffixation<sup>3</sup>. Ce procédé consiste à ajouter un élément final (le suffixe) à une base lexicale déjà existante dans la langue. Par cette construction, le concept primaire

---

<sup>3</sup> On ne considère évidemment ici que les phénomènes de morphologie lexicale et non ceux de morphologie flexive.

représenté par la base lexicale est orienté vers un nouveau sens et, selon le type de suffixe utilisé, la nouvelle formation lexicale peut aussi acquérir une catégorie grammaticale différente au lexème d'origine. Ainsi, on peut obtenir, à partir d'une même base lexicale, plusieurs mots dérivés qui auront des catégories grammaticales différentes selon le suffixe utilisé<sup>4</sup>.

La suffixation en langue yaqui peut apparaître à partir de 4 formes de base (substantif, verbe, adjectif et adverbe) qui vont créer, selon le suffixe employé, un substantif, un verbe, un adjectif ou un adverbe. Auteurs d'une description générale de la grammaire de la langue yaqui, Dedrick et Casad (1999: 119-155) indiquent à juste titre qu'il s'agit d'une langue qui fait un grand usage du mécanisme de suffixation dans la formation de son vocabulaire. Tant la morphologie nominale, comme adjectivale, verbale et adverbiale, utilisent, en effet, grandement cette stratégie lexicogénétique ce qui montre la grande productivité du phénomène dérivatif en yaqui.

La révision critique du travail de Dedrick et Casad (1999) ainsi que la consultation du *Dictionnaire Bilingue Yaqui-Espagnol* élaboré à l'Université de Sonora par une équipe dirigée par la professeure Zarina Estrada Fernández (Estrada *et al.* 2004), nous ont permis d'établir une liste des principaux suffixes du yaqui auxquels nous avons essayé d'attribuer une valeur sémantique selon le(s) changement(s) de sens provoqué(s) à la base lexicale. Les exemples de mots suffixés yaquis qui apparaissent plus bas ont été pris des deux ouvrages de référence précédemment cités et ils montrent également les différentes bases lexicales possibles selon les cas.

### 1.1.1. Suffixes nominaux

	<b>-wa</b> : "qualité abstraite", "semblable à"		
déadjectival :	<i>kia</i> (savoureux)	+	-wa = <i>kiawa</i> (saveur)
dénominal :	<i>ba'a</i> (eau)	+	-wa = <i>ba'awa</i> (bouillon)
	<b>-wame</b> : "résultat"		
déverbal :	<i>etbwa</i> (voler)	+	-wame = <i>etbwawame</i> (vol)
	<b>-me</b> : "agent"		
déverbal :	<i>nenka</i> (vendre)	+	-me = <i>nenkame</i> (vendeur)
	<b>-reo</b> : "agent"		
déverbal :	<i>jiyoka</i> (peindre)	+	-reo = <i>jiyokareo</i> (peintre)
	<b>-raa</b> : "qualité"		
déadjectival:	<i>awi</i> (gros)	+	-raa= <i>awiraa</i> (embonpoint, gras)

<sup>4</sup> C'est le cas par exemple du mot yaqui *ba'a* 'eau' qui figure dans notre corpus et donne comme dérivés, le substantif *ba'awa* 'bouillon', les verbes *ba'ate* 'rendre liquide' et *ba'au* 'extraire un certain objet d'un liquide' et l'adjectif *ba'ari* 'humide, mouillé'.

**-i** : "résultat", "responsable"

déverbal :	<i>etejo</i> (parler)	+	<i>-i</i>	=	<i>etejoi</i> (histoire)
dénominal :	<i>ko'oko</i> (douleur)	+	<i>-i</i>	=	<i>ko'okoi</i> (chili, piment)

**-ria** : "résultat"

déverbal :	<i>chuyu</i> (charger l'estomac)	+	<i>-ria</i>	=	<i>chuyuria</i> (indigestion)
------------	----------------------------------	---	-------------	---	-------------------------------

### 1.1.2. Suffixes verbaux

**-te** : "faire, convertir"

déadjectival :	<i>bwe'u</i> (grand)	+	<i>-te</i>	=	<i>bwe'ute</i> (agrandir)
dénominal :	<i>ba'a</i> (eau)	+	<i>-te</i>	=	<i>ba'ate</i> (rendre liquide)

**-u** : "mouvement"

dénominal :	<i>ba'a</i> (eau)	+	<i>-u</i>	=	<i>ba'au</i> (extraire un certain objet d'un liquide)
-------------	-------------------	---	-----------	---	---

**-oa** : "processus"

déverbal (emprunts de l'espagnol) : \**piar* (vendre à crédit) + *-oa* = *piaroa* (vendre à crédit)

**-tu** : "faire, convertir", "action provoquée par"

déadjectival :	<i>yo'otui</i> (vieux)	+	<i>-tu</i>	=	<i>yo'otuitu</i> (vieillir)
dénominal :	<i>ko'okoi</i> (chili, piment)	+	<i>-tu</i>	=	<i>ko'okoitu</i> (irriter)

**-tua** : "faire, provoquer"

déadjectival :	<i>yosi</i> (chaud)	+	<i>-tua</i>	=	<i>yositua</i> (réchauffer)
déadverbial :	<i>bepaa</i> (en haut)	+	<i>-tua</i>	=	<i>bepaatua</i> (couvrir)
déverbal :	<i>elpea</i> (aller mieux)	+	<i>-tua</i>	=	<i>elpeatua</i> (soulager)

**-ria** : "faire, provoquer"

déadjectival :	<i>awi</i> (gros)	+	<i>-ria</i>	=	<i>awiria</i> (grossir)
déverbal :	<i>yo'otu</i> (croître)	+	<i>-ria</i>	=	<i>yo'oturia</i> (cultiver)

### 1.1.3. Suffixes adjectivaux

**-la** : "résultat"

déverbal :	<i>biika</i> (aigrir)	+	<i>-la</i>	=	<i>biikala</i> (aigre)
------------	-----------------------	---	------------	---	------------------------

**-i** : "résultat"

déverbal :	<i>etapo</i> (ouvrir)	+	<i>-i</i>	=	<i>etapoi</i> (ouvert)
------------	-----------------------	---	-----------	---	------------------------

**-ri** : "résultat", "qualité"

déverbal :	<i>bato'owa</i> (baptiser)	+	<i>-ri</i>	=	<i>bato'owari</i> (baptisé)
dénominal :	<i>ba'a</i> (eau)	+	<i>-ri</i>	=	<i>ba'ari</i> (humide, mouillé)

### 1.1.4. Suffixes adverbiaux

**-si** : "fois", "manière"

déadjectival :	<i>baji</i> (trois)	+	<i>-si</i>	=	<i>bajisi</i> (trois fois)
dénominal :	<i>ko'oko</i> (douleur)	+	<i>-si</i>	=	<i>ko'okosi</i> (péniblement)

## 1.2. La composition yaqui

Dans la composition, nous avons au moins deux unités lexicales qui peuvent apparaître de manière autonome dans la langue tandis que dans la dérivation, l'élément affixal n'a pas ce type d'autonomie. La composition implique par conséquent la conjonction d'au moins deux éléments constitutifs avec capacité pour des emplois linguistiques indépendants, c'est-à-dire des mots que l'on peut trouver de manière isolée dans la langue. Tout comme la dérivation, la composition peut mettre en marche différents procédés qui peuvent être regroupés en deux types de formation : la formation lexicale qui crée des unités lexicales minimales (un mot) et la formation syntagmatique dont les créations correspondent à des unités lexicales plus grandes (plusieurs mots). Dans les deux cas, le nouvel élément ainsi formé se définit à travers la dénotation des mots du composé.

Etant données les caractéristiques du programme informatique que nous voulons illustrer et qui se circonscrit aux limites graphiques du mot, nous allons traiter exclusivement la composition lexicale et nous ne considérerons pas alors les composés syntagmatiques. Toutefois, il est important de rappeler que la langue yaqui est une langue avec peu de tradition écrite et que, par conséquent, les limites entre les deux types de composés ne sont pas claires. En effet, du fait que les critères orthographiques du système d'écriture de la langue yaqui ne sont pas totalement homogènes, ni pleinement établis et par là-même, acceptés, déterminer si un composé est le résultat d'une composition lexicale ou d'une composition syntagmatique est bien souvent une décision problématique<sup>5</sup>. Cette situation n'affecte pas toutefois la présentation du traitement automatique de la lexicogénèse combinatoire du yaqui que nous prétendons mener à bien.

### 1.2.1. La composition lexicale yaqui

La composition lexicale apparaît à travers le procédé d'agglutination en vertu duquel on unit au moins deux mots pour n'en former qu'un seul qui fonctionne alors comme une nouvelle unité de signification. Les unités lexicales simples ainsi créées que Benveniste a appelées conglomerats (Benveniste 1977: 172), se caractérisent donc par la fusion des éléments participants dans le composé, fusion qui apparaît complète dans la graphie tout comme le processus de lexicalisation et de grammaticalisation du composé.

---

<sup>5</sup> Un exemple de ce manque de consensus en ce qui concerne les composés, apparaît pour le concept "miel d'abeille" qui, selon Dedrick et Casad (1999: 158), a comme signifiant yaqui *mumuba'awa*, composé lexical à partir des formes *muumu* "abeille" et *ba'awa* "bouillon, jus" tandis que, pour Estrada *et al.* (2004), la forme yaqui correspondante est le résultat d'une composition syntagmatique *mumu ba'awa*.

La consultation de notre corpus d'étude nous a permis d'identifier les exemples suivants que nous avons classés selon la catégorie grammaticale du composé et des éléments constitutifs.

#### 1.2.1.1. Composition nominale

- **N + N** : *mamam* (mains) + *pusiam* (doigts) = *mampusiam* (doigts de la main)

#### 1.2.1.2. Composition verbale

- **V + V** : *ji'ika* (coudre) + *butta* (détacher) = *ji'ikbutta* (découdre)

- **N + V** : *mamam* (mains) + *tojte* (décolorer, déteindre) = *mamtojte* (applaudir)

- **ADJ + V** : *tata* (chaud) + *bujte* (monter en épi) = *tatabujte* (transpirer)

- **ADV + V** : *nat* (à côté de) + *temae* (interroger) = *nattetae* (demander)

#### 1.2.1.3. Composition adjectivale:

- **N + ADJ** : *ba'a* (eau) + *yejtela* (debout) = *ba'ayejtela* (mou)

#### 1.2.1.4. Composition adverbiale:

- **ADV + ADV** : *bat* (avant, en premier) + *tuuka* (hier) = *battuuka* (avant-hier)

Nous pouvons remarquer que la composition lexicale en yaqui permet de former aussi bien des substantifs que des verbes, adjectifs ou adverbes. Contrairement à la suffixation, ce mécanisme ne paraît pas être très productif en yaqui si nous nous basons sur le peu d'exemples identifiés. En outre, les composés yaquis par agglutination que nous avons mis en évidence dans le corpus d'étude utilisent seulement deux éléments lexicaux et la catégorie grammaticale du composé paraît être déterminée par l'élément lexical de droite. Selon les exemples enregistrés, le composé a en effet la même catégorie grammaticale que l'élément lexical situé en seconde position. Quant à la relation sémantique entre les éléments du composé, les données laisseraient penser que nous avons une relation déterminant + déterminé.

### 1.3. La parasyntèse yaqui

Les phénomènes de composition et de suffixation peuvent se combiner dans la parasyntèse. En effet, comme le signale Alvar Ezquerro (1993: 64), la parasyntèse consiste en un processus de dérivation d'un mot composé. Le corpus étudié nous fournit l'exemple de mot parasyntétique suivant :

[*tata* (chaud) + *bujte* (monter en épi)] + -*ia* (suffixe nominal) = *tatabujtia* (sueur)

Il est important de signaler que cette brève présentation des mécanismes de suffixation et de composition de la langue yaqui ne prétend être ni définitive, ni exhaustive. Toutefois, elle nous permet d'avoir une vision générale de la lexicogénèse combinatoire du yaqui et, à travers les exemples de mots suffixés et composés de la langue yaqui que nous avons présentés, nous allons à présent illustrer un programme de traitement automatique de la morphologie combinatoire yaqui.

## **2. La modélisation informatique**

Dans cette partie, nous exposerons après un bref commentaire sur les présupposés linguistiques du système, les principes de formalisation de la ZStation pour, dans une troisième partie, mentionner ses réalisations informatiques appliquées à la lexicogénèse du yaqui.

### **2.1. Présupposés linguistiques**

Dans le cadre d'une étude sur les néologismes, le professeur H. Zinglé (1996) a développé un ensemble de processus qui permettent de représenter le sens des mots dérivés et composés et, de manière symétrique, de les produire à partir d'une représentation sémantique. L'étude s'est limitée aux cas de dérivation et composition calculables, en privilégiant une analyse cognitive fondée sur le sémantisme des unités morphologiques et le sémantisme des relations entre concepts. La perspective adoptée unit dérivation et composition en introduisant comme seuls critères typologiques, la capacité des unités morphologiques à admettre une expansion à droite et à gauche et la possibilité ou non de régir la construction. Ainsi, l'unité lexicale dérivée est considérée comme le résultat de la combinaison d'un élément lexical pivot et d'un ou plusieurs affixes qui sont ou des préfixes ou des suffixes selon qu'ils soient associés avec l'élément ultérieur ou précédent. La distinction est uniquement établie sur la base de critères formels de combinaison :

- élément lexical pivot : combinaison à droite ou à gauche
- préfixe : combinaison exclusive à droite
- suffixe : combinaison exclusive à gauche.

Nous pouvons signaler que l'élément situé à droite ou à gauche de l'affixe n'est pas forcément un élément lexical pivot, mais il peut être un autre affixe. Cependant, la construction doit posséder au moins un élément lexical pivot.

Contrairement à l'unité lexicale dérivée, l'unité lexicale composée est constituée par plusieurs éléments lexicaux pivots. Une unité morphologique sera alors représentée par tout élément formatif, c'est-à-dire



tout élément lexical ou affixal qui entre dans la formation d'une unité composée, dérivée ou simple.

Le corpus étudié est constitué par les mots yaquis suivants qui nous ont servis à illustrer la lexicogénèse combinatoire de cette langue :

*kiawa, ba'awa, etbwawame, nenkame, jiyokareo, awiraa, etejoi, ko'okoi, chuyuria, tasia, bwe'ute, ba'ate, ba'au, piaroa, yo'otuitu, ko'okoitu, yositua, bepaatua, elpetua, awiria, yo'oturia, biikala, etapoi, bato'owari, ba'ari, bajisi, ko'okosi, mampusiam, ji'ikbutta, mamtojte, tatabujte, nattemae, ba'ayejtela, battuuka, tatabujtia.*

De l'analyse du corpus précédent, nous avons identifié les unités morphologiques suivantes de la langue yaqui :

**éléments lexicaux pivots:** *kia, ba'a, etbwa, nenka, jiyoka, awi, etejo, ko'oko, chuyu, tas, bwe'u, piar, yosi, bepaa, elpea, yo'otu, biika, etapo, bato'owa, baji, mam, pusiam, ji'ik, butta, tojte, tata, bujte, nat, temae, yejte, bat, tuuka, bujt*

**suffixes:** *-wa, -wame, -me, -reo, -raa, -i, -ria, -ia, -te, -u, -oa, -tu, -tua, -la, -ri, -si*

## 2.2. Principes de formalisation

### 2.2.1. Le programme ZStation

Le LILLA a été créé en 1988 avec l'orientation principale d'appliquer les contributions de la recherche en linguistique à la résolution de problèmes concrets en rapport avec les industries de la langue. Pour atteindre cet objectif, le professeur Henri Zinglé, directeur et fondateur du LILLA, a élaboré un programme appelé ZStation qui permet la modélisation des connaissances linguistiques (Zinglé 1994).

La ZStation est un environnement d'ingénierie linguistique qui fonctionne comme un atelier de tâches linguistiques qui permet la définition et l'expérimentation de la connaissance linguistique. La formalisation s'appuie sur l'approche générale sur laquelle est fondée la ZStation (Zinglé 1994). Ainsi, la distinction méthodologique établie entre le niveau interlinguistique (interlinguistic knowledge) et le niveau intralinguistique (intralinguistic knowledge) permet de dissocier les caractéristiques formelles de la dérivation et de la composition de leurs caractéristiques sémantiques. De cette manière, même si les caractéristiques formelles du signifiant peuvent varier d'une langue à une autre, cela n'affecte pas le transfert interlinguistique car celui-ci s'effectue sur la base du signifié et non du signifiant. La formalisation lexicale se limite alors à expliciter les relations entre les caractéristiques intralinguistiques et interlinguistiques.

Dans cette optique, l'objectif du processus d'analyse sera d'associer à une unité lexicale dérivée ou composée une représentation sémantique indépendante des langues et le processus de génération consistera à l'inverse à exprimer le contenu d'une représentation sémantique en unité lexicale. Ici, apparaît justement la question du type de formalisation sémantique à choisir. Fondamentalement, la ZStation est fondée sur la représentation des connaissances à l'aide de graphes conceptuels (Sowa 1984) qui permettent d'exprimer les caractéristiques intrinsèques des concepts et les relations qui les unissent. Ainsi, un graphique conceptuel de base est composé de deux concepts unis par une relation [Concept 1] → (Relation) → [Concept 2]<sup>6</sup>.

Ce programme peut alors être vu comme un système de traduction automatique avec une analyse de type pivot qui est composée de deux phases analyse/génération et qui est orienté vers les réseaux sémantiques utilisant les graphes conceptuels comme représentation interlangue.

La ZStation fonctionne ainsi à travers la modélisation des connaissances linguistiques grâce à deux niveaux :

- le niveau intralinguistique qui concerne l'analyse et la génération des langues naturelles. Le linguiste a ici la possibilité d'éditer des dictionnaires et des grammaires qu'il pourra ensuite utiliser pour l'analyse et la création du langage naturel.
- le niveau interlinguistique qui permet d'éditer la connaissance linguistique et de faire une compilation vers des réseaux sémantiques ou des bases de données pour les graphes conceptuels.

### **2.2.2. Analyse et génération automatiques**

La ZStation peut par conséquent apporter une aide à la localisation et la création de néologismes grâce à un système à bases de connaissances. L'idée développée dans le cadre des recherches du LILLA considère que les deux processus (analyse/génération) peuvent être menés à bien à partir d'une seule et même base de connaissances dans laquelle sont explicitées les connaissances morphologiques (propriétés formelles situées dans le niveau intralinguistique) et sémantiques (propriétés conceptuelles situées dans le niveau interlinguistique) des unités linguistiques impliquées dans la dérivation ou dans la composition (Zinglé 1996).

---

<sup>6</sup> Par exemple, le graphe conceptuel de « enregistreur » serait [enregistrer] → (instrument) → [appareil] et il pourrait se lire comme « appareil qui est l'instrument utilisé pour enregistrer ».

Le processus d'analyse apparaît alors comme un calcul qui prend en considération ces connaissances pour obtenir la construction d'une représentation de nature sémantique. On associe donc à une unité lexicale dérivée ou composée une représentation sémantique indépendante des langues. Le processus inverse de génération apparaît aussi comme un calcul qui va permettre de dériver ou de composer un mot à partir d'une représentation sémantique du même type.

### 2.3. Réalisation informatique

L'analyse et la génération automatiques de mots dérivés constituent une des applications possibles de la ZStation. Comme toutes les applications de cet environnement, elles font appel à un dictionnaire structuré d'unités morphologiques, constitué au moyen d'un éditeur de ressources linguistiques. L'analyse et la génération automatiques utilisent ainsi la même information source contenue dans une base de connaissances commune.

#### 2.3.1. Élaboration du dictionnaire d'unités morphologiques

Le dictionnaire d'unités morphologiques est structuré en articles consacrés chacun à un élément formatif. Chaque article est constitué par une ou plusieurs rubriques, selon que l'unité morphologique donne lieu à une ou plusieurs analyses différentes.

Chaque rubrique est composée de 4 champs :

- < Champ de classe >
- < Champ sémantique >
- < Champ morpho-syntaxique >
- < Champ structural >

Le *champ de classe* permet d'explicitier si l'élément formatif est un lexème, un préfixe ou un suffixe.

Le *champ sémantique* est occupé par une formule qui exprime la pile<sup>7</sup> de concepts objet de l'évaluation sémantique. Les identificateurs de concepts sont librement choisis par l'utilisateur. Nous pouvons distinguer trois cas :

- la pile est réduite à un concept unique : *ba'a* → *agua0*<sup>8</sup>
- la pile de concepts est pré-définie : *elpea* → *sentir0(alivio0)*

<sup>7</sup> Il s'agit d'un terme informatique utilisé dans le programme ZStation, terme repris des travaux de H. Zinglé.

<sup>8</sup> Les identificateurs de concepts ont été capturés en espagnol afin de faciliter l'utilisation de cet outil informatique à la communauté yaqui, dont la plupart des membres sont bilingues (yaqui-espagnol).

– la pile de concepts comporte une pile argument constituée par les conditions de combinaison du concept et marquée formellement par (\*) :  
*-reo* → agente0(\*)

Les deux premiers cas concernent en règle générale les éléments lexicaux, le troisième cas est spécifique des préfixes et des suffixes mais peut aussi affecter les lexèmes qui gouvernent la structure du composé lexical.

Le *champ morpho-syntaxique* sert à déterminer la catégorie de l'élément lexical formé. Comme en yaqui la catégorie du dérivé est donnée par le suffixe et celle du composé par l'élément lexical situé à droite, on indiquera alors pour les suffixes et les lexèmes correspondants les catégories grammaticales pertinentes (voir infra, liste de lexèmes). Au contraire, ce champ n'aura aucune information pour les lexèmes qui ne déterminent pas la catégorie grammaticale de l'unité lexicale formée. Dans ces cas, nous l'indiquerons de manière explicite avec l'identificateur < nil >.

Le *champ structural* permet d'explicitier les restrictions formelles et sémantiques attachées aux piles de concepts associés à une pile argument (principalement les suffixes et les lexèmes qui déterminent la catégorie grammaticale du dérivé et du composé lexical). Ainsi, quand une même unité morphologique produit des évaluations sémantiques différentes, on peut déterminer les obligations sémantiques qui permettent de les différencier. Notre corpus de mots yaquis nous fournit, entre autres, l'exemple du suffixe *-wa* (voir infra, liste de lexèmes), auquel nous pouvons associer la pile *cualidad\_abstracta0(\*)* ou la pile *parecido\_a0(\*)*. Dans les deux cas, l'élément formatif argument doit être un lexème. Dans *cualidad\_abstracta0(\*)*, la pile argument est validée si le lexème argument est associé au concept de qualité (*cualidad0*) tandis qu'en *parecido\_a0(\*)*, le lexème sera seulement associé avec le concept d'objet (*objeto0*). Les restrictions sémantiques sont notées à l'aide du formalisme relationnel adopté pour l'élaboration d'ontologies dans la ZStation. Ainsi, *conds=[r(1, isa, +cualidad0)]* indique que, dans le calcul structural, la pile argument peut être associée avec le concept de qualité. Dans le sous-champ *conds*, peut figurer un ensemble de relations sémantiques.

L'association du suffixe *-wa* à la pile *cualidad\_abstracta0(\*)* sera ainsi liée à la condition *r(1, isa, +cualidad0)* et son association avec la pile *parecido\_a0(\*)* à la condition *r(1, isa, +objeto0)*. Dans le premier cas, le suffixe sera seulement associé à un lexème qui se réfère à une qualité et dans le second cas, à un lexème qui dénote un objet. Les propriétés conceptuelles associées aux lexèmes sont capturées dans les ontologies (voir infra).

Le dictionnaire d'unités morphologiques des mots yaquis de notre corpus est constitué par les éléments formatifs suivants :

### Lexèmes

#### awi

{CLS=lex  
SEM=gordo0  
CAT=nil  
arg(cls=nil,conds=[])

#### ba'a

{CLS=lex  
SEM=agua0  
CAT=nil  
arg(cls=nil,conds=[])

#### baji

{CLS=lex  
SEM=tres0  
CAT=nil  
arg(cls=nil,conds=[])

#### bat

{CLS=lex  
SEM=antes0  
CAT=nil  
arg(cls=nil,conds=[])

#### bato'owa

{CLS=lex  
SEM=bautizar0  
CAT=nil  
arg(cls=nil,conds=[])

#### bepaa

{CLS=lex  
SEM=arriba0  
CAT=nil  
arg(cls=nil,conds=[])

#### biika

{CLS=lex  
SEM=agriarse0  
CAT=nil  
arg(cls=nil,conds=[])

#### bujt

{CLS=lex  
SEM=espigar0  
CAT=nil  
arg(cls=nil,conds=[])

#### bujte

{CLS=lex  
SEM=espigar0(\*)  
CAT=verb  
arg(cls=nil,conds=[r(1,isa  
,+cualidad0)])

#### butta

{CLS=lex  
SEM=desamarrar0(\*)  
CAT=verb  
arg(cls=nil,conds=[r(1,isa  
,+acción0)])

#### bwe'u

{CLS=lex  
SEM=grande0  
CAT=nil  
arg(cls=nil,conds=[])

#### chuyu

{CLS=lex  
SEM=empacharse0  
CAT=nil  
arg(cls=nil,conds=[])

#### elpea

{CLS=lex  
SEM=sentir0(alivio0)  
CAT=nil  
arg(cls=nil,conds=[])

#### etapo

{CLS=lex  
SEM=abrir0  
CAT=nil  
arg(cls=nil,conds=[])

#### etbwa

{CLS=lex  
SEM=robar0  
CAT=nil  
arg(cls=nil,conds=[])

#### etejo

{CLS=lex  
SEM=platicar0  
CAT=nil  
arg(cls=nil,conds=[])

#### ji'ik

{CLS=lex  
SEM=coser0  
CAT=nil  
arg(cls=nil,conds=[])

#### jiyoka

{CLS=lex  
SEM=pintar0  
CAT=nil  
arg(cls=nil,conds=[])

#### kia

{CLS=lex  
SEM=sabroso0  
CAT=nil  
arg(cls=nil,conds=[])

#### ko'oko

{CLS=lex  
SEM=dolor0  
CAT=nil  
arg(cls=nil,conds=[])

#### mam

{CLS=lex  
SEM=manos0  
CAT=nil  
arg(cls=nil,conds=[])

#### nat

{CLS=lex  
SEM=además0  
CAT=nil  
arg(cls=nil,conds=[])

#### nenka

{CLS=lex  
SEM=vender0  
CAT=nil  
arg(cls=nil,conds=[])

#### piar

{CLS=lex  
SEM=fiar0  
CAT=nil  
arg(cls=nil,conds=[])

#### pusiam

{CLS=lex  
SEM=dedos0(\*)  
CAT=sub  
arg(cls=lex,conds=[r(1,isa  
,+objeto0)])

#### tata

{CLS=lex  
SEM=caliente0  
CAT=nil  
arg(cls=nil,conds=[])

#### tas

{CLS=lex  
SEM=toser0  
CAT=nil  
arg(cls=nil,conds=[])

**temae**  
 {CLS=lex  
 SEM=interrogar0(\*)  
 CAT=verb  
 arg(cls=lex,conds=[  
 r(1,isa,+manera0)])

**tojte**  
 {CLS=lex  
 SEM=desteñirse(\*)  
 CAT=verb  
 arg(cls=lex,conds=[  
 r(1,isa,+objeto0)])

**tuuka**  
 {CLS=lex  
 SEM=ayer0(\*)  
 CAT=adv  
 arg(cls=lex,conds=[  
 r(1,isa,+manera0)])

**yejte**  
 {CLS=lex  
 SEM=levantar0(\*)  
 CAT=verb  
 arg(cls=lex,conds=[  
 r(1,isa,+objeto0)])

**yo'otu**  
 {CLS=lex  
 SEM=crecer0  
 CAT=nil  
 arg(cls=nil,conds=[])

**yosi**  
 {CLS=lex  
 SEM=caliente0  
 CAT=nil  
 arg(cls=nil,conds=[])

On peut observer que dans les cas des lexèmes qui apparaissent dans la composition lexicale comme élément situé à droite, nous avons spécifié, conformément à l'analyse linguistique préalable, la catégorie grammaticale de l'élément composé obtenu et les restrictions formatives qui doivent être satisfaites pour la correcte combinaison lexématique et pour obtenir ainsi les composés du corpus.

### Suffixes

**i**  
 {CLS=suff  
 SEM=resultado0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

{CLS=suff  
 SEM=causante0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+cualidad0)])

{CLS=suff  
 SEM=resultado0(\*)  
 CAT=adj  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

**la**  
 {CLS=suff  
 SEM=resultado0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

**ia**  
 {CLS=suff  
 SEM=resultado0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

**me**  
 {CLS=suff  
 SEM=agente0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

**oa**  
 {CLS=suff  
 SEM=\*  
 CAT=verb  
 arg(cls=lex,conds=[r(1,char,+préstamo0)])

**raa**  
 {CLS=suff  
 SEM=cualidad\_abstracta0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+cualidad0)])

**reo**  
 {CLS=suff  
 SEM=agente0(\*)  
 CAT=sub  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

**ri**  
 {CLS=suff  
 SEM=resultado0  
 CAT=adj  
 arg(cls=lex,conds=[r(1,isa,+acción0)])

{CLS=suff  
 SEM=resultado0(\*)  
 CAT=adj  
 arg(cls=lex,conds=[r(1,isa,+objeto0)])

{CLS=suff  
 SEM=cualidad0(\*)  
 CAT=adj  
 arg(cls=lex,conds=[r(1,isa,+objeto0)])

```

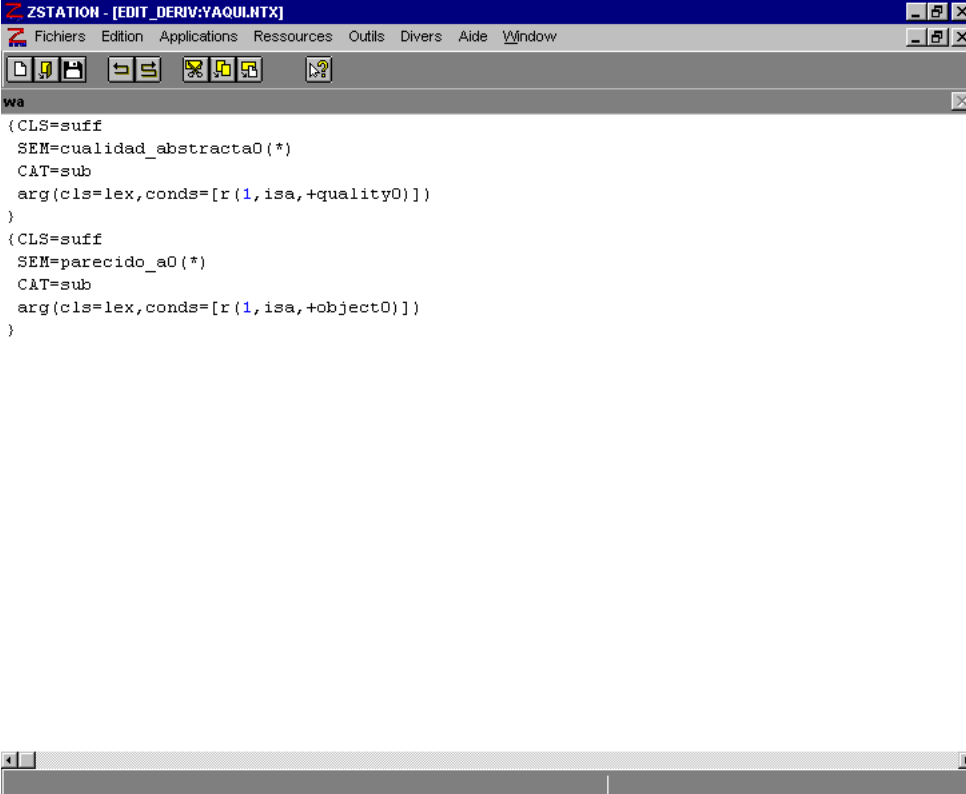
ria
{CLS=suff
SEM=causar0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+cualidad0)])
{CLS=suff
SEM=resultado0(*)
CAT=sub
arg(cls=lex,conds=[r(1,isa,+acción0)])
{CLS=suff
SEM=causar0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+acción0)])
si
{CLS=suff
SEM=veces0(*)
CAT=adv
arg(cls=lex,conds=[r(1,char,+número0)])
{CLS=suff
SEM=manera0(*)
CAT=adv
arg(cls=lex,conds=[r(1,isa,+sensación0)])
te
{CLS=suff
SEM=convertir0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+cualidad0)])
{CLS=suff
SEM=convertir0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+objeto0)])
tua
{CLS=suff
SEM=causar0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+cualidad0)])
{CLS=suff
SEM=causar0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+manera0)])
tu
{CLS=suff
SEM=convertir0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+cualidad0)])
{CLS=suff
SEM=acción_provocado_por0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+objeto0)])
u
{CLS=suff
SEM=movimiento0(*)
CAT=verb
arg(cls=lex,conds=[r(1,isa,+objeto0)])
wa
{CLS=suff
SEM=cualidad_abstracta0(*)
CAT=sub
arg(cls=lex,conds=[r(1,isa,+cualidad0)])
{CLS=suff
SEM=parecido_a0(*)
CAT=sub
arg(cls=lex,conds=[r(1,isa,+objeto0)])
wame
{CLS=suff
SEM=resultado0(*)
CAT=sub
arg(cls=lex,conds=[r(1,isa,+acción0)])

```

Nous pouvons signaler que certains des suffixes du yaqui qui ici apparaissent sont polysémiques, c'est-à-dire qu'ils ont des évaluations sémantiques différentes et que nous avons même des cas de suffixes qui peuvent former des mots avec des catégories grammaticales différentes (le suffixe nominal et adjectival *-i* et le suffixe nominal et verbal *-ria*). Les caractéristiques formelles et sémantiques qui restreignent les options de suffixation apparaissent alors dans le champ structural, avec les sous-champs de classe (*cls*) et de conditions (*conds*). Dans le sous-champ *conds*, on a donné priorité à l'explicitation des relations sémantiques avec l'identificateur de relation *isa* sauf dans les cas du suffixe verbal *-oa* et du suffixe adverbial *-si*, (seulement dans sa valeur sémantique de "fois") où on a utilisé l'identificateur de relation *char* qui permet de restreindre

formellement la dérivation aux lexèmes qui sont des emprunts verbaux de l'espagnol dans le premier cas et des adjectifs numéraux dans le second cas. Ces restrictions formatives devront coïncider avec les conditions des lexèmes capturées dans l'ontologie pour qu'on puisse avoir ce type de combinaison morphématique.

À la suite, nous indiquons un exemple d'édition d'article avec l'aide de l'éditeur de ressources spécifiques de la ZStation. L'exemple correspond au suffixe *-wa*.



```

ZSTATION - [EDIT_DERIV:YAQUILNTX]
Fichiers Edition Applications Ressources Outils Divers Aide Window
wa
{CLS=suff
SEM=cualidad_abstracta0(*)
CAT=sub
arg(cls=lex,conds=[r(1, isa, +quality0)])
}
{CLS=suff
SEM=parecido_a0(*)
CAT=sub
arg(cls=lex,conds=[r(1, isa, +object0)])
}

```

L'élaboration d'ontologies apparaît alors comme indispensable pour la prise en compte des restrictions sémantiques. Principalement, une ontologie est représentée par un treillis de concepts de type hiérarchique multidimensionnel. Chaque concept à l'intérieur de ce treillis est relié à ses hyperonymes par des relations typées, définies par l'ingénieur linguiste qui les choisit en fonction des connaissances à modéliser.

Les propriétés conceptuelles sont par conséquent formalisées au moyen de réseaux multidimensionnels à héritage multiple où chaque élément du treillis est caractérisé par un ensemble de relations conceptuelles ce qui permet de connaître à tout moment l'ensemble de propriétés d'un concept donné ou de vérifier s'il est en mesure de satisfaire une restriction sémantique donnée.



Ces relations sont exprimées à l'aide du nuplet r(Valeur, Relation, Concept) où Valeur indique s'il s'agit d'une relation obligatoire (Valeur=1) ou facultative (Valeur=0); Relation représente l'identificateur de relation que l'utilisateur a choisi (*isa, partOf, cont, char, etc.*) et Concept un identificateur de concept précédé de + o -, selon son assertion ou réfutation. La réfutation intervient essentiellement quand un concept est associé à un hyperonyme dont il n'hérite pas toutes les propriétés<sup>9</sup>. Le mécanisme utilisé est un mécanisme de multihéritage transitif qui inclut la propagation de la négation sur les hyponymes; on donne la priorité à la négation quand un concept hérite de ses hyperonymes des propriétés contradictoires. La transitivité permet d'obtenir les propriétés éventuellement héritées par l'hyponyme d'un concept. Ce double mécanisme de multihéritage transitif nous permet de connaître l'ensemble des propriétés d'un concept ou bien de savoir si un concept répond à une propriété donnée. Fondé sur la programmation orientée objets, le résultat est obtenu dans les deux cas par l'envoi d'un message au concept concerné. Pour pouvoir obtenir une validation efficace des restrictions sémantiques, une description minutieuse des différents concepts est donc nécessaire.

Le corpus étudié nous a ainsi permis d'élaborer l'ontologie nécessaire, conformée par la liste de concepts suivants :

CONCEPTS	CONDITIONS	LEXEMES
abrir0	r(1,isa,+acción0)	<i>etapo</i>
además0	r(1,isa,+manera0)	<i>nat</i>
agriarse0	r(1,isa,+acción0)	<i>biika</i>
agua0	r(1,isa,+objeto0)	<i>ba'a</i>
alivio0	r(1,isa,+cualidad0)	<i>elpea</i> sentir0(alivio0)
antes0	r(1,isa,+manera0)	<i>bat</i>
arriba0	r(1,isa,+manera0)	<i>bepaa</i>
ayer0	r(1,isa,+momento0)	<i>tuuka</i>
bautizar0	r(1,isa,+acción0)	<i>bato'owa</i>
caliente0	r(1,isa,+cualidad0)	<i>yosi</i>
caliente0	r(1,isa,+cualidad0)	<i>tata</i>
coser0	r(1,isa,+acción0)	<i>ji'ika</i>
crecer0	r(1,isa,+acción0)	<i>yo'otu</i>
dedos0	r(1,isa,+objeto0)	<i>pusiam</i>
desamarrar0	r(1,isa,+acción0)	<i>butta</i>
desteñirse0	r(1,isa,+acción0)	<i>tojte</i>
dolor0	r(1,isa,+sensación0)	<i>ko'oko</i>
espigar0	r(1,isa,+acción0)	<i>bujte</i>
espigar0	r(1,isa,+acción0)	<i>bujt</i>
empacharse0	r(1,isa,+acción0)	<i>chuyu</i>
fiar0	r(1,isa,+préstamo0)	<i>piar</i>
gordo0	r(1,isa,+cualidad0)	<i>awi</i>
grande0	r(1,isa,+cualidad0)	<i>bwe'u</i>

<sup>9</sup> Par exemple, le pingouin (hyponyme) est un oiseau (hyperonyme) qui n'a pas la capacité de voler.

interrogar0	r(1,isa,+acción0)	<i>temae</i>
manos0	r(1,isa,+objeto0)	<i>mam</i>
pintar0	r(1,isa,+acción0)	<i>jiyoka</i>
platicar0	r(1,isa,+acción0)	<i>etejo</i>
robar0	r(1,isa,+acción0)	<i>etbwa</i>
sabroso0	r(1,isa,+cualidad0)	<i>kia</i>
levantar0	r(1,isa,+acción0)	<i>yejte</i>
sentir0	r(1,isa,+acción0)	<i>elpea</i> sentir0(alivio0)
toser0	r(1,isa,+acción0)	<i>tas</i>
tres0	r(1,char,+número0)	<i>baji</i>
vender0	r(1,isa,+acción0)	<i>nenka</i>

En tenant compte du fait que les éléments lexicaux sont les éléments pivots dans la construction lexicogénétique, dans l'ontologie il apparaîtra uniquement les restrictions sémantiques des concepts des lexèmes dont les conditions doivent coïncider, pour permettre la combinaison formative, avec les conditions présentes dans le champ structural des suffixes ou des lexèmes impliqués dans la composition (voir 2.3.1 supra). Nous pouvons remarquer que les conditions qui utilisent l'identificateur de relation *isa* se réfèrent à des restrictions sémantiques mais aussi indirectement à des conditions structurales (qui obéissent aux possibilités combinatoires identifiées dans l'analyse linguistique) car le sémantisme d'action correspond avec la catégorie verbale, celui de qualité avec la catégorie adjectivale, ceux d'objet, de sensation et de moment avec la catégorie nominale et celui de manière avec la catégorie adverbiale. Dans tous les cas, nous avons privilégié des valeurs obligatoires parce que l'analyse linguistique effectuée jusqu'à présent ne nous permet pas d'attribuer une plus grande flexibilité formative à ces formants du yaqui.

### 2.3.2. Analyse et génération automatiques

Les processus d'analyse et génération automatiques ont été implantés à l'origine en Pdc-Prolog (version Windows 4.1). Ils sont fondamentalement indépendants des langues naturelles puisque les stratégies de traitement de chaque langue sont codifiées dans la base de données morphologiques. Il s'agit de favoriser au maximum l'interactivité pour permettre à l'ingénieur linguiste de passer facilement de la formalisation des données linguistiques à l'analyse ou à la génération.

#### 2.3.2.1. Analyse automatique de mots dérivés ou composés

L'interface d'analyse se présente comme une fenêtre de dialogue activée à partir du menu de la ZStation (Application/Analyse/ Analyse lexicale). Elle contient :

- une zone de saisie (pour l'acquisition du mot à analyser)
- une zone de visualisation du résultat de l'analyse

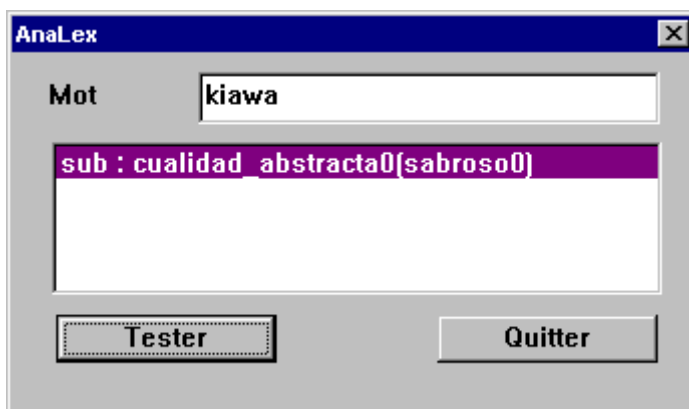
- un bouton pour activer l'analyse après la saisie
- un bouton pour sortir de l'interface.

Le processus d'analyse utilise d'une part un dictionnaire compilé d'unités morphologiques (ressource intralinguistique) et une ontologie compilée (ressource interlinguistique) et il se déroule selon les phases suivantes :

- repérage des unités morphologiques dans le mot à analyser
- construction d'hypothèses sur la base des unités morphologiques repérées
- validation des hypothèses (en particulier des restrictions sémantiques).

L'analyse automatique se sert par conséquent d'un générateur d'hypothèse qui permet de construire une hiérarchisation des unités morphologiques détectées. Pour chaque construction où sont impliquées deux unités morphologiques, on vérifie si les restrictions formelles et sémantiques sont satisfaites, en fonction de la combinaison à gauche ou à droite.

Nous montrons à la suite un échantillon de l'interface d'analyse avec le résultat obtenu pour le mot *kiawa* 'saveur' :



L'analyse automatique des mots yaquis contenus dans notre corpus nous permet d'obtenir les descriptions sémantiques suivantes :

– Mots suffixés:

<b>kiawa</b>	sub:	cualidad_abstracta0(sabroso0)
<b>ba'awa</b>	sub:	parecido_a0(agua0)
<b>etbwawame</b>	sub:	resultado0(robar0)
<b>nenkame</b>	sub:	agente0(vender0)
<b>jiyokareo</b>	sub:	agente0(pintar0)
<b>awiraa</b>	sub:	cualidad0(gordo0)
<b>etejoi</b>	sub:	resultado0(platicar0)
	adj:	resultado0(platicar0)
<b>ko'okoi</b>	sub:	causante0(dolor0)
<b>chuyuria</b>	sub:	resultado0(empacharse0)
	vb:	causar0(empacharse0)

<b>tasia</b>	sub:	resultado0(toser0)
<b>bwe'ute</b>	vb:	convertir0(grande0)
<b>ba'ate</b>	vb:	convertir0(agua0)
<b>ba'au</b>	vb:	movimiento0(agua0)
<b>piaroa</b>	vb:	fiar0
<b>yo'otuitu</b>	vb:	convertir0(resultado0(crecer0))
<b>ko'okoitu</b>	vb:	acción_provocada_por0(causante0(dolor0))
<b>yositua</b>	vb:	causar0(caliente0)
<b>bepaatua</b>	vb:	causar0(arriba0)
<b>elpeatua</b>	vb:	causar0(sentir0(alivio0))
<b>awiria</b>	vb:	causar0(gordo0)
<b>yo'oturia</b>	sub:	resultado0(crecer0)
	vb:	causar0(crecer)
<b>biikala</b>	adj:	resultado0(agriarse0))
<b>etapoi</b>	adj:	resultado0(abrir0)
<b>bato'owari</b>	adj:	resultado0(bautizar0)
<b>ba'ari</b>	adj:	resultado0(agua0)
	adj:	cualidad0(agua0)
<b>ko'okosi</b>	adv:	manera0(dolor0)
<b>bajisi</b>	adv:	veces0(tres0)

– Mots composés:

<b>mampusiam</b>	sub:	dedos0(manos0)
<b>ji'ikbutta</b>	vb:	desamarrar0(coser0)
<b>mamtojte</b>	vb:	desteñirse0(manos0)
<b>tatabujte</b>	vb:	espigar0(caliente0)
<b>nattemae</b>	vb:	interrogar0(además0)
<b>ba'ayejtela</b>	adj:	resultado0(levantar0(agua0))
<b>battuuka</b>	adv:	ayer0(antes0)

– Mot parasyntétique:

<b>tatabujtia</b>	sub:	resultado0(espigar0(caliente0))
-------------------	------	---------------------------------

### 2.3.2.2. Génération automatique de mots dérivés ou composés

L'interface de génération se présente également comme une fenêtre de dialogue activée à partir du menu de la ZStation (Application/ Génération/ Génération lexicale). Elle utilise les mêmes ressources que l'analyse et contient :

- une zone de saisie (pour l'acquisition de la description sémantique du mot à générer)
- une liste d'édition pour la sélection de la catégorie morphosyntaxique du mot à créer
- une liste de concepts identifiés dans le dictionnaire d'unités morphologiques
- une zone de visualisation du résultat de la génération
- un bouton pour activer l'analyse après la saisie
- un bouton pour sortir de l'interface.

Ce processus de génération est composé des phases suivantes :

- repérage des concepts utilisés dans la pile de concepts décrivant l'information que doit exprimer le mot à créer
- recherche des unités morphologiques qui peuvent être associées aux concepts repérés
- construction d'hypothèses sur la base des concepts repérés et des unités morphologiques associées
- validation des hypothèses.

Le processus de génération automatique, se déroulant de manière inverse et avec les mêmes informations que le processus d'analyse, s'initie à partir des descriptions sémantiques qui permettront ainsi de produire le mot souhaité.

Les deux exemples suivants de génération correspondent aux substantifs *kiawa* 'saveur' et *ba'awa* 'bouillon'.



Le programme permet évidemment (c'est un de ses objectifs) de produire ou d'analyser des mots nouveaux à partir de l'activation d'une même base de connaissances. Ainsi, avec les informations linguistiques

saisies, le système fournit les exemples suivants de candidats lexicaux combinés du yaqui, exemples que nous accompagnons avec l'analyse effectuée par le système et le possible concept correspondant en espagnol :

<b>yo'otuwame, yo'oturia</b>	sub:	resultado0(crecer0)	“vejez”
<b>jiyokawame, jiyokaria</b>	sub:	resultado0(pintar0)	“pintura”
<b>piaroame, piaroareo</b>	sub:	agente0(fiar0)	“fiador”
<b>yosituame, yosituareo</b>	sub:	agente0(causar0(caliente0))	“calentador”
<b>yo'otuitume, yo'otuitureo</b>	sub:	agente0(convertir0(resultado0(crecer0)))	“envejecedor”
<b>biikalaraa</b>	sub:	cualidad0(resultado0(agriarse0))	“agrura”
<b>biikawa</b>	sub:	cualidad_abstracta0(resultado0(agriarse0))	“amargura”
<b>bwe'uraa</b>	sub:	cualidad0(grande0)	“tamaño grande”
<b>bwe'uwa</b>	sub:	cualidad_abstracta0(grande0)	“grandeza”
<b>ko'okote</b>	vb:	convertir0(dolor0)	“lastimar”
<b>ba'aritia, ba'ariria</b>	vb:	causar0(resultado0(agua0))	“mojar”
	vb:	causar0(cualidad0(agua0))	“humedecer”
<b>ba'arite, ba'aritu</b>	vb:	convertir0(resultado0(agua0))	“mojar”
	vb:	convertir0(cualidad0(agua0))	“humedecer”
<b>biikaria</b>	sub:	resultado0(agriarse0)	“agrura, amargura”
	vb:	causar0(agriarse0)	“agriar, amargar”
<b>ko'okoitula, ko'okoitui</b>	adj:	resultado0(acción_provocada_por0(causante0(dolor0)))	“enchilado/a”
<b>ko'okoituri</b>	adj:	cualidad0(acción_provocada_por0(causante0(dolor0)))	“enchiloso/a”
	adj:	resultado0(acción_provocada_por0(causante0(dolor0)))	“enchilado/a”
<b>yosibujte</b>	vb:	espigar0(caliente0)	“sudar”
<b>nattemaeme, nattemaereo</b>	sub:	agente0(interrogar0(además0))	“preguntador”
<b>ji'ikbuttala, ji'ikbuttai</b>	adj:	resultado0(desamarrar0(coser0))	“descocado/a”

### 3. Problèmes et perspectives

Comme tout projet de linguistique appliquée, cette étude prétend à partir d'une approche interdisciplinaire contribuer à la résolution de certains problèmes de la communauté yaqui, problèmes en relation bien évidemment avec leur langue. La proposition d'un outil informatique capable de développer des analyses lexicales automatiques et d'aider à la création de néologismes dans une langue de minorité pourrait ainsi avoir des applications intéressantes dans des domaines liés notamment à la politique et à la planification linguistique de la communauté yaqui (problèmes de revitalisation de la langue, de lutte contre les emprunts linguistiques, de gestion et d'harmonisation terminologique, d'enseignement de la lexicologie yaqui, etc.). Cependant, la future utilisation de cet instrument par la propre communauté yaqui ne peut se faire sans avoir préalablement évalué d'une certaine manière l'outil informatique et d'avoir pris en compte également la réponse de la communauté. La recherche de solutions aux problèmes que l'on mettra alors en évidence permettra sans nul

doute l'amélioration de l'outil proposé ce qui pourrait faciliter en définitive son acceptation et son utilisation au sein de la propre communauté. Les objectifs appliqués poursuivis auraient alors peut-être une chance d'être atteints. Il convient donc, avant de terminer cette présentation, de réfléchir aux problèmes que nous avons pu identifier pour le moment et de signaler les perspectives d'applications entrouvertes par ce type de travail.

### 3.1. Problèmes

Les premiers problèmes que nous voulons aborder nous renvoient à l'analyse linguistique réalisée et aux caractéristiques des mots créés par le système. Les mots fournis par la ZStation que nous avons présentés plus haut (voir 2.3.2.2.) sont apparemment des mots inexistant dans la langue. Ils n'apparaissent ainsi ni dans le Dictionnaire Yaqui-Espagnol d'Estrada *et al.* (2004), ni dans la littérature révisée<sup>10</sup>.

Il est évident qu'une étude plus détaillée des mécanismes de lexicogénèse combinatoire du yaqui est nécessaire pour pouvoir déterminer si ces candidats lexicaux sont linguistiquement acceptables, c'est-à-dire s'ils remplissent toutes les restrictions formatives de la langue yaqui. Pour déterminer ces règles de formation de mots combinés du yaqui, il est nécessaire de considérer outre les aspects strictement morphologiques et sémantiques, d'autres aspects comme par exemple les restrictions morpho-phonologiques des processus combinatoires yaquis et les caractéristiques de la structure syllabique du mot yaqui (patrons et quantités syllabiques). C'est justement dans ce sens que nous avons réalisé un travail sur les principaux processus phonologiques liés à la suffixation de la langue yaqui (Alvarez 2004). Cette étude nous a permis d'identifier trois changements provoqués par la suffixation et nous avons pu énoncer les règles phonologiques qui s'y rattachent et que nous exposons à la suite :

#### 1. Perte de voyelles finales (apocopes vocaliques)

a. /a/  $\Rightarrow$   $\emptyset$  / \_\_# + {-i}

La voyelle /a/ en position finale de la base lexicale disparaît quand on a une suffixation en *-i*.

b. /e/  $\Rightarrow$   $\emptyset$  / \_\_# + {-i; -ri}

La voyelle /e/ en position finale de la base lexicale disparaît quand on a une suffixation en *-i*, *-ri*.

---

<sup>10</sup> Les seuls concepts avec une dénomination yaqui trouvée sont "blesser" qui se dit en yaqui *ko'okosi yaak* (faire péniblement), "mouiller" qui se dit *konomia* et "transpirer" qui apparaît dans notre corpus comme *tatabujte* (Estrada *et al.* 2004).





signifié change car le changement fonctionnel provoque un changement sémantique représenté, le plus souvent, par une relation métonymique entre le sens d'origine et le sens dérivé.

Cette possibilité lexicogénétique devrait également pouvoir être formalisée à travers la ZStation. En définitive, les problèmes rencontrés ici nous renvoient aux possibilités de la modélisation des connaissances permises par la ZStation. Les trois cas exposés (changements morpho-phonologiques, structure syllabique, conversion) montrent en effet que l'outil informatique doit pouvoir intégrer, si l'on veut qu'il soit performant (c'est-à-dire qu'il propose de bons candidats à néologismes), les éléments mis à jour par l'analyse linguistique. L'ingénieur informatique devra ainsi élaborer les programmes nécessaires pour modéliser les connaissances mises en évidence par le linguiste descriptif.

Sur un plan plus social, il est clair que la validation des candidats à néologismes reviendra en dernier lieu à la propre communauté, d'où l'intérêt de consulter les indiens yaquis pour établir le degré d'acceptation et de reconnaissance des candidats lexicaux produits par le système et d'essayer d'établir ainsi des critères d'acceptabilité non seulement linguistique mais aussi socioculturelle. Nous avons pour le moment présenté cet outil à certains membres de la communauté, représentants d'une « commission de néologismes » autochtone, groupe qui a la charge de réfléchir sur les besoins lexicogénétiques du yaqui et de proposer des néologismes yaquis si le besoin s'en fait sentir. Hormis les difficultés liées aux réticences légendaires des indiens yaquis envers les *yoris* (les non-yaquis), réticences qui s'accompagnent généralement d'une résistance<sup>12</sup> à tout apport du « monde blanc », la présentation des capacités d'analyse et de génération automatiques de la ZStation s'est vue accompagnée tantôt de sourires intrigués, tantôt de visages circonspects, et, que trop rarement, par des commentaires précis sur le programme. En ce qui concerne les analyses automatiques réalisées par le système, certains commentaires critiquaient par exemple le peu de lisibilité des formules proposées et pour ce qui est des mots créés par le système, la plupart des mots ont été rejetés par ces locuteurs yaquis, comme ne faisant pas partie de leur langue même s'ils reconnaissaient dans certains cas les formants mis en jeu mais refusaient d'associer un sens à ces mots. Ces attitudes de rejet ne sont pas en soi une condamnation pour l'outil présenté ici mais révèlent sans aucun

---

<sup>12</sup> L'histoire de cette résistance peut être consultée dans le livre *Parlons yaqui* (Estrada & Alvarez 2008: 12-44) ou encore de manière plus approfondie dans l'ouvrage de Cécile Gouy-Guilbert (1983) intitulé *Une résistance indienne, les Yaquis du Sonora*.

doute la difficulté d'établir des relations de confiance et de coopération avec les membres de cette communauté. S'il est vrai que ces réactions de méfiance sont certainement typiques de leurs rapports avec les non-yaquis, il nous appartiendra cependant de gagner peu à peu leur confiance et de poursuivre dans la recherche d'une coopération car c'est dans l'évaluation et l'utilisation de cet outil informatique par la communauté que réside en fait l'objectif appliqué souhaité. Dans cette optique, il faudra rechercher en premier lieu l'appui des autorités locales de la communauté yaqui mais aussi l'aide des gouvernements de l'État de Sonora et, même à un niveau national, du Mexique<sup>13</sup>, instances de décision qu'il conviendra de sensibiliser aux applications possibles de la ZStation, notamment en ce qui concerne les aspects liés à la politique linguistique et éducative des communautés indigènes.

Nous voyons donc que d'une part l'amélioration de cet outil informatique passe inévitablement par une meilleure connaissance de la morphologie lexicale de la langue yaqui et par sa modélisation et représentation informatique et que, d'autre part, son usage au sein de la communauté yaqui devra passer l'écueil de la méfiance, voire de l'indifférence. Ces problèmes ne doivent pas cependant nous faire perdre de vue les perspectives entrouvertes par cette étude. Les applications pratiques peuvent être en effet intéressantes.

### **3.2. Perspectives**

La ZStation permet donc d'avoir un système d'analyse et de génération automatiques de mots dérivés et composés qui utilise une base de connaissances dans laquelle sont explicitées les restrictions sémantiques et formelles des éléments formatifs grâce à une méthode de représentation du sens basée sur la théorie des graphes conceptuels. Cette analyse peut s'adapter à des langues différentes à condition de construire les bases de connaissances nécessaires. Pour cela, notre première tâche a été d'élaborer un dictionnaire d'unités morphologiques du yaqui le plus complet possible tant au niveau morphologique que sémantique. La constitution de cette banque de données morphosémantiques des formants du yaqui s'est également vue accompagnée d'autres travaux de description linguistique qui ont eu pour but d'améliorer notre connaissance de la morphologie

---

<sup>13</sup> Notamment à travers l'Institut National des Langues Indigènes (INALI). Depuis le mois de juillet 2011, nous avons ainsi initié, avec l'appui de cet organisme d'état, un projet de traduction de la Constitution Politique du Mexique, de l'espagnol à trois langues indigènes de la région du Nord-Ouest du Mexique, dont le yaqui. Une partie de ce projet est consacrée aux problèmes de traduction des termes rencontrés et aux possibles utilisations de la ZStation pour la proposition de néologismes en langue yaqui.

lexicale de la langue yaqui, connaissance qui devra être modélisée par la suite afin d'améliorer l'outil informatique proposé. Ces différentes études descriptives concernent notamment l'étude de la structure syllabique du mot yaqui et des différents phénomènes morpho-phonologiques associés à la suffixation (Alvarez 2004) ainsi que l'identification et la description des différents mécanismes de création de mots yaquis (Alvarez & Estrada 2009), étude qui a démontré l'omniprésence de la suffixation dans la lexicogénèse du yaqui et qui justifie d'une certaine manière la pertinence d'un outil qui aide à la création de mots suffixés et composés en yaqui.

Outre l'amélioration de la connaissance de la morphologie lexicale de la langue yaqui, le projet cherche à développer à travers la modélisation linguistique proposée par la ZStation, un outil informatique qui peut interpréter sémantiquement et produire morphologiquement des mots existants et des néologismes en langue yaqui. En suivant sur ce point Richard Sproat (1992: 8), nous pensons que le fait d'appliquer un instrument technologique nouveau de morphologie informatique à une langue qui n'est pas l'anglais ou une des autres langues de circulation internationale, cela aidera sans aucun doute au développement de ce domaine de la linguistique informatique. L'application de ces nouvelles technologies aux langues de minorité constitue donc un enjeu d'envergure non seulement pour le développement du traitement automatique des langues naturelles mais aussi pour les langues concernées et, par voie de conséquence, pour les communautés indigènes qui les parlent. Ainsi, nous croyons que ce travail peut avoir des applications très intéressantes et très prometteuses qui permettront d'aider à résoudre des problèmes sociaux en rapport avec la langue, par exemple en ce qui concerne la gestion et l'harmonisation terminologique en langues indigènes. En effet, les politiques linguistiques et éducatives pour les différentes communautés indigènes du Mexique doivent faire face souvent au problème de la traduction de beaucoup de termes (éducatifs mais aussi économiques, industriels, scientifiques, etc.) de l'espagnol à ces langues minoritaires. Notre projet pourrait alors aider à la génération de néologismes en langues indigènes à travers un système informatique qui proposerait des options de néologismes en se basant sur une analyse morphosémantique préalable des formants de la langue indigène en question, en activant et en favorisant ainsi un dynamisme lexicogénétique utilisant des ressources linguistiques propres. Ce système pourrait ainsi éviter la profusion d'emprunts de l'espagnol et pourrait travailler de la même manière à la revitalisation de cette langue indigène, en fournissant à la communauté yaqui un précieux instrument d'aide pour la génération de néologismes yaquis qui utiliseraient des mécanismes et des lexèmes de cette langue, et en œuvrant de la sorte dans le cadre de l'aménagement lexical des

politiques et de la planification en langue indigène. Il convient de rappeler que la communauté yaqui a constitué justement une commission pour créer des néologismes ce qui montre une sensibilisation à cette problématique lexicogénétique. L'outil informatique qui est ici proposé pourrait ainsi servir d'aide aux travaux effectués par cette commission. Il est évident que l'acceptation ou le rejet des propositions néologiques produites par le système devra dépendre en dernier ressort de la communauté yaqui elle-même. L'étude des justifications de ces décisions pourra même nous donner des informations très précieuses sur l'intuition et sur la connaissance linguistique que la communauté indigène a des mécanismes lexicogénétiques de sa langue, de ses restrictions, préférences et acceptions. D'autre part, la possibilité d'analyse automatique de la ZStation représenterait une aide pour la traduction de mots yaquis, en plus des applications évidentes dans l'enseignement de la lexicologie yaqui. Les applications futures de la ZStation ne se réduisent d'ailleurs pas à la langue yaqui. En effet, les caractéristiques agglutinantes des langues uto-aztèques impliquent une grande importance des mécanismes combinatoires dans la création du lexique de ces langues, ce qui permettrait et justifierait l'ouverture de ce projet de morphologie informatique à d'autres langues de la même famille. Nous avons à ce propos initié l'élaboration de plusieurs dictionnaires d'unités morphologiques du mayo, du guarijío et du tarahumara, trois autres langues uto-aztèques du nord-ouest du Mexique et l'objectif est d'inclure au fur et à mesure d'autres langues de la même famille, y compris des langues mortes comme le opate ou le tehuèque. L'utilisation future de ces bases de données morphosémantiques des formants des langues uto-aztèques pourrait servir aussi bien à des entreprises d'analyse et de comparaison linguistiques, de reconstruction ou à des applications dans le domaine de l'ingénierie linguistique comme celles proposées par la ZStation.

L'étude proposée permet en définitive d'entrevoir quelques uns des bénéfices que l'on pourrait obtenir si la collaboration entre le linguiste indigéniste et l'ingénieur linguiste se développent, collaboration que souhaitait voir grandir M. Zinglé et qui s'annonce à maints égards, fructueuse. Il nous appartiendra d'essayer de poursuivre le chemin signalé par notre regretté professeur.

### Liste des abréviations

- ADJ: adjectif
- ADV: adverbe
- N: nom
- V: verbe

### Abréviations et signes utilisés dans la modélisation informatique

- : réfutation de concept
- + : assertion de concept
- 0 : valeur facultative
- 1 : valeur obligatoire
- adj : détermination catégorielle (adjectif)
- adv : détermination catégorielle (adverbe)
- arg : champ structural (pile argument)
- CAT : champ morpho-syntaxique (pile catégorie)
- char: identificateur de relation (concept: caractéristique)
- CLS : champ de classe (élément formatif)
- conds : sous-champ de conditions
- isa: identificateur de relation (concept: signification)
- lex : élément formatif (lexème)
- nil: sans détermination catégorielle
- SEM : champ sémantique (pile de concepts)
- sub: détermination catégorielle (substantif)
- suff: élément formatif (suffixe)
- verb: détermination catégorielle (verbe)

### Références

ALVAR EZQUERRA, Manuel

1993 *La formación de palabras en español*. Paris: Arcos Libros.

ALVAREZ, Albert

1998a La lexicogénèse de l'espagnol du Mexique. *Travaux du LILLA* 3: 49-69.

1998b Mecanismos de creación neológica del español de México: la neología de forma. *Oserí* 4: 18-24.

2002 Mecanismos de creación neológica del español de México: la neología de sentido. *Oserí* 5: 23-26.

2004 Principales procesos fonológicos de la sufijación en lengua yaqui. Barreras Aguilar I. & Castro Llamas M. (eds), *Memorias del VII Encuentro Internacional de Lingüística en el Noroeste*. Vol. 1: 149-172. Hermosillo: Universidad de Sonora.

ALVAREZ, Albert & ESTRADA, Zarina

2009 Mecanismos de creación de palabras en lengua yaqui. *Lingüística Amerindia. Aportaciones recientes*, Arzápalo M. (ed.), 147-168. México: UNAM-IIA.

BENVENISTE, Emile

1977 Formas nuevas de la composición nominal. *Problemas de Lingüística General II*. México: Siglo XXI, 164-177.

DEDRICK, John & CASAD, Eugene

1999 *Sonora Yaqui Language Structure*. Tucson: University of Arizona Press.

ESTRADA, Zarina & ALVAREZ, Albert

2008 *Parlons Yaqui*. Paris: L'Harmattan.

ESTRADA, Zarina, BUITIMEA, Cresencio, GURROLA, Adriana, CASTILLO, María Elena & CARLÓN, Anabela

2004 *Diccionario Yaqui-Español y Textos. Obra de preservación lingüística*. Hermosillo: Universidad de Sonora/Plaza y Valdés.

GOUY-GUILBERT, Cécile

1983 *Une résistance indienne. Les Yaquis du Sonora*. Lyon: Fédérop.

LEHMANN, Alise & MARTIN-BERTHET, Françoise

1998 *Introduction à la Lexicologie : sémantique et morphologie*. Paris: Nathan Université.

SOWA, John

1984 *Conceptual structures: information processing in man and machine*. Reading, Massachusetts: Addison-Wesley.

SPROAT, Richard

1992 *Morphology and computation*. Cambridge: MIT Press.

ZINGLÉ, Henri

1994 The ZStation workbench and the modelling of linguistic knowledge. *Current issues in mathematical linguistics*, 423-432. Amsterdam: Elsevier-NHLS.

1996 Analyse et génération automatiques de mots dérivés en français. *Travaux du LILLA*, 27-42.

ZINGLÉ, Henri & ALVAREZ, Albert

2003 Morphologie dérivative et traitement automatique de la langue yaqui. *Proceedings of the XVII International Congress of Linguists*, Prague, 24-29 juillet.