Des dictionnaires éditoriaux aux représentations XML standardisées

Mathieu Mangeot, Chantal Enguehard

Chapitre 10

Des dictionnaires éditoriaux aux représentations XML standardisées

Mathieu Mangeot
GETALP-LIG, 41 rue des Mathématiques, BP 53
F-38041 GRENOBLE CEDEX 9, Université de Savoie
Chantal Enguehard
LINA CNRS UMR 6241, Université de Nantes

Introduction

Créer un dictionnaire électronique *ex nihilo* est un travail coûteux car cette tâche mobilise, sur une longue période, le travail de personnes qualifiées, si ce n'est en lexicographie, au moins en linguistique. Lorsque l'environnement socio-économique ne permet pas de rassembler les ressources nécessaires à la confection d'un dictionnaire électronique spécialement dédié au Traitement Automatique des Langues Naturelles (TALN) et que des dictionnaires éditoriaux existent, ces dictionnaires représentent une ressource importante qu'il s'agit d'utiliser pour initialiser la création de ressources lexicales électroniques.

Cet article présente des aspects théoriques et pratiques concernant la conversion de dictionnaires éditoriaux en dictionnaires électroniques. Il prend en compte la question de la limitation des moyens économiques et techniques et de la faible disponibilité des personnes qualifiées. Il est particulièrement destiné aux lexicographes linguistes ou formateurs qui réaliseront la conversion de dictionnaires éditoriaux. C'est pourquoi il détaille des points de méthodologie qui pourront sembler triviaux à des informaticiens spécialistes du traitement automatique des langues mais qui sont pourtant susceptibles d'engendrer des blocages lors d'un processus de conversion menés par des personnes peu familières de la technologie informatique, des expressions régulières ou des formats.

Nos expériences de terrain concernent les langues peu dotées en logiciels et ressources informatiques[1] et qui sont parlées principalement en Asie du sud-est (khmer, malais, vietnamien) et au Sahel (bambara, haoussa, kanouri, tamajaq, zarma), aussi la majeure partie des exemples cités et des situations socio-linguistiques décrites concerne-elle ces zones.

Après un rapide historique consacré aux formats des dictionnaires électroniques, nous présentons deux normes qui leurs sont dédiées. La question des langues peu dotées est exposée et est suivie de quelques exemples de dictionnaires éditoriaux les concernant. Les principales difficultés techniques sont détaillées. Les grandes lignes de la méthodologie de conversion sont énoncées dans la sixième partie et ensuite détaillées : conversion vers un format passerelle à l'aide d'expressions régulières (partie 7) ou d'outils spécialisés (partie 8) ; la partie 9 présente la conversion vers le format cible. La dernière partie est dédiée à la consultation des ressources à travers une plate-forme de gestion de ressources en ligne.

1. Historique des formats électroniques de dictionnaire

Dans cette partie, nous abordons uniquement les formats électroniques de dictionnaire. Pour un historique plus complet, se reporter au chapitre 2.

[1] que nous désignerons simplement par “langues peu dotées”.

1.1. Bandes de photocomposition et première standardisation : SGML

Dès les débuts de l'informatique, il est apparu intéressant d'utiliser un ordinateur pour élaborer des dictionnaires. Les premiers formats électroniques sont d'une part les dictionnaires compilés spécialement pour une application informatique et d'autre part les bandes de photocomposition, représentant un ensemble de commandes envoyées aux imprimantes afin d'imprimer des dictionnaires au format papier. Dans ce chapitre, nous laisserons de côté les dictionnaires compilés pour nous concentrer sur les bandes de photocomposition, représentant les dictionnaires éditoriaux.

À l'époque (au début des années 80), chaque maison d'édition avait mis au point son propre langage de commande pour ses imprimantes. Les bandes n'étaient pas standardisées et ne pouvaient donc pas être échangées entre maisons d'édition ou servir sur des imprimantes de marque différente.

Charles Goldfarb, employé par IBM, a alors mis au point le premier langage de balisage, SGML[2], qui deviendra une norme ISO[3] en 1986. Immédiatement, cette innovation a permis l'échange de documents électroniques de taille importante et peut être considérée comme un premier pas vers la standardisation des formats. Ce langage de balisage, directement inspiré par les bandes d'impression, reflètait principalement des indications de style et de mises en forme plutôt qu'une structuration logique de document. Ainsi, les exemples d'usage exprimés en italique sont-ils marqués à l'aide de l'élément <i>. De même, les articles sont repérés avec un élément de paragraphe <p> Par conséquent la structure des articles est alors majoritairement implicite.

1.2. Simplification du balisage : XML[4]

Il apparaît rapidement un problème avec SGML : la définition de sa structure assez lâche permet d'omettre certaines balises fermantes, ce qui nécessite de décrire précisément la structure de chaque document dans un document annexe appelé Document Type Definition (DTD). Si la DTD n'est pas disponible lors de la lecture, des ambiguïtés d'interprétation peuvent apparaître.

Article au format SGML tiré du Trésor de la Langue Française

```
<a><b>Lexicologie<i>subst. f&ea;m. <br><p>&Ea;tude scientifique du lexique. <i>L'objet de la lexicologie est une th&ea;orie compr&ea;hensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unit&ea;s (mot, idiome)</i> (REY, <i>Le Lexique : images et mod&eg;les</i>, Paris, Colin, 1977, p. 159).</a>
```

Interprétation

Les balises <b> et <i> indiquent que les textes qui les suivent doit être mis, respectivement, en gras ou en italique. Comme la fin de certaines portions de textes concernées par ces balises n'est pas spécifiée, plusieurs interprétations sont possibles, en voici deux :

Lexicologie *subst. fém.*
Étude scientifique du lexique. *L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).* (REY, *Le Lexique : images et modèles*, Paris, Colin, 1977, p. 159).

Lexicologie *subst. fém.*
Étude scientifique du lexique. *L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).* (REY, *Le Lexique : images et modèles*, Paris, Colin, 1977, p. 159).

Figure 1. Exemple d'article au format SGML avec ambiguïté d'interprétation

[2]SGML : Standard Generalized Markup Language.
[3]ISO : International Organization for Standardization.
[4]XML : Extensible Markup Language.

Pour analyser de tels documents SGML, il fallait développer des analyseurs contextuels, capables de tenir compte du contexte dans lequel apparaissent les éléments afin de les interpréter correctement. Le contexte étant décrit à l'aide d'une DTD. Ces analyseurs sont difficiles à programmer. Il est apparu souhaitable de mettre au point un nouveau format de balisage que des analyseurs hors contexte, bien plus simples à programmer, puissent traiter.

C'est pourquoi, en 1997, sous la houlette du W3C[5], s'est formé un groupe de travail ayant comme objectif de définir un nouveau format de balisage. Les travaux de ce groupe ont débouché en 1998 sur le standard XML. XML est un sous-ensemble de SGML qui oblige à associer une balise fermante à chaque balise ouvrante. Il supprime de fait les ambiguïtés d'interprétation et simplifie la programmation des analyseurs. La référence aux DTD pour analyser les documents est alors optionnelle.

En parallèle, les codages de caractères évoluent. Le jeu de caractères codés ASCII[6], premier codage standardisé a été élaboré dans les années 60 par des américains. Il définit seulement 128 caractères, la plupart des diacritiques du français n'y sont pas inclus. SGML utilisant le codage ASCII utilise des entités caractères pour tous les caractères qui ne sont pas inclus dans la table ASCII. Ces entités commencent par le caractère « & », suivi du nom de l'entité et terminé par un autre caractère particulier (habituellement un « ; »). SGML n'a pas standardisé le nom des entités. C'est pourquoi on trouve des noms différents pour les mêmes caractères. Par exemple *&ea.* et *é* représentent un « é ».

La mise au point du langage de balisage HTML, sous-ensemble de SGML utilisé pour le codage des documents Web en 1991 a été l'occasion de standardiser la définition des entités. Le e avec accent aigü sera représenté par l'entité *é*, le a avec accent grave par *à*, etc.

À la même époque est mis au point le standard Unicode ambitionnant de définir une seule table de codage pour tous les alphabets utilisés à travers le monde. La version 2.0, sortie en 1996 à la suite des travaux du comité ISO 10646 (Haralambous, 2004) permet de représenter des caractères sur quatre octets, ce qui dépasse le million de possibilités.

XML a donc choisi un encodage Unicode, l'UTF-8, comme encodage par défaut. Il n'est alors plus nécessaire d'utiliser des entités caractères pour les caractères qui ne sont pas inclus dans la table ASCII. Ils peuvent être notés directement dans le document.

La norme XML est immédiatement utilisée pour représenter des dictionnaires. Tim Bray, co-rédacteur de son cahier des charges, s'est directement inspiré de son travail sur l'informatisation de l'Oxford English Dictionary entre 1987 et 1989.

Article au format SGML	Article au format XML
`<a>**Lexicologie**<i>*subst. f&ea.m.* <p>&Ea.tude scientifique du lexique. <i>*L'objet de la lexicologie est une th&ea.orie compr&ea.hensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unit&ea.s (mot, idiome).*</i> (REY, <i>*Le Lexique : images et mod&eg.les*</i>, Paris, Colin, 1977, p. 159).`	`<a>**Lexicologie**<i>*subst. fém.*</i> Étude scientifique du lexique. <i>*L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).*</i> (REY, <i>*Le Lexique : images et modèles*</i>, Paris, Colin, 1977, p. 159). `

Figure 2. Exemple d'article au format SGML et au format XML[7]

[5]W3C : World Wide Web Consortium.
[6]ASCII : American Standard Code for Information Interchange.
[7]Le texte balisé n'est pas stylé ; les stylages en gras et italique ont été ajoutés pour améliorer la lisibilité du texte.

1.3. Séparation du fond et de la forme : sémantique et feuilles de style

L'arrivée de XML s'accompagne d'un changement de rôle des documents électroniques. Destinés jusqu'ici exclusivement à l'impression, ils vont pouvoir également servir d'autres buts comme l'affichage direct à l'écran. Il est alors possible et souhaitable de séparer la forme (représentation graphique du texte) du fond (texte et structuration des articles).

Les dictionnaires, qui étaient jusqu'alors représentés de manière implicite avec principalement des informations de style (ex : texte en gras pour la vedette), vont être représentés avec une structure explicite associée à des informations sémantiques. Les informations de mise en page et de style seront stockées de manière séparée.

Convertir un dictionnaire éditorial représenté avec des informations de style vers une représentation standardisée nécessite donc de rendre explicite la structure implicite des articles en indiquant la sémantique de chaque information (figure 2). Ainsi, lorsque la structure est explicite, la vedette peut-elle être repérée par l'élément <*mot-vedette*> et, dans les indications de style stockées à part, il est indiqué que les vedettes doivent être affichés en gras.

Article avec structure implicite	Article avec structure explicite
<a> **Lexicologie** subst. *fém.* Étude scientifique du lexique.*L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).* (REY, *Le Lexique : images et modèles*, Paris, Colin, 1977, p. 159). 	<article id="a23301"> <bloc-vedette> <mot-vedette>**Lexicologie**</mot-vedette> <grammaire>subst. *fém.*</grammaire> </bloc-vedette> <bloc-sens> <définition>Étude scientifique du lexique. </définition> <exemple>*L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).*</exemple> <ref-exemple> <auteur>REY</auteur> <œuvre>*Le Lexique : images et modèles*<œuvre> <référence>Paris, Colin, 1977, p. 159).</référence> </ref-exemple> </bloc-sens> </article>

Figure 3. Marquage explicite des informations d'un article

Cette séparation du fond et de la forme apporte de grandes améliorations.

En premier lieu, les informations étant repérées explicitement, il est aisé d'extraire certaines d'entre elles ou de leur appliquer des traitement spécifiques. Ainsi, l'exemple d'usage de l'article balisé explicitement de la figure 3 est le texte encadré par les balises <*exemple*>. Cette extraction est bien plus complexe lorsque l'article est balisé implicitement puisque l'information recherchée est repérée par des balises qui ne lui sont pas spécifiques. Ainsi, dans l'article balisé implicitement de la figure 2, la balise indiquant l'italique étant présente pour plusieurs types d'informations (la catégorie lexicale d'une part et l'exemple d'usage d'autre part) il est difficile pour une machine de distinguer ces deux types d'information.

En second lieu, comme les traitements sont devenus plus simples à réaliser, il devient possible de créer facilement plusieurs traitements. Cette capacité est surtout exploitée en ce qui concerne l'affichage : plusieurs mises en pages et plusieurs styles différents peuvent être associés au même fichier contenant les informations à afficher. Le langage informatique CSS[8] a été développé spécialement pour gérer l'affichage tandis que XSLT[9] vise la fonctionnalité plus large de conversion d'un docu-

[8]CSS : Cascading Style Sheets (feuilles de style en cascade).
[9]XSLT : Extensible Stylesheet Language Transformations. XSLT est un langage de transformation d'arbres XML.

ment XML vers un autre format (y compris XML, ou encore des formats spécifiquement dédiés à l'affichage tels que HTML) à l'aide de transformations structurelles.

L'exemple de la figure 4 est issu d'une feuille CSS. Il permet de spécifier des informations de style destinées à l'affichage. La première règle affiche le mot-vedette en gras, la deuxième affiche la classe grammaticale, les exemples et le nom de l'œuvre en italique.

```
mot-vedette {
  font-weight: bold;
}
grammaire, exemple, œuvre {
  font-style: italic;
}
```

Figure 4. Feuille CSS permettant la mise en page d'un article

L'exemple de la figure 5, issu d'une feuille XSLT, permet d'afficher le numéro des blocs de sens et d'ajouter des parenthèses autour des références bibliographiques d'un l'exemple d'usage.

```
<xsl:template match="bloc-sens-num">
       <br/><xsl:apply-templates select="@num"/>
       <xsl:apply-templates/>
</xsl:template>

<xsl:template match="ref-exemple"><xsl:text> (</xsl:text><xsl:apply-templates/>
<xsl:text>) </xsl:text></xsl:template>
```

Figure 5. Feuille XSLT permettant la mise en page d'un article

2. Normes de représentation de dictionnaires

Dans cette partie, nous discutons des normes ayant pour but de standardiser les structures des articles de dictionnaires.

2.1. La Text Encoding Initiative (TEI)

2.1.1. Présentation

La TEI[10] est pilotée par un consortium regroupant des organismes principalement étatsuniens (ACH[11], ACL[12], ALLC[13]) et européens (ATILF[14], LORIA[15], INIST[16]) financés par des fonds de recherche publics.

L'objectif de la TEI est de définir un format d'échange, de création et de stockage de textes annotés à l'aide d'un jeu d'éléments standardisés. L'organisation est modulaire. Il existe un ensemble commun d'éléments "core tag set" complété par huit ensembles de base pour les différents genres de textes : prose, poésie, drame, parole, dictionnaires, terminologie, base générale et mixé. Des modules additionnels viennent enrichir les descriptions : corpus, alignement, etc.

Le standard TEI est un ensemble de directives (guidelines) pour l'encodage d'un texte. Les débuts des travaux en 1986 sont fondés sur SGML. Plusieurs versions se succèdent : la première version complète est P1 (1990) ; P3 (1994) constitue un standard *de facto* pour les corpus ; P4 (2002) est compatible avec XML ; P5 (2007) introduit la notion de version intermédiaire tous les six mois.

[10] www.tei-c.org
[11] ACH : The Association for Computers and the Humanities.
[12] ACL : The Association for Computational Linguistics.
[13] ALLC : The Association for Literary and Linguistic Computing.
[14] ATILF : Laboratoire Analyse et Traitement Informatique de la Langue Française.
[15] LORIA : Laboratoire lorrain de recherche en informatique et ses applications.
[16] INIST : Institut de l'Information Scientifique et Technique.

Nous nous focalisons ci-dessous sur cette dernière version dont les directives sont structurées en vingt-trois chapitres.

2.1.2. En-tête

Chaque document encodé selon le standard TEI doit comporter un en-tête ayant pour rôles l'identification du document, la caractérisation globale de son contenu, la mémoire électronique de ses variations, et la documentation fine de la stratégie d'encodage qui a été appliquée au texte. L'en-tête est donc nécessaire à la réutilisation et à la maintenance du texte. Le chapitre 2 de la version P5 y est entièrement consacré (Text Encoding and Interchange, 2004).[17]

2.1.3. Dictionnaire

Le dictionnaire peut apparaître comme un genre sur mesure pour être décrit par un langage structuré et pour être consulté en ligne du fait de sa structuration en articles théoriquement construits selon un même modèle. Dans les faits, des régularités structurelles existent mais il y a aussi beaucoup de variations avec des régularités locales comme le fait qu'un composant d'identification des items traités précèdent le(s) composant(s) de traitement. Si nous considérons un dictionnaire donné, la régularité structurelle prévaut le plus souvent, même s'il peut y avoir des séquences de composants d'articles optionnels et répétables, et non ordonnés (parfois codés en tant que contenus mixtes XML). Il s'ensuit que n'importe quel élément peut apparaître presque n'importe où dans un article de dictionnaire.

Le chapitre 9 du standard TEI est centré sur les dictionnaires. Il est composé de constituants génériques (<*front*> = texte préliminaire, <*body*> = corps du document, <*back*> = texte postliminaire <*div*>, <*div0*>, <*div1*> = divisions du texte, <*entry*> = entrée structurée, <*entryFree*> = entrée libre, <*superEntry*> = super-entrée), de données structurantes (<*form*> = informations sur une forme, <*hom*> = homographe, <*sense*> = sens de mot, etc.) et de données informatives (<*def*> = définition, <*pos*> = catégorie grammaticale, <*usg*> = usage).

La figure 6 montre un exemple de structure d'article décrite à l'aide d'éléments de la TEI. Les pointillés indiquent des éléments optionnels comme <number>. En bas de certains éléments, sont indiquées les cardinalités : l'élément <cit> doit être présent au moins une fois et peut être répété à l'infini.

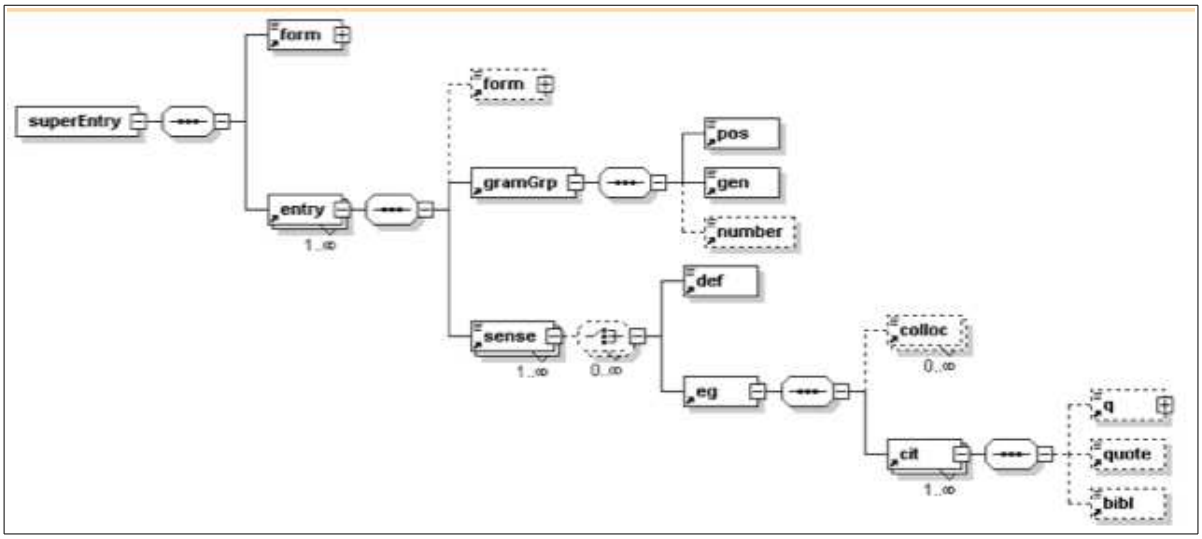

Figure 6. *Article décrit à l'aide d'éléments de la TEI*

La figure 7 reprend l'exemple d'entrée précédent et l'encode selon la TEI.

[17] Voir par exemple l'en-tête du lexique Morphalou de formes fléchies du français : http://www.cnrtl.fr/lexiques/morphalou/

```
<entry>
  <form>
    <orth>lexicologie</orth>
  </form>
  <gramGrp>
    <pos>subst.</pos>
    <gen>fém.</gen>
  </gramGrp>
  <sense n="1">
    <def>Étude scientifique du lexique.</def>
    <eg>
      <cit type="example">
        <quote>L'objet de la lexicologie est une théorie compréhensive du fait
          lexical, tant au niveau des structures (lexique, vocabulaires) que des
          unités (mot, idiome).</quote>
      </cit>
      <bibl>
        <author>REY</author>, <title>Le Lexique : images et modèles</title>.
        <pubPlace>Paris</pubPlace>, <publisher>Colin</publisher>,
        <date>1977</date>, <biblScope type="pp">p.159</biblScope>.
      </bibl>
    </eg>
  </sense>
</entry>
```

Figure 7. Article *lexicologie* encodé selon la TEI

Pour faire face au problème de structuration des articles, la TEI propose une solution binaire : un article peut être représenté par un élément <*entry*> dont la structure est rigide et très codifiée ; la représentation par un élément <*entryFree*> peut lui être préférée car elle admet d'insérer dans l'article n'importe quels éléments et dans n'importe quel ordre.

Dans la pratique, l'élément <*entry*> se révèle trop contraignant à utiliser. Par conséquent, les lexicographes préfèrent utiliser l'élément <*entryFree*>, mais celui-ci est trop lâche pour permettre une réelle standardisation et des échanges de données encodées avec la TEI.

2.1.4. Utilisation de la TEI

La TEI a rencontré un réel succès pour le codage des corpus. Il n'en est malheureusement pas de même pour les dictionnaires. La solution proposée (dichotomie entre les éléments <*entry*> et <*entryFree*>) n'est pas satisfaisante. De plus, les données des corpus ne sont pas la propriété des éditeurs au même titre que les articles de dictionnaires que leurs lexicographes ont rédigés. Les collaborations et adoptions de standards peuvent être plus aisées les concernant (même si les éditeurs britanniques se sont montrés plus collaboratifs en la matière que les éditeurs français). Il s'ensuit que dans la pratique, très peu de dictionnaires sont encodés avec la TEI. Il s'agit plutôt de lexiques non commerciaux (CJKV English dictionary, dictionary of buddhism[18]).

Un autre facteur limitant est le manque d'outils de description de macrostructures[19] pour les bases lexicales plurilingues. De telles bases comprennent plusieurs volumes logiques, chaque volume regroupant des articles d'une même langue. Il est alors nécessaire de définir précisément les volumes et les liens entre ceux-ci qui peuvent être parfois très complexes, par exemple dans le cas de structures pivot à plusieurs étages, de réseaux lexicaux, etc.).

2.2. Lexical Markup Framework

Dans le domaine du TALN, les dictionnaires électroniques sont vus comme des ressources au même titre que les corpus. Pour les désigner, nous utiliserons par la suite, le terme de *ressource lexicale* qui regroupe les lexiques, les dictionnaires et les bases lexicales. La nécessité de partager ces res-

[18] http://www.buddhism-dict.net/

[19] Nous définissons la macrostructure comme l'organisation des volumes d'une base lexicale et liens inter-volumes.

sources a abouti en 2003 au projet de standardisation Lexical Markup Framework (Romary, Salmon-Alt & Francopoulo, 2004) qui consiste en un méta-modèle séparant les parties lexicale, grammaticale et sémantique (voir figure 8). La classe principale est une ressource lexicale *Lexical Resource*. Elle contient une classe décrivant la méta-information sur le lexique *Global Information* et un ou plusieurs lexiques *Lexicon*. Le lexique contient une ou plusieurs entrées lexicales *Lexical Entry*. Une entrée lexicale contient une ou plusieurs formes de l'entrée *Form* ainsi qu'un ou plusieurs sens *Sense*. Les formes contiennent des variantes orthographiques *Form representation*. Les sens peuvent à leur tour contenir des sens de manière récursive. Les sens peuvent contenir des définitions *Definition* qui contiennent des contenus textuels *Text Representation* et des descriptions narratives *Statement*. La classe *representation* permet de faire un lien entre les variantes orthographiques *Form Representation* et leurs occurrences dans un texte *Text Representation*. Ce méta-modèle est devenu une norme ISO sous le numéro 24613:2008 en novembre 2008 (Francopoulo, Bel, George, Calzolari, Monachini & Soria, 2009).

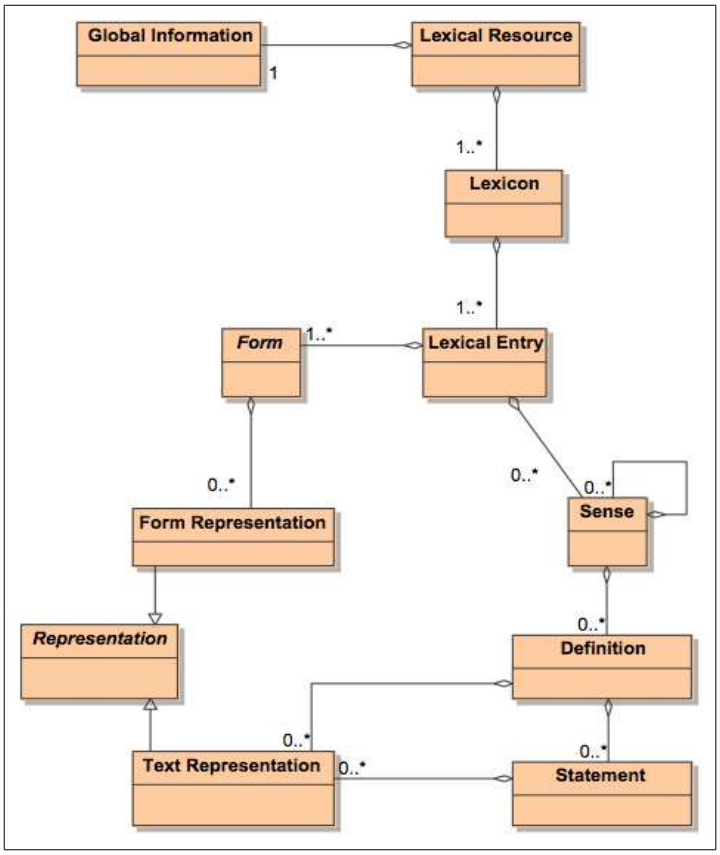

Figure 8. Méta-modèle de base de LMF

Ce méta-modèle, véritable partie centrale de la norme, peut être enrichi par des extensions qui doivent être des sous-classes des classes existantes dans le modèle de base. Il existe des extensions pour la morphologie, la syntaxe, la sémantique, etc. De nouvelles extensions pourront être développées par la suite pour répondre à des besoins spécifiques.

Les données peuvent être de deux types : soit une chaîne de caractères Unicode, soit une valeur choisie dans une liste de catégories provenant de la spécification MARTIF, (norme ISO 12620:2009).

La partie normative s'arrête à ce stade, en particulier elle ne définit aucun nom d'élément à utiliser. Il existe bien sûr des exemples de documents XML dans la partie informative de la norme mais il n'est en aucun cas obligatoire d'utiliser ces noms. Selon nous, c'est précisément l'intérêt de la norme LMF de permettre la conception d'un dictionnaire respectant la norme LMF tout en utilisant ses propres éléments. Il se peut même que, tout comme Monsieur Jourdain faisait de la prose sans le sa-

voir, un certain nombre de ressources lexicales respectent déjà le format LMF même si elles ont été conçues avant l'établissement de cette norme.

Afin de faciliter l'échange de données, il conviendra toutefois de proposer systématiquement, en accompagnement d'une ressource, un programme permettant de convertir la ressource avec ses éléments propres vers les éléments utilisés dans la partie informative de LMF et vice-versa. Un programme XSLT est par exemple tout à fait approprié pour ce genre de manipulation.

En conclusion, nous conseillons au lecteur d'utiliser la TEI uniquement pour l'en-tête de son dictionnaire car celui-ci est très détaillé et son usage est plus répandu que le chapitre concernant les dictionnaires, notamment pour les corpus. Pour le contenu, il est préférable d'utiliser une structure respectant le standard LMF. L'en-tête de la TEI correspondra alors au contenu de la classe *Global Information* de la norme LMF..

3. Des ressources lexicales pour les langues peu dotées

3.1. Problématique des langues peu dotées

Quoiqu'un inventaire précis soit difficile à réaliser, il existerait actuellement environ 6 000 langues parlées par les êtres humains. Parmi celles-ci 200 à 300 seulement sont écrites. Le passage de l'oral à l'écrit est complexe et ne peut se limiter à une simple transcription des sons entendus. Il est nécessaire de mener des études linguistiques afin de réaliser une description de la langue. Il s'agit de régler de multiples questions : déterminer le système de transcription qui sera utilisé[20] et, à l'intérieur de celui-ci, choisir les signes les plus adéquats, puis les règles orthographiques, syntaxiques, etc. Enfin, les langues sont plus ou moins bien dotées en ce qui concerne leur support par des outils informatiques : clavier adapté, correcteur orthographique, synthèse de la parole, traduction automatique, etc. Une classification fondée sur l'estimation de l'équipement d'une langue en outils et ressources informatiques détermine trois classes : les langues très bien dotées ou langues-τ (par exemple, l'anglais ou le français), les langues moyennement dotées ou langues-μ (par exemple le portugais ou le suédois), et les langues peu dotées ou langues-π (par exemple le bambara ou le kanouri) (Berment, 2004).

L'appellation langues peu dotées recouvre des situations contrastées. Nous citons ici trois conjonctures :

- il s'agit de la langue officielle d'un État, comme l'est l'irlandais (ou gaélique d'Irlande) en Irlande.

- il s'agit d'une langue sans statut officiel, devenue langue régionale : en France, par exemple, le basque ou le breton ; le ladin en Italie, le cornish (langue de Cornouailles) au Royaume-Uni.

- il s'agit d'une langue nationale d'un État dont la langue officielle (celle des actes officiels, de l'enseignement, des lois) est différente et souvent issue d'un autre État anciennement colonisateur (Calvet, 1996). C'est le cas par exemple, du kanouri au Niger, où il y a dix autres langues nationales et où la langue officielle est le français.

Nous focalisons notre attention sur les langues peu dotées dont le contexte socio-économique est caractérisé par des ressources réduites : d'une part il y a peu de de linguistes ayant comme langue maternelle une langue peu dotée et exerçant leur activité professionnelle sur cette langue, d'autre part le budget consacré au développement de ressources linguistiques est faible. C'est le cas des pays du Sahel où le budget de l'État est insuffisant, notamment en ce qui concerne l'éducation, et où le taux d'analphabétisme est très élevé. Les investissements de l'État consacrés à la planification linguistique et, en particulier, au développement de ressources linguistiques électroniques, sont par conséquent très limités. Les quelques travaux qui sont menés sont caractérisés par leur discontinuité

[20]La situation politique ou religieuse influence parfois le choix du système de transcription, par exemple quand il s'agit de trancher entre le système alphabétique occidental ou arabe, ou un alphabet local (Calvet 1987).

dans le temps et leur dissémination spatiale qui nuisent à leur pérennisation (Streiter, Scannell & Stuflesser, 2006).

Développer *ex nihilo* des ressources lexicales nécessite des budgets importants, des personnes qualifiées et disponibles, et la capacité à mener un projet durant plusieurs années, conditions qui ne peuvent être réunies dans de nombreux pays. Cependant, il existe parfois des dictionnaires éditoriaux (souvent bilingues) qui peuvent être exploités[21] pour réaliser, en quelques semaines et à faible coût, une première version d'une ressource électronique. La collecte des fichiers informatiques les constituant constitue un préalable important. Toutefois, en leur absence, le dictionnaire peut être saisi à nouveau quand seul un exemplaire imprimé a pu être collecté.

Quel que soit son format (électronique ou imprimé) un dictionnaire représente une somme de connaissances importantes et qui peut être récupérée. Dans tous les cas les auteurs ou l'éditeur du dictionnaire initial doivent être associés au projet afin d'obtenir leur accord pour que la ressource lexicale qui sera produite puisse être largement diffusé sous forme électronique et visible sur internet.

La méthodologie de conversion que nous présentons fait intervenir des linguistes spécialistes des langues concernées et des informaticiens connaissant le traitement automatique des langues. Elle prend en compte la limitation des ressources économiques en limitant le temps de travail des personnes et en privilégiant l'utilisation de logiciels gratuits.

3.2. Dictionnaires rédigés par un seul auteur

Certains dictionnaires sont l'œuvre d'un seul auteur. Celui-ci y a en général consacré part importante de sa vie.

Nombre de ces dictionnaires sont bilingues car leur auteur vise à faire connaître une langue qu'il a appris à maîtriser alors qu'il a pour langue maternelle une autre langue. C'est le cas par exemple du dictionnaire français-khmer de Denis Richer (Association Pays perdu). Beaucoup ont été rédigés par des religieux qui à l'origine s'étaient installés pour évangéliser les peuples de territoires colonisés (pères blancs en Afrique, jésuites portugais en Asie). Nous avons par exemple travaillé sur le dictionnaire bambara-français du Père Charles Bailleul (Bailleul, 1996).

Il existe aussi des dictionnaires élaborés par des personnes lettrées, souvent linguistes, désireuses de mettre leur savoir au service de leur langue maternelle comme le dictionnaire élémentaire hausa-français d'Abdou Minjinguini (2003) ou encore le dictionnaire zarma de Issoufi Alzouma Umarou (1997) (qui, fait rare, est monolingue).

Les plus récents de ces dictionnaires sont généralement rédigés entièrement avec un logiciel de traitement de texte (Word, WordPerfect). Leur structure évolue au fil de leur élaboration, sans nécessairement respecter une standardisation. En ce qui concerne les contenus, des listes de valeurs *a priori* fermées comme les catégories grammaticales peuvent fluctuer ; les abréviations connaissent également des variations, certaines peuvent même être absentes de la liste des abréviations située en début d'ouvrage. Ces dictionnaires connaissent également beaucoup d'instabilité dans la structuration des entrées surtout si celles-ci sont complexes.

3.3. Dictionnaires construits dans le cadre de projets

Les dictionnaires construits dans le cadre de projets rassemblant un groupe de plusieurs personnes sont généralement le fruit d'un travail de réflexion en amont sur la structure utilisée et la définition des listes fermées de valeurs comme les classes grammaticales.

[21] Dans la mesure où les auteurs ont donné leur accord.

Les dictionnaires de ce type les plus récents sont construits avec des outils de lexicographie comme Linguist Shoebox/Toolbox[22] et FieldWorks Language Explorer (FLEx)[23] de la Société Internationale de Linguistique (qui construit des dictionnaires là aussi dans un but évangélique) ou TshwaneLex (TLex)[24].

Bien souvent, même si ces outils sont capables d'exporter leurs contenus vers une structure XML, cette opération n'a pas été effectuée et les fichiers produits par les outils lexicographiques ne sont pas disponibles. Seuls sont utilisables les fichiers Word destinés à l'impression.

Nous avons par exemple travaillé sur :

- les dictionnaires en langues nationales du Niger du projet DiLAF de conversion de dictionnaires éditoriaux bilingues langue africaine-français (Enguehard, Kane, Mangeot, Modi & Sanogo, 2012) ;

- Le dictionnaire FeM français-anglais-malais (Gut et al., 1996) et ses dérivés (FeV, pour le vietnamien, FeT pour le thaï).

4. Difficultés techniques

4.1. Des langues écrites mais peu standardisées

Bien que le caractère peu doté d'une langue ait été défini uniquement en référence à son équipement en outils et ressources informatiques, il s'inscrit souvent dans un contexte de rareté des connaissances linguistiques : les études sur la langue sont peu nombreuses, elles sont peu accessibles car elles ne sont pas publiées dans des revues ou des actes de conférence, et elles ne sont pas disponibles sur internet. De plus, ces langues sont également peu présentes à l'école, que ce soit comme matière d'étude ou comme langue d'enseignement. Quelques exceptions méritent cependant d'être citées :

Au Niger, des écoles expérimentales ont été créées dans les années 1980. L'enseignement y est entièrement dispensé dans une langue nationale pendant la première moitié du cycle primaire puis, pendant la seconde moitié de ce cycle, le français fait son apparition et la langue nationale est étudiée en tant que matière. La dernière année, l'enseignement de déroule uniquement en français. Il en sera de même pendant le reste de la scolarité : au collège, au lycée et dans l'enseignement supérieur (Programme Décennal du Développement de l'Éducation du Niger, 2003).

En Équateur, le peuple shuar (que nous appelons improprement jivaro) s'est structuré en une Fédération des Centres Shuars en 1964. Dans les années 1970 cette fédération a organisé, entre autres, la fondation d'écoles primaires dans les villages avec un soutien par des programmes radiophoniques. Ces écoles sont bilingues : l'enseignement y est dispensé quasiment intégralement en shuar les deux premières années pour évoluer vers un enseignement à parité en shuar et en espagnol en dernière année (Calvet, 1987).

Au-delà de la réussite de ces approches quant à l'alphabétisation des enfants, la création de cursus scolaires favorise la rédaction et l'édition de quelques manuels pédagogiques qui constituent des corpus de textes écrits par des spécialistes des langues, parfois linguistes, ayant une bonne connaissance de la langue de rédaction[25]. De tels corpus constituent des ressources électroniques susceptibles d'être exploitées pour constituer des ressources lexicales. Leur taille reste cependant réduite.

D'autres textes en langues nationales voient le jour. Ils sont écrits par des journalistes, des auteurs de contes, de romans, etc., n'ayant la plupart du temps aucun accès à des ressources linguistiques. Par conséquent, ces textes sont écrits dans une langue peu standardisée présentant en particulier de

[22]www.sil.org/computing/toolbox/
[23]fieldworks.sil.org/flex/
[24]tshwanedje.com/tshwanelex/
[25]Au Niger, par exemple, les manuels (en cinq langues nationales) sont rédigés et édités par l'Institut National de Documentation, de Recherche et d'Animation Pédagogiques (INDRAP).

nombreuses variations orthographiques. Ces corpus ne peuvent donc pas être utilisés comme source de données pour construire automatiquement des ressources lexicales.

Dans ce contexte de dénuement, la récupération d'un dictionnaire éditorial constitue une première étape susceptible d'accélérer la constitution de ressources utilisables pour des applications de traitement automatique des langues naturelles. Dans certains dictionnaires, l'orthographe des mots est standardisée et est conforme aux études linguistiques ; dans d'autres, comme (Bailleul, 1996), les variantes sont explicitement signalées et situées géographiquement tandis que l'orthographe officielle est signalée en sus des graphies usuelles. De plus, les définitions, et les exemples d'usage constituent un corpus de phrases exploitable et de nombreuses entrées sont accompagnées d'informations morphologiques permettant de calculer les différentes formes d'une même entrée.

4.2. Des caractères spéciaux

Le développement d'outils de traitements automatiques d'une langue sous sa forme écrite nécessite (même si ce n'est pas suffisant) que tous les caractères soient codés de manière identique dans tous les textes, ce codage devant être partagé par tous les ordinateurs. Les codages élaborés depuis les années 1960 pour la table ASCII et les années 1970 pour les tables adaptées aux langues non anglophones ont constitué un compromis entre cet objectif et la taille mémoire réduite des ordinateurs de cette époque. Au tournant des années 1990, les progrès technologiques quant au stockage d'informations de plus en plus volumineuses ont permis d'envisager d'associer un code unique à chacun des caractères de toutes les écritures de langues : c'est ce que vise le standard Unicode (Haralambous, 2004).

Définis dans les années 1960, donc bien avant Unicode, les alphabets de la plupart des langues nationales africaines utilisent des caractères spéciaux qui étaient absents des tables de caractères de l'époque. Par exemple, le caractère b crossé (ɓ, Ɓ) apparaît dans l'alphabet haoussa tandis que le caractère n vélaire voisé (ŋ, Ŋ) figure dans les alphabets bambara, tamajaq et soŋay-zarma.

Constituées avant tout dans un but d'édition et donc d'impression sur support papier, des polices permettant d'afficher ces caractères spéciaux ont alors été créées en redessinant le glyphe de certains caractères (Chanard & Popescu-Belis, 2001). Ces polices ont permis pendant des décennies l'édition de textes en langues nationales mais interdisent tout traitement automatique des langues portant sur les textes édités (Enguehard, 2009). L'habitude de les utiliser s'étant installée, les fichiers sources des dictionnaires au format initial utilisent de telles polices. Il est donc nécessaire de les convertir vers Unicode afin que le codage des caractères respecte les standards internationaux.

Identifier les caractères à convertir est une tâche difficile car, d'une part, les dictionnaires, de taille importante (plusieurs milliers d'entrées), ne peuvent donner lieu à une relecture exhaustive et, d'autre part, il est parfois fait usage de plusieurs polices de caractères au sein du même document.

Établir les caractères de remplacement peut s'avérer délicat si les polices originales ne sont pas disponibles, ce qui est la situation la plus courante. Dans ce cas, il est préférable de disposer d'une version imprimée afin d'être certain des conversions à établir. Cette étape peut devenir plus subtile quand un même caractère est utilisé pour afficher des glyphes différents. Par exemple l'esperluette & est redessinée comme le t avec point suscrit ṭ de l'alphabet tamajaq dans la police 'Albasa Tamjq', comme le d crossé ɗ de l'alphabet haoussa dans les polices 'AlbasaRockwellhau' et 'Hausa' et comme le e ouvert ɛ de l'alphabet bambara dans les polices 'Times New bambara' et 'Arial Bambara'. Il peut arriver aussi que des polices différentes portent le même noms. Enfin, un même caractère peut être utilisé, au sein du même document, pour afficher des glyphes différents. Par exemple dans le dictionnaire bilingue tamajaq-français (Programme de soutien à l'éducation de base, 2007), le caractère p minuscule (U+0070) a été utilisé comme tel dans les parties des entrées rédigées en français, et redessiné comme un schwa ə dans la police 'Tamajaq Literacy2 TT20.4 SIL-Sop' pour les parties en tamajaq .

Les caractères avec signe(s) diacritique(s), souvent utilisés pour noter la tonalisation, doivent parfois être convertis car leurs différents codages ne sont pas identifiés comme identiques par tous les logiciels. Par exemple, la lettre u minuscule avec accent aigu existe en deux codages, "U+0075 U+0301" et "U+00FA"[26]. Comme nous souhaitons que les ressources électroniques produites soient utilisables par des outils de TALN que nous ne connaissons pas *a priori*, il est nécessaire d'uniformiser les codages.

4.3. Caractères, alphabets et Unicode

Le standard Unicode vise à représenter tous les caractères de toutes les écritures des langues (Desgraupes, 2005). Bien qu'il évolue à chaque nouvelle version (la version 6.1 code 110 116 caractères) (Unicode, 2012), certains alphabets restent encore incomplets, en particulier ceux des langues peu dotées, car introduire de nouveaux caractères nécessite des moyens humains et financiers qui leur font défaut. Voici quelques exemples de lacunes que nous avons relevés.

4.3.1. Digraphes

Les digraphes notent un seul son mais sont graphiquement composés de deux caractères. Leur usage modifie l'ordre du tri lexicographique. Ainsi, en haoussa et en kanouri, le digraphe sh est situé après la lettre s[27]. Donc le verbe *sha* « boire » est situé après le mot *suya* « frite » dans un dictionnaire haoussa, et le nom *shadda* « basin » est situé après le verbe *suwuttu* « dénouer » dans un dictionnaire kanouri.

Ces subtilités peuvent être difficile à traiter au niveau logiciel. Deux solutions peuvent être envisagées :

1 – Les digraphes manquants peuvent être introduits en tant que signe dans le répertoire Unicode. Certains, utilisés par d'autres langues, y figurent déjà, parfois avec leur différentes casses. Par exemple, DZ (U+01F1), Dz (U+01F2), dz (U+01F3) sont utilisés en slovaque ; NJ (U+01CA), Nj (U+01CB), nj (U+01CC) en croate et pour transcrire la lettre Њ de l'alphabet cyrillique en serbe ; etc.

2 – Les alphabets sont modifiés afin de ne plus faire apparaître les digraphes, ce qui a pour conséquence de modifier l'ordre lexicographique . C'est le cas, par exemple, de l'alphabet français dans lequel le son /ʃ/ est noté par la suite des deux caractères c et h ; le même son est noté par la suite des deux caractères s et h en anglais. Cet aménagement présente l'inconvénient de rompre avec le principe de représenter chaque son par un caractère[28]. De plus, des difficultés politiques peuvent surgir puisqu'il s'agit de modifier la définition d'alphabets de langues nationales qui ont été fixés par décrets.

4.3.2. Caractères avec signes diacritiques

Certains des caractères portant des signes diacritiques figurent dans Unicode comme un unique signe, d'autres ne peuvent être obtenus que par composition. Ainsi, la lettre j avec caron de l'alpha-

[26]En ce qui concerne les éditeurs de texte que nous avons utilisés, les fonctions 'Recherche' d'Open Office et Notepad++ considèrent les caractères correspondants aux deux codages énoncés comme différents alors que celle de gedit les assimile à un même caractère.

[27] Ordre alphabétique de l'alphabet haoussa du Niger : a b ɓ c d ɗ e f fy g gw gy h i j k kw ky ƙ ƙw ƙy l m n o p r s sh t ts u w y y' z (République du Niger, 1999a).
Ordre lexicographique de l'alphabet kanouri du Niger : a b c d e ə f g h i j k l m n ny o p r ɾ s sh t u w y z (République du Niger, 1999b).
Neuf digraphes absents d'Unicode avec leurs trois casses apparaissent dans ces alphabets : fy, gw, gy, ky, kw, ƙy, ƙw, sh, ts.

[28]Ce principe a été adopté lors de réunions ayant pour but de fixer des alphabets communs à plusieurs langues transfrontalières d'Afrique. La première, en septembre 1978, organisée par l'UNESCO au CELTHO (Centre d'études linguistiques et historiques par tradition orale) à Niamey, crée l'« Alphabet africain de référence » fondé sur les conventions de l'IPA (International Phonetic Association) et de l'IAI (International African Institute).

bet tamajaq existe dans Unicode en tant que signe ǰ (U+1F0), mais sa forme majuscule doit être composée avec la lettre J et le signe caron (U+30C).

5. Processus de conversion des ressources lexicales dans un format standardisé

5.1. Différents niveaux de ressources

Les ressources lexicales peuvent être classées selon trois niveaux (Mangeot, 2001) :

— Le premier niveau (format original) est constitué de ressources exprimées dans un format original, généralement peu standardisé provenant par exemple d'un logiciel de traitement de texte au format .doc, ou de n'importe quel autre format différent de XML. Bien souvent, la structure des articles est implicite et les informations de mise en forme ne sont pas séparées de celles de la structuration. Toutefois, lorsque le document texte est issu d'un logiciel de lexicographie, comme Shoebox (Buseman, Buseman, Jordan & Coward, 2000) par exemple, les noms de style reprennent souvent les noms des champs qui avaient été définis dans l'outil (Lexeme, Section1, Part_of_speech, etc.). Il est alors relativement facile de reconstruire la structure d'origine.

— Le deuxième niveau comprend les mêmes ressources converties en un format XML conservant la structure originale (format passerelle). Les informations sont cette fois toutes marquées explicitement et la structure des articles est apparente.

— Le troisième niveau (format cible) est celui d'une nouvelle ressource dont la structure est définie en fonction des besoins et des objectifs du projet. Bien souvent, le choix du format cible est guidé par l'objectif de produire des ressources lexicales réutilisables pour des applications de TALN.

La conversion de chaque dictionnaire éditorial produira donc deux nouvelles ressources distinctes : une au format passerelle et une au format cible (voir Figure 9).

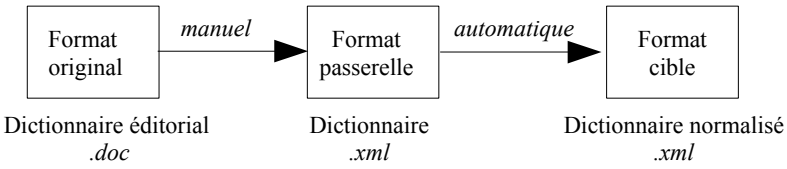

Figure 9. Processus de conversion

La ressource produite au format passerelle est la conversion la plus fidèle possible de la version initiale du dictionnaire dans un format XML. Il est préférable que cette conversion soit réalisée par des linguistes afin de corriger des éventuelles erreurs dans les données. Il s'agit de garder la structure originale des articles en explicitant les parties implicites. Lors de cette étape, il faut garder à l'esprit le format cible pour essayer de s'en approcher lorsque des choix doivent être effectués mais l'objectif n'est pas de respecter des standards à la lettre.

Concernant la microstructure, si, par exemple, dans le dictionnaire original, le choix a été fait de ne faire qu'un seul article pour toutes les classes grammaticales d'un mot, cette structuration n'est pas modifiée, même si elle ne correspond pas au standard du format cible.

Concernant la macrostructure, celle-ci ne devra pas être modifiée. Si le dictionnaire original est constitué d'un seul volume, le dictionnaire au format cible contiendra également un seul volume.

La ressource au format passerelle reste lisible et facile à appréhender par les linguistes chargés de corriger et d'enrichir le dictionnaire (par exemple, traduire les exemples dans une autre langue). Le format étant XML, il est également possible de la visionner directement dans un navigateur Web en y associant une feuille de style ou de l'importer sur une plate-forme de manipulation de ressources

lexicales telle que Jibiki (Mangeot, 2003) pour consultation et édition en ligne. Cette ressource servira de référence dans le futur si de nouvelles versions sont produites.

La ressource au format cible est produite à partir de celle au format passerelle. Les choix gouvernants la structuration peuvent s'éloigner du format initial pour aller vers un format cible défini par les acteurs du projet.

Pour le format cible, nous conseillons de se conformer au standard LMF afin de faciliter la réutilisation des données dans des projets de traitement automatique des langues. Cependant, ce format devra être, si nécessaire, adapté afin de prendre en compte la spécificité des langues.

Concernant la macrostructure du format cible, il est alors possible de modifier celle du format original pour en définir une nouvelle adaptée aux objectifs du projet. Par exemple, dans le cas d'un dictionnaire bilingue langue A → langue B ne contenant qu'un seul volume dans le format original, la macrostructure du format cible pourra être composée d'un volume langue A → langue B et du volume inverse langue B → langue A ou même d'une macrostructure pivot : un volume langue A, un volume pivot contenant les liens entre les deux langues et un volume langue B (Mangeot et al., 2003). Le volume inverse ainsi constitué sert dans un premier temps de « squelette » et doit être enrichi par la suite afin de constituer une ressource de qualité.

La ressource au format passerelle peut être ensuite convertie automatiquement dans le format cible en utilisant par exemple un programme XSLT car il s'agit dans les deux cas de fichiers XML.

5.2. Établissement des jeux d'éléments du format passerelle

Pour repérer les types d'information des articles, il faut choisir un jeu d'éléments. Se pose alors la question du choix de la langue utilisée pour les éléments. Le choix de l'anglais, langue internationale de la recherche, peut être privilégié. Mais, dans bien des cas, d'une part l'anglais n'est pas une langue présente dans les dictionnaires manipulés, et d'autre part, elle n'est pas maîtrisée par tous les linguistes travaillant sur le projet. Dans le cas de projets d'informatisation de langues peu dotées, il est important d'inciter les partenaires à utiliser les termes de leur langue pour définir le nom des éléments en langue source (ou langues nationales). Cette démarche peut éventuellement donner lieu à la création de nouveaux termes qui n'existaient pas dans ces langues et ainsi contribuer au transfert de connaissances et d'idées et, par conséquent, au développement scientifique et technologique (Diki-Kidiri, 2004). Dans le cas des langues peu dotées, d'un point de vue politique, il s'agit de s'éloigner d'une vision post-coloniale des statuts sociaux de ces langues et de contribuer à leur valorisation. Les partenaires linguistes peuvent donc définir des jeux d'éléments XML avec des noms qui, dans leur majorité, sont exprimés dans leur propre langue.

À titre d'exemple, voici les noms d'éléments choisies pour le dictionnaire kanouri-français (Programme de soutien à l'éducation de base, 2004) dans le projet DiLAF :

Nom d'élément en kanouri	**Équivalent français**
kalma	mot
bowodu	prononciation
naptu_curo_nahauyen	catégorie grammaticale
maana	signification
misal	exemple
kalakta	traduction en langue
maana_tiloa	synonyme
fərəm	antonyme
bowodu_gade	variante
mane	voir

Figure 10. *Jeu d'éléments choisis pour le dictionnaire kanouri-français du projet DiLAF*

6. Conversion vers le format passerelle à l'aide d'expressions régulières

L'ensemble des opérations décrites dans cette étape a été conçu afin de pouvoir être principalement mis en œuvre par des linguistes ou des lexicographes avec l'assistance d'informaticiens spécialistes en traitement automatique des langues.

6.1. Conversion du format texte vers XML

La figure 11 montre un extrait du dictionnaire kanouri-français (Programme de soutien à l'éducation de base, 2004) au format original .odt (ouvert avec OpenOffice) après conversion des caractères spéciaux . Les exemples suivants seront également extraits de ce dictionnaire.

abərwa[1]	[àbərwà] cu. Kəska təngəri, kalu ngəwua dawulan tada cakkidə. *Kəryende kannua nangaro, abərwa cakkiwawo.* Fa.: ananas
abərwa[2]	[àbərwà] cu. Tada abərwaye. *Abərwa bafiya, jauro lelea.*. Fa.: fruit d'ananas

Figure 11. *Extrait du dictionnaire kanouri-français au format original*

Le format Open Document Format (ODF) ~~est un standard OASIS. La version 1.0 est également un standard ISO 26300:2006. La version 1.1 a été approuvée par OASIS le 2 février 2007. La version 1.2 est en cours de rédaction. Ce format est utilisé de manière native par les suites bureautiques de la famille de OpenOffice (StarOffice, NeoOffice, LibreOffice).~~ Il a le précieux avantage d'être fondé sur un format XML. Le titre de cette partie est donc un peu trompeur. Au lieu d'une conversion d'un format vers un autre, il s'agit de récupérer le contenu XML du document puis de le transformer pour obtenir un document XML dans le format souhaité.

Un document .odt au format ODF est en fait une archive zippée de plusieurs fichiers, dont le contenu textuel est balisé en XML. Ce contenu est stocké dans le fichier content.xml situé à la racine de cette archive. Pour récupérer ce fichier, il suffit de suivre quelques manipulations astucieuses. Sur MacOs, il est nécessaire de créer un dossier vide puis d'y copier le fichier .odt. Ensuite, il faut ouvrir un terminal puis exécuter la commande unzip sur le fichier .odt. Sur Windows, il faut changer l'extension du fichier .odt en .zip puis ouvrir l'archive .zip.

Le fichier content.xml peut alors être extrait de l'archive et renommé puis placé dans un autre répertoire. Il devient le fichier de base sur lequel se poursuivent les traitements.

L'étape suivante consiste à éditer ce fichier avec un éditeur de texte "brut". Cet éditeur devra *a minima* comporter des fonctionnalités de recherche et de remplacement supportant un langage d'expressions régulières et une coloration de syntaxe (pour faciliter le travail). Les outils gratuits en source ouverte comme Notepad++ font très bien l'affaire.

Le fichier étant constitué d'une seule ligne, la première manipulation est l'ajout des sauts de ligne. Si le document initial est bien rédigé, un article de dictionnaire correspond, dans la grande majorité des cas, à un paragraphe. Insérer un saut de ligne devant ou à la fin de chaque paragraphe permet donc de visualiser chaque article sur une seule ligne. Ce traitement est réalisé par l'expression régulière suivante[29].

```
s/<text:p/\r<text:p/g
```

L'en-tête contenant les informations spécifiques à OpenOffice peut maintenant être supprimé.

[29]Dans cet article, nous utilisons la syntaxe d'expressions régulières du langage Perl.

6.2. Marquage explicite des informations

Cette étape consiste à marquer explicitement toutes les parties d'information constituant les articles.

Dans le fichier d'origine, chaque type d'information (par exemple, la définition, les exemples d'usage, etc.) se distingue souvent des autres par l'usage d'un style différent. Si le fichier d'origine a été bien conçu, il est donc possible d'identifier un style correspondant à l'information. La figure 12 montre une partie de l'article *abaɾwa* « ananas » du dictionnaire kanouri-français au format ODF. Le style utilisé pour marquer la prononciation est "Phonetic_20_form". Ce fichier, conçu à l'origine avec le logiciel Shoebox, garde dans le nom du style, l'étiquette utilisée pour marquer l'information avec Shoebox. Pour d'autres dictionnaires, le nom du style est plus cryptique. Par exemple, pour le dictionnaire bambara-français du père Charles Bailleul, qui a été rédigé directement avec WordPerfect, les traductions françaises sont notées avec le style "T21". Il est cependant possible de se repérer en comparant avec le fichier d'origine.

```
<text:span text:style-name="Phonetic_20_form"><text:span text:style-
name="T7">[àbàdárò]</text:span> </text:span>
```

Figure 12. Extrait d'article (phonétique) au format ODF XML

Le marquage explicite des informations peut donc être réalisé par le remplacement judicieux du balisage ODF par un balisage conforme au jeu d'éléments provisoire précédemment définis. Le balisage ODF reste cependant complexe, et il serait délicat de mettre au point, pour chaque dictionnaire, une transformation XSLT. Nous avons choisi de le réaliser par une succession d'opérations de recherche et de remplacement utilisant des expressions régulières car ces opérations peuvent être réalisées par les partenaires linguistes[30]. Pour l'exemple de la figure 12, une première expression régulière supprime l'élément "T7" :

```
s/<text:span text:style-name="T7">([^<]+)<\/text:span>//g
```

La deuxième expression remplace l'élément de style "Phonetic_20_form" par "bowodu" :

```
s/<text:span text:style-name="Phonetic_20_form">([^<]+)
<\/text:span>/<bowodu>$1<\/bowodu>/g
```

Le remplacement de tous les éléments aboutit au résultat de la figure 13. L'étape de marquage des informations est maintenant terminée.

```
<kalma>dole</kalma><bowodu>[dólè]</bowodu><naptu_curo_na-
hauyen>cu.</naptu_curo_nahauyen><maana>Karwu cəragəna bi wa-
jənama    tədidə.</maana><misal><version    təlam="ka">Kambi
nokkənidə, dole maararo ngaakke ingi falləkke kokki.</ver-
sion></version></misal><kalakta   təlam="fa">obligation</ka-
lakta>
```

Figure 13. Article converti avec des éléments de la version passerelle

Cette méthodologie simple à mettre en place convient bien aux dictionnaires construits par des projets (voir 1.4.3) qui utilisent généralement des logiciels spécifiques. Nous l'avons utilisée pour de nombreux dictionnaires tels que ceux du projet DiLAF ou le dictionnaire français-khmer de Denis Richer (Richer, 2007).

[30] Il s'agit aussi de favoriser le transfert de connaissances : les personnes ne connaissant pas les expressions régulières apprennet à les manipuler à l'occasion de ce projet.

6.3. Validation XML

Pour pouvoir utiliser des outils manipulant des fichiers XML, il faut que ceux-ci soient bien formés d'un point de vue de la syntaxe XML. Lors de l'étape précédente, les remplacements successifs ont été effectués sur un fichier au format ODF bien formé, aussi le résultat devrait-il être théoriquement bien formé. Il est apparu que, dans la pratique, ce n'est jamais le cas car les expressions régulières qui ont été successivement appliquées sont écrites par des linguistes récemment formés et donc peu aguerris à leur pratique. Il s'avère donc indispensable de vérifier que le fichier est bien formé. Cette étape permet de détecter certaines erreurs de balisage dues à l'étape précédente et de les corriger.

Un analyseur (parseur) XML est nécessaire pour vérifier la syntaxe du document. L'usage de logiciels gratuits étant privilégié, il est préférable d'utiliser un simple navigateur Web tel que FireFox. Celui-ci inclut un analyseur XML capable d'indiquer la localisation de la première erreur rencontrée dans le fichier. La figure 14 montre le résultat de l'analyse d'un fichier qui n'est pas bien formé (il manque le guillemet fermant de la valeur de l'attribut *@lamba*).

Figure 14. Affichage d'une erreur de syntaxe XML par Firefox

Une fois l'erreur localisée, il faut vérifier si elle se répète dans le fichier (ce qui est probable dans le cas d'une manipulation erronée lors d'une étape précédente). Si c'est le cas, il faut élaborer une expression régulière qui corrige l'erreur de manière systématique plutôt que d'effectuer des corrections au fur et à mesure du texte.

Le fichier XML est maintenant bien formé. Il est alors possible de le manipuler avec des outils XML.

6.4. Vérification des listes de valeurs

Pour un dictionnaire, la classe grammaticale prend sa valeur dans une liste fermée. L'étape de la vérification des informations prenant théoriquement leur valeur dans une liste fermée est importante car elle permet de détecter certaines erreurs introduites par des manipulations erronées lors des étapes précédentes ou déjà présentes dans le fichier d'origine avant conversion. Il s'agit aussi d'établir ou de corriger cette liste fermée qui, généralement, n'est pas connue ou comporte des erreurs.

Cette vérification systématique procède de manière destructrice et est donc opérée sur une copie du fichier à traiter. Il s'agit de ne garder que les valeurs à vérifier.

Dans l'exemple de la figure 13, l'expression suivante extrait la classe grammaticale balisée par <naptu_curo_nahauyen> :

```
s/^.*<naptu_curo_nahauyen>([^<]+)<\/.*$/$1/
```

La liste obtenue est ensuite triée par ordre alphabétique. Elle peut alors être scrutée pour détecter rapidement les irrégularités : si une valeur n'apparaît qu'une seule fois, il est fort probable que ce soit une erreur. Dans le dictionnaire kanouri-français, nous avons corrigé "kəld." en "kəl." (conjonction), "n." en "cu." (nom), "ono" en "cok." (idéophone), etc.

6.5. Corrections structurelles et sémantiques

Le fichier de travail étant bien formé, une feuille de style CSS peut être définie pour visualiser les données directement dans un navigateur. Un affichage compact et un style différent pour chaque type d'information permettent au linguiste de déceler les erreurs de structuration d'un article. Dans l'exemple de la figure 15, nous voyons qu'il manque l'exemple (en italique) pour l'entrée "acca ambe".

Le langage XSLT permet de modifier les données avant l'affichage. Il est alors possible, par exemple, d'ajouter comme identifiant unique d'un article, son mot-vedette puis, pour chaque renvoi ou synonyme de définir un lien hypertexte pointant vers l'article correspondant. Lorsque le linguiste parcourt le fichier, il peut alors cliquer sur les liens hypertextes (signalés par un soulignement) pour vérifier que les renvois et synonymes font également l'objet d'articles dans le dictionnaire. Dans l'exemple de la figure 15, l'entrée "acca" contient un renvoi vers l'entrée "hâjjà". Le mot est souligné, ce qui indique la présence d'un lien hypertexte.

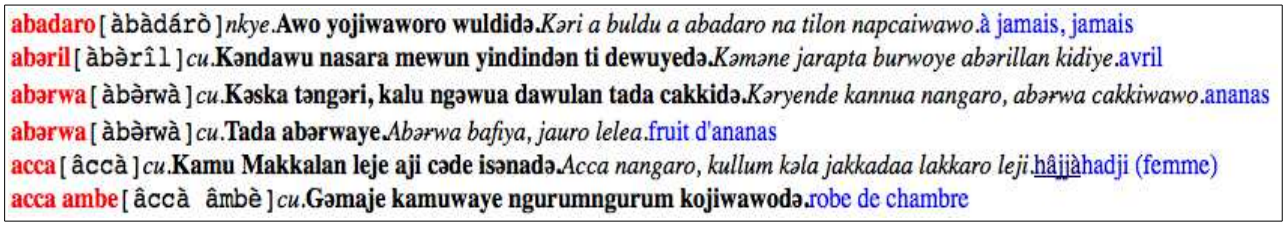

Figure 15. Vue compacte dans un navigateur

Lorsque des programmes XSLT s'avèrent nécessaires il est indispensable que des informaticiens participent au projet de conversion.

Cette étape de visualisation des données est essentielle, non seulement pour détecter des erreurs, mais aussi parce qu'elle permet aux linguistes de comprendre les avantages de l'encodage des données en XML, en particulier le fait que différentes mises en formes (styles) peuvent être associées aux articles balisés. En apprenant des rudiments du langage CSS, ils peuvent alors modifier eux-mêmes les feuilles de style.

7. Conversion vers le format passerelle à l'aide d'outils spécialisés

Lorsque les dictionnaires sont susceptibles de contenir beaucoup d'erreurs de structuration ou que celle-ci est implicite et relâchée, ce qui est souvent le cas avec des dictionnaires destinés à l'impression et rédigés par un seul auteur, la conversion fondée sur des expressions régulières devient difficile. Il est alors possible d'utiliser des outils de récupération beaucoup plus puissants basés sur une grammaire de description de la structure explicite que l'on vise à récupérer. Des générateurs d'analyseurs lexicaux et syntaxiques robustes comme Lex et Yacc (ou Flex et Bison) peuvent être utilisés. Il existe également des outils plus évolués programmés sur mesure pour la récupération de dictionnaires tels que RECUPDIC écrit par Hai Doan-Nguyen dans le cadre de sa thèse (Doan-Nguyen, 1998). Toutefois, l'usage de ces outils nécessite des compétences informatiques élevées excluant, de fait, les linguistes et lexicographes de cette étape.

La traduction par analyse utilise un formalisme nommé H-grammar. L'utilisateur décrit la grammaire du dictionnaire à récupérer en H-grammar. Il ajoute ensuite les actions de construction d'objets et de détection d'erreurs. La détection d'erreurs permet de corriger automatiquement les erreurs les plus fréquentes. Si un détail est faux dans un article, il n'est pas rejeté en bloc. Un compilateur utilise ensuite la description pour construire l'ensemble d'objets constituant une représentation structurée du dictionnaire.

7.1. Exemple d'article avant récupération

La figure 16 représente un article du dictionnaire BABEL, un dictionnaire monolingue anglais d'abréviations dans le domaine informatique (Kind, 1997) au format d'origine avant la récupération.

```
.COM    Command (file name extension) +
        Commercial Business (Domain Name) [Internet]
```

Figure 16. Article de BABEL avant récupération

Il arrive fréquemment qu'un article ne vérifie pas la syntaxe indiquée par ses auteurs. Dans BABEL, par exemple, on peut trouver des parenthèses en trop, des "+" oubliés, etc. Il faut alors normaliser. L'article de la figure 16 donné en exemple est correct.

Cet article a une structure implicite : c'est sa présentation qui reflète sa structure. Les différentes informations sont distinguées par leur mise en forme et des caractères spéciaux (les parenthèses "()", le "+", les crochets "[]", etc.). Il faut donc modifier cet article pour le réutiliser. Pour récupérer l'article avec l'outil H-grammar, il faut écrire une grammaire de récupération dans ce formalisme. Voici quelques éléments concernant l'écriture d'une grammaire H-grammar.

7.2. Grammaire de récupération

Une grammaire de récupération H-grammar se compose de six mots-clefs suivis de leurs instructions :

`#grammar` : nom de la grammaire ;

`#syntax-rules` : règles d'analyse syntaxique pour la récupération ;

`#start-symbol` : symbole de départ de la grammaire ;

`#lexical-rules` : règles d'analyse lexicale pour construire les items lexicaux ;

`#lexical-order` : ordre de préférence entre les items lexicaux ;

`#working-code` : fonctions Common Lisp intégrables dans les règles syntaxiques.

La figure 17 montre une grammaire H-grammar permettant la récupération des articles de BABEL. Dans les règles d'analyse syntaxique, le caractère "§" correspond au symbole nul, le caractère ">" devant un nom de symbole comme ">hwd" indique que ce symbole est terminal. Dans les règles d'analyse lexicale, la notation "_¨10" signifie que le symbole est composé de 10 caractères, le symbole `to-cparen` correspond à une chaîne de caractères se terminant par une parenthèse fermante, le symbole `to-cbrak` correspond à une chaîne de caractères se terminant par un crochet fermant.

```
#grammar babel-glossary /* Acquisition du glossaire BABEL */
#syntax-rules
:1: babel-entry(;entry) -> >hwd(;hwd) body(;body) -- (#!babel((trim-whites hwd) body); entry).
:2: body(;body) -> sense(;S1) sense*(;S*) -- cons(S1 S*; body).
:3: sense(;S)-> >exps(;exps) expl?(;expl) subj?(;subj) -- (#!sense((trim-whites exps) (if expl (trim-
whites expl)) (if subj (trim-whites subj))); S).
:4: expl?(;expl) -> "(" >to-cparen(;expl) ")".
:5: expl?(;expl) -> §(nil;expl).
:6: subj?(;subj) -> "[" >to-cbrak(;subj) "]".
:7: subj?(;subj) -> §(nil; subj).
:8: sense*(;S*) -> "+" sense(;S1) sense*(;S*1) -- cons(S1 S*1; S*).
:9: sense*(;S*) -> §(nil; S*).
#start-symbol babel-entry /* départ de la grammaire*/
#lexical-rules hwd -> _¨10. /* Headword prend 10 cars */
exps -> ![+([]*. to-cparen -> >()]. to-cbrak -> >[\]].
#lexical-order ("+" "(" ")" "[" "]" hwd exps expl subj)
#working-code (sia-defclass babel () (hwd body))
```

```
(sia-defclass sense () (exps expl subj))
(defun trim-whites (string) (string-trim '(#\Space #\Tab #\Newline) string))
```

Figure 17. Grammaire H-grammar de récupération de BABEL

Expliquons maintenant les règles d'analyse syntaxique **#syntax-rules** :

`sense*` est un élément non-terminal. Le signe **`*`** n'est pas l'opérateur de Kleene mais signale une liste de **`sense`**.

`expl?` est un élément non terminal. Le signe **`?`** Note la présence ou l'absence de **`expl`**.

La règle 1 produit un article babel **`babel-entry`** à partir du mot-vedette **`hwd`** et d'un corps **`body`**.

La règle 2 produit un corps **`body`** à partir d'une liste de sens **`sense*`**.

La règle 3 produit un sens à partir d'une définition **`exps`**, d'une explication **`expl`** et d'un domaine **`subj`**.

Les règles 4 et 5 produisent une explication **`expl`** à partir d'un texte entre parenthèses "**()**".

Les règles 6 et 7 produisent un domaine **`subj`** à partir d'un texte entre crochets "**[]**".

Les règles 8 et 9 produisent une liste de sens **`sense*`** à partir de deux sens **`sense`** séparés par un "+".

Cette grammaire est interprétée ensuite par un compilateur Macintosh Common Lisp qui produit des objets LISP correspondant aux articles récupérés.

7.3. Exemple d'article après récupération

La figure 18 montre le résultat de la récupération de l'article BABEL original après compilation avec H-grammar :

```
(BABEL
  (HWD . ".COM")
  (BODY LIST
    (SENSE
      (EXPS . "Command")
      (EXPL . "file name extension")
      (SUBJ . NIL))
    (SENSE
      (EXPS . "Commercial Business")
      (EXPL . "Domain Name")
      (SUBJ . "Internet"))))
```

Figure 18. Article de BABEL après récupération (objet LISP)

L'article obtenu est un objet LISP dans lequel les informations sont marquées explicitement. Il est alors facile de les réutiliser automatiquement pour produire de nouveaux ensembles lexicaux, ou encore de transformer l'article en un autre format exprimant une structure explicite, comme XML par exemple, et obtenir ainsi un article dans un format passerelle.

8. Du format passerelle vers le format cible

Dans cette partie, nous montrons comment représenter des ressources dans un format cible en conformité avec la partie informative de la version 16 de la DTD de la norme LMF 24613:2008. Les exemples sont tirés des dictionnaires du projet DiLAF.

Chaque information est présentée sous forme d'une valeur de l'attribut *@val* de l'élément *"feat"* (pour "feature") et associée à l'attribut *@att* qui renseigne quant à la nature de l'information.

8.1. Ressource lexicale

Un dictionnaire LMF est une ressource lexicale pour laquelle est indiquée la référence ISO 639-3 du codage des noms de langues. Elle englobe le lexique dans lequel figure le code de la langue source du dictionnaire et l'ensemble des articles.

Nom francophone de la langue	**Nom anglophone de la langue**	**Code ISO 639-3**
bambara	Bambara	bam
français	French	fra
haoussa	Hausa	hau
kanouri	Kanuri	kau
soŋay zarma	Zarma	dje
tamajaq	Tamajaq	ttq

Figure 19. Codes ISO 639-3 des noms des langues des dictionnaires du projet DiLAF

```
<LexicalResource>
 <GlobalInformation>
  <feat att="languageCoding" val="ISO 639-3"/>
 </GlobalInformation>
 <Lexicon>
  <feat att="language" val="kau"/>
  <LexicalEntry id="a">
   ...
```

Figure 20. Début du dictionnaire kanouri-français au format cible LMF

Les articles figurent dans le lexique (élément <*Lexicon*>), chacun est encadré par l'élément <*LexicalEntry*> associé à un identifiant unique.

Nous détaillons maintenant la représentation des parties composant une notice.

8.2. Catégorie lexicale

La catégorie lexicale d'une entrée est directement signalée dans l'entrée lexicale comme valeur associée à *"partOfSpeech"*. Par conséquent, une entrée de dictionnaire possédant plusieurs catégories lexicales doit être traduite en plusieurs entrées lexicales.

Exemple : en kanouri, *dole* « obligation » est un nom commun.

```
    <feat att="partOfSpeech" val="commonNoun"/>
```

Figure 21. Notation de la catégorie lexicale dans LMF

~~Le catalogue des catégories de parties du discours de l'ISO figure en Annexe B.~~

Le travail sur des langues peu dotées peut amener à découvrir de nouvelles catégories de parties du discours telle que “idéophone” pour le kanouri. Elles seront alors communiquées à l'ISO pour être incluses au catalogue existant.

8.3. Lemme

Les formes orthographique et phonétique d'une entrée sont associées, respectivement aux attributs *@writtenForm* et *@phoneticForm* et enchassées dans le bloc <*Lemma*>.

Voici la partie lexicale de l'entrée **`dole`** du dictionnaire kanouri-français :

```
<LexicalEntry id="dole">
  <feat att="partOfSpeech" val="commonNoun"/>
  <Lemma>
    <feat att="writtenForm" val="dole"/>
    <feat att="phoneticForm" val="dólè"/>
  </Lemma>
```

Figure 22. *Partie lexicale de l'entrée "dole" au format LMF*

8.4. Variations morphologiques

L'élément <*Wordform*> entoure les informations portant sur les variations morphologiques exprimées en extension car elles ne sont pas régulières, comme les pluriels irréguliers ou encore l'état d'annexion, fréquents dans le dictionnaire tamajaq-français. Les formes au singulier et au pluriel figurent explicitement en combinant deux éléments <feat>, l'un indiquant la forme ("*writtenForm*") et l'autre le nombre (*"grammaticalNumber"*).

Voici un exemple avec l'entrée *ălawwa* « lance-pierres » :

```
<WordForm>
  <feat att="writtenForm" val="ălawwa"/>
  <feat att="grammaticalNumber" val="singular"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ilawwan"/>
  <feat att="grammaticalNumber" val="plural"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ălawwa"/>
  <feat att="grammaticalNumber" val="singular"/>
  <feat att="contextualVariation" val="annexion"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ălawwan"/>
  <feat att="grammaticalNumber" val="plural"/>
  <feat att="contextualVariation" val="annexion"/>
</WordForm>
```

Figure 23. Variations morphologiques de l'entrée *ălawwa* au format LMF

8.5. Bloc sémantique

Le bloc sémantique est délimité par l'élément <Sense>. Il est possible d'avoir plusieurs blocs sémantiques. Chaque bloc sémantique regroupe la définition et les éventuelles relations sémantiques de synonymie ou antonymie.

La définition est délimitée par l'élément <Definition>. L'équivalent français est délimité par l'élément <Equivalent>.

Les relations sémantiques sont traitées différemment selon la nature de la cible mise en relation. Si la cible est une entrée lexicale monosémique, elle ne possède qu'un seul bloc sémantique, l'identi-

fiant de l'entrée lexicale visée peut donc être directement indiqué comme valeur de l'attribut *@RelatedForm*. En revanche, si la cible est une entrée lexicale possédant plusieurs blocs sémantiques, il est nécessaire de préciser celui qui est visé. Dans ce cas il est nécessaire d'utiliser l'élément <SenseRelation> et d'indiquer l'identifiant du bloc sémantique visé comme valeur de l'attribut *@targets*.

Ce distingo apparaît peu adapté à une ressource lexicale en évolution car, en cas d'enrichissement du dictionnaire par l'ajout d'un sens à une entrée par ailleurs en relation sémantique avec une autre entrée, il faudrait modifier les représentations de ces entrées. Dans cette situation, il est préférable d'identifier tous les blocs sémantiques par un identifiant unique susceptible de servir de référence.

Exemple : en kanouri, *alau* « créature » est synonyme d'*alitta*. Ces deux entrées sont monosémiques. Chaque bloc sémantique est identifié par l'entrée concaténée au chiffre 1.

```
<LexicalEntry id="alau">
  <Lemma>
    <feat att="writtenForm" val="alau"/>
    <feat att="phoneticForm" val="álâu"/>
  </Lemma>
  <Sense id="alau1">
    <Definition>
      <feat att="text" val="Awo duwon dunia adəlan Kəmandeye alakcənadə."/>
    </Definition>
    <SenseRelation targets="alitta1">
      <feat att="type" val="synonym"/>
    </SenseRelation>
  </Sense>
</LexicalEntry>
```

Figure 24. Entrée *alau* comportant un renvoi vers l'entrée *alitta* au format LMF

La relation d'antonymie se signale par la valeur l'usage de *"antonym"* comme valeur de type sémantique tandis qu'une relation sémantique non précisée sera caractérisée par la valeur *"seeAlso"*.

Les exemples d'usage présentent de potentielles difficultés que révèle l'examen attentif du dictionnaire bambara-français. Ce dictionnaire présente l'originalité de traduire tous les exemples en français et de les présenter sous une forme tonalisée. De plus, certains proverbes, dont la traduction ne permet pas de comprendre le sens, sont munis d'une explication supplémentaire. Un exemple d'usage peut donc présenter jusqu'à quatre caractéristiques auxquelles pourraient s'ajouter d'autres informations telles la traduction dans d'autres langues.

exemple	bamusokɔrɔ kɔ tɛ a ŋɛɲɛ ni a ma dɛnɛ ye.
forme tonalisée	bàmùsòkɔrɔ kɔ tɛ à ŋɛɲɛ ni à ma dɛ̀nɛ ye.
traduction	le dos de la chèvre ne lui démange pas, tant qu'elle n'a pas vu un mur.
explication	créer des besoins, rôle de la publicité.

Figure 25. Exemple d'usage pour l'entrée "bamusokɔrɔ" ("chèvre adulte") issu du dictionnaire bambara-français

Un exemple d'usage est entouré par l'élément <*Context*>. Chacune des caractéristiques est entouré par l'élément <*TextRepresentation*>. L'exemple précédent devient :

```
<Context>
 <TextRepresentation>
  <feat att="language" val="bam"/>
  <feat att="writtenForm" val="bamusokɔrɔ kɔ tɛ a
      ŋɛɲɛ ni a ma dɛnɛ ye."/>
 </TextRepresentation>
 <TextRepresentation>
```

```
  <feat att="language" val="bam"/>
  <feat att="phoneticForm" val="bàmùsòkɔrɔ kɔ tɛ à
      ŋɛɲɛ ni à ma dɛ̀nɛ ye."/>
 </TextRepresentation>
 <TextRepresentation>
  <feat att="language" val="fra"/>
  <feat att="writtenForm" val="le dos de la chèvre
      ne lui démange pas, tant qu'elle n'a pas vu
      un mur."/>
 </TextRepresentation>
 <TextRepresentation>
  <feat att="language" val="fra"/>
  <feat att="explanation" val="créer des besoins,
      rôle de la publicité."/>
 </TextRepresentation>
</Context>
```

Figure 26. Exemple d'usage pour l'entrée "bamusokɔrɔ" ("chèvre adulte") au format LMF

9. Mise en ligne sur une plate-forme de gestion de ressources en ligne

Il existe de nombreux logiciels de gestion de ressources en ligne conçus, pour la plupart, spécifiquement pour une ressource particulière. Très peu de logiciels permettent de manipuler des ressources XML sans modifier leur structure. Il existe néanmoins quelques outils commerciaux tels que TshwaneLex[31] ou le Dictionary Publishing System[32] de IDM. Dans un contexte de travail sur les langues peu dotées, nous préférons cependant utiliser un outil gratuit et en source ouverte (open-source). C'est pourquoi nous avons choisi la plate-forme générique de gestion de ressources lexicales en ligne Jibiki.

9.1. Description de la plate-forme utilisée

Jibiki est une plate-forme générique en ligne dédiée à la manipulation de ressources lexicales avec gestion d'utilisateurs et groupes, consultation de ressources hétérogènes et édition générique d'articles de dictionnaires. La plate-forme est programmée entièrement en Java, elle est fondée sur l'environnement "Enhydra". Toutes les données sont stockées au format XML dans une base de données (Postgres).

Ce site Web communautaire propose principalement deux services : une interface unifiée permettant d'accéder simultanément à de nombreuses ressources hétérogènes (dictionnaires monolingues, dictionnaires bilingues, bases multilingues, etc.) et une interface d'édition spécifique pour contribuer directement aux dictionnaires disponibles sur la plate-forme.

Plusieurs projets de construction de ressources lexicales ont utilisé ou utilisent cette plate-forme avec succès. C'est le cas par exemple du projet GDEF de dictionnaire bilingue estonien-français (Chalvin & Mangeot, 2006) ou plus récemment du projet MotÀMot (Mangeot, 2009). Le code de cette plate-forme est disponible gratuitement en source ouverte en téléchargement depuis la forge du laboratoire LIG[33].

9.2. Import d'une ressource existante sur la plate-forme

L'unique condition à respecter pour importer une ressource existante sur la plate-forme Jibiki est que celle-ci soit au format XML. En effet, un système de pointeurs communs dans des structures hétérogènes permet de manipuler les ressources sans modifier leur structure. Chaque pointeur est

[31] tshwanedje.com/tshwanelex/
[32] www.idm.fr/products/dictionary_writing_system_dps/27/
[33] ligforge.imag.fr/projects/jibiki/

indexé dans une base de données et permet d'effectuer une recherche rapide. Ce système est appelé Common Dictionary Markup (CDM). Il existe des pointeurs communs prédéfinis pour les objets lexicaux que l'on trouve fréquemment dans la plupart des dictionnaires (voir Figure 29). Il est possible également de définir des pointeurs spécifiques à une ressource.

L'importation d'un dictionnaire peut donc se faire aussi bien au format passerelle qu'au format cible ; il peut s'agir d'un dictionnaire structuré suivant les recommandations de la TEI comme de la norme LMF, etc.

Pour préparer l'import, il est nécessaire de décrire la structure du dictionnaire et des volumes dans des fichiers de méta-données. Ceux-ci seront utilisés par la plate-forme lors de l'import.

Le dictionnaire et sa macrostructure sont décrits à l'aide d'un formulaire HTML dans un premier fichier de méta-données (voir figure 27).

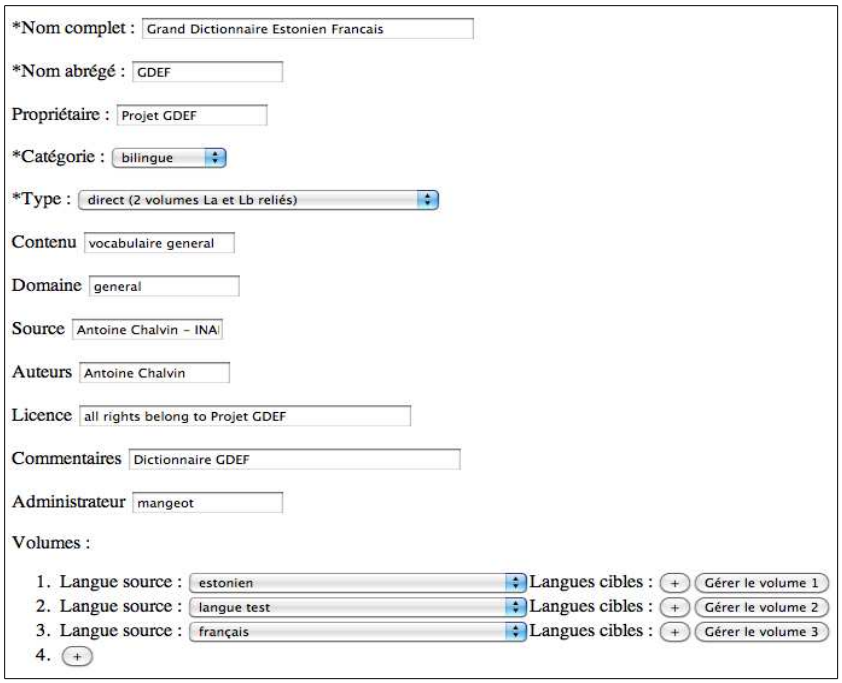

Figure 27. Méta-données du dictionnaire GDEF estonien-français

Chaque volume et sa microstructure sont décrits dans un fichier de méta-données distinct (voir figure 28).

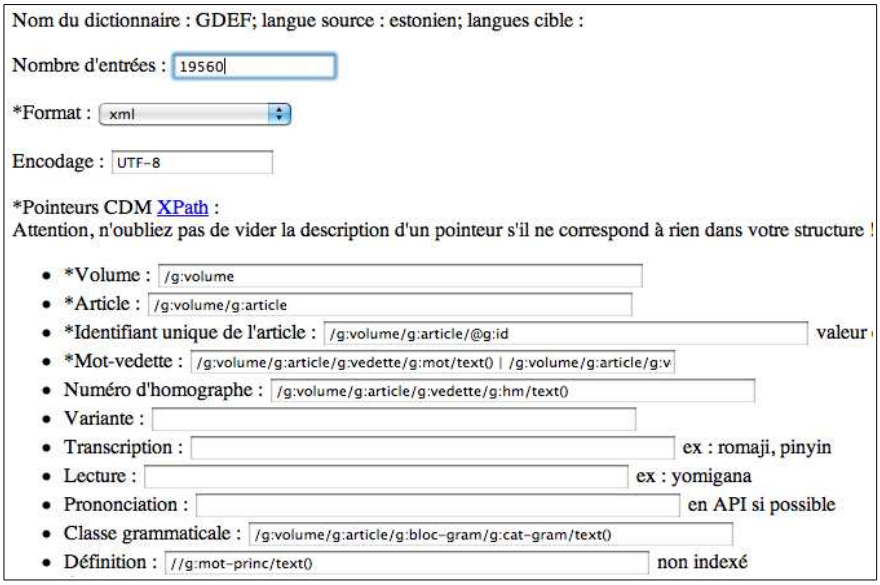

Figure 28. Méta-données du volume estonien du dictionnaire GDEF

La microstructure est décrite à l'aide de pointeurs communs identifiant les mêmes éléments d'information, les pointeurs CDM utilisant le standard XPath. La figure 29 montre la valeur des pointeurs CDM pour les dictionnaires FeM (français-anglais-malais), Oxford-Hachette (français-anglais) et JMdict (japonais-multilingue).

Pointeur CDM	FeM	OHD	JMdict
Volume	/volume	/volume	/JMdict
Entry (article)	/volume/entry	/volume/se	/Jmdict/entry
Entry ID (identifiant de l'article)	/volume/entry/@id		/Jmdict/entry/ent_seq /text()
Headword (mot-vedette)	/volume/entry/headword/ text()	/volume/se/hw/text()	/Jmdict/entry/k_ele/ keb/text()
Pronunciation (prononciation)	/volume/entry/prnc/text()	/volume/se/pr/ph/text()	
PoS (catégorie grammaticale)	//sense-list/sense/pos-list/text()	/volume/se/hg/ps/text()	/Jmdict/entry/sense/ pos/text()
Domain (domaine)		//u/text()	
Example (exemple)	//sense1/expl-list/expl/fra	//le/text()	/Jmdict/entry/sense/ gloss/text()

Figure 29. Pointeurs CDM pour les dictionnaires FeM, OHD et JMdict

9.3. Consultation et édition des données en ligne

Les dictionnaires mis en ligne sur une instance de la plate-forme Jibiki sont consultables en ligne par le grand public. L'interface de consultation est multi-critères et multi-ressources. Elle s'appuie sur les pointeurs CDM décrits précédemment.

Figure 30. Interface de recherche avancée de la plate-forme Jibiki

Il est possible de contribuer directement en ligne. Le système de gestion de permissions intégré à la plate-forme permet de demander la révision et la validation des contributions par un responsable avant leur intégration finale. La plate-forme autorise également, grâce à une interface de programmation (API), l'utilisation à distance par d'autres logiciels ou services.

L'éditeur est fondé sur un modèle d'interface HTML instancié par l'article à éditer. Le modèle peut être généré automatiquement depuis une description de la structure de l'entrée à l'aide d'un schéma XML. Ce modèle peut être modifié ensuite pour améliorer le rendu à l'écran. La seule information nécessaire à l'édition d'un article de dictionnaire est donc le schéma XML représentant la structure de cette entrée.

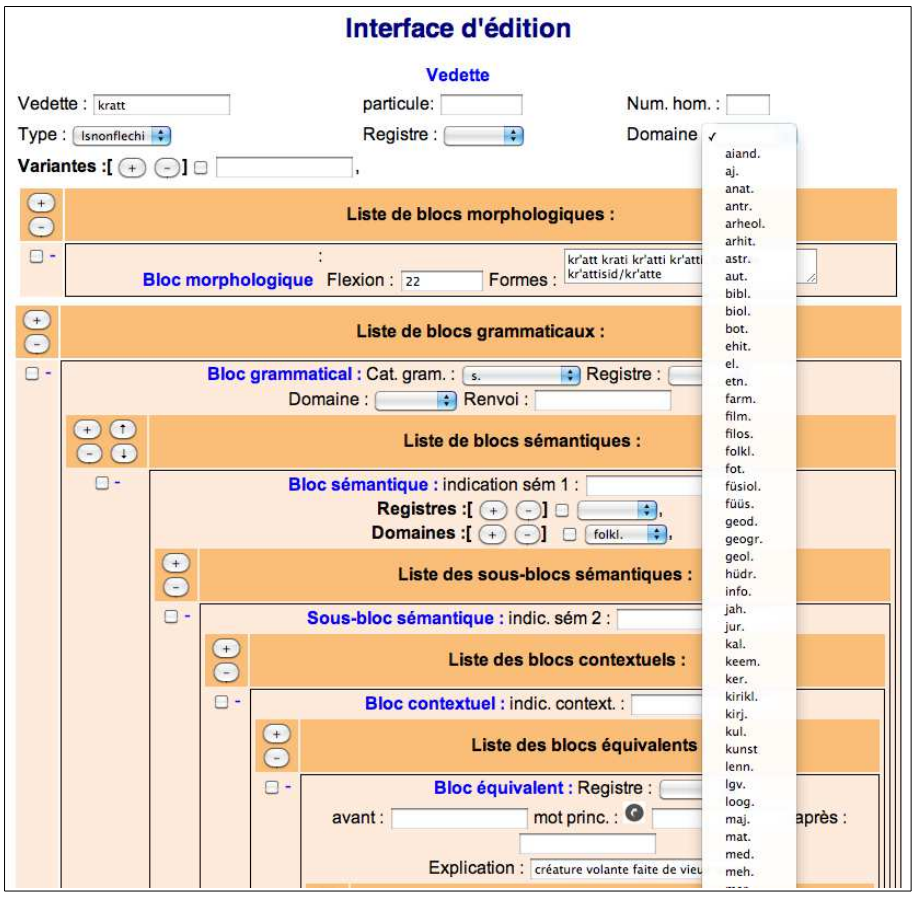

Figure 31. Interface d'édition d'un article estonien du dictionnaire GDEF

Conclusion

Récupérer des dictionnaires éditoriaux pour les convertir en des dictionnaires électroniques s'est révélé être un processus délicat, semé d'embûches, mais pourtant réalisable. La démarche est économiquement valide car le temps de travail et le coût sont minimisés : un dictionnaire peut être converti en quelques semaines. La part de travail humain reste importante et l'horizon d'une conversion entièrement automatisée reste donc lointain. Toutefois le gain est bien réel par rapport aux années d'investissements qu'exigerait la création d'un dictionnaire électronique. Il apparaît que l'indispensable collaboration entre linguistes et informaticiens constitue une aubaine pour effectuer des transferts de connaissances. Non seulement, les connaissances de chacun sont indispensables, mais il faut aussi que chaque partenaire accroisse ses compétences dans le savoir de l'autre discipline.

Ces ressources électroniques ainsi élaborées sont développées a priori pour servir le traitement automatique des langues. Au-delà de cet usage spécialisé, les locuteurs des langues peu dotées peuvent aussi y avoir accès car elles sont conçues pour être visibles sur la Toile. Dans des contextes de pauvreté où les locuteurs n'ont souvent jamais vu un dictionnaire de leur langue mais où l'accès au web s'améliore, l'enjeu quant aux retombées pour les populations est majeur. De plus, la visibilité des résultats constitue une motivation supplémentaire pour les personnes participant à la conversion.

Les dictionnaires ainsi convertis sont incomplets et loin d'être parfaits, et des corrections devront être apportées. Toutefois, ils constituent un marche-pied vers d'autres développements : construction de corpus bilingues quand les entrées présentent des exemples traduits ; mise au point d'analyseurs morphologiques portant sur les flexions quand ces informations sont disponibles ; etc. La mise à disposition de ces ressources peut susciter la vocation de chercheurs ne travaillant pas sur ces langues, et donc accroître le potentiel de recherches.

Financement

Le projet DiLAF est financé par le Fonds Francophone des Inforoutes de l'Organisation Internationale de la Francophonie.

Œuvres citées

Andries, P. (2004). Proposition d'ajout de l'écriture tifinaghe. Organisation internationale de normalisation. Jeu universel des caractères codés sur octets (JUC). ISO/IEC JTC 1/SC 2 WG 2 N2739.

Charles, B. (1996). Dictionnaire bambara-français.

Berment, V. (2004). Méthodes pour informatiser des langues et des groupes de langues "peu dotées". Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, Grenoble, France.

Buseman, A., Buseman, K., Jordan, D., Coward, D. (2000). The linguist's shoebox: tutorial and user's guide: integrated data management and analysis for the field linguist. volume viii. Waxhaw, North Carolina : SIL International.

Calvet, L.-J. (1987). La guerre des langues. Paris, Payot.

Calvet, L.-J. (1996). Les politiques linguistiques. Paris, PUF.

Chalvin, A. Mangeot, M. (2006). Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français. Actes d'EURALEX 2006, Turin, Italie, 6-9 septembre.

Chanard, C. Popescu-Belis, A. (2001). Encodage informatique multilingue : application au contexte du Niger. Cahiers du Rifal (cont. Terminologies Nouvelles), n. 22, pages 33-45.

Desgraupes, B. (2005). Passeport pour Unicode. Vuibert France. (ISBN 2-7117-4827-8)

Diki-Kidiri M. (2004). Multilinguisme et politiques linguistiques en Afrique. Colloque Développement durable, leçons et perspectives, Ouagadougou.

Doan-Nguyen H. (1998). Techniques génériques d'accumulation d'ensembles lexicaux structurés à partir de ressources dictionnairiques informatisées multilingues hétérogènes. Thèse de nouveau doctorat, Spécialité Informatique, Institut National Polytechnique de Grenoble, 168 pages.

Enguehard, C. (2009). Les langues d'Afrique de l'Ouest : de l'imprimante au traitement automatique des langues. Sciences et Techniques du Langage, 6, pages 29-50. (ISSN 0850-3923)

Enguehard C., Kane S., Mangeot M., Issouf M., Sanogo M-L. (2012). Vers l'informatisation de quelques langues d'Afrique de l'Ouest. JEP-TALN-RECITAL 2012, Workshop TALAf 2012: NLP for African Languages, pages 27-40, Grenoble, France, juin.

Francopoulo G., Bel N., George M., Calzolari N., Monachini M., Pet M., and Soria C. (2009). Multilingual resources for NLP in the lexical markup framework (LMF). Language Resources and Evaluation, 43(1), pages 57–70, March.

Gut Y. Ramli P. R. M., Yusoff Z., Kim Ch. Ch., Samat S. A., Boitet Ch., Nédobejkine N., Lafourcade M. et al. (1996). Kamus Perancis-Melayu Dewan, dictionnaire français-malais. Dewan Bahasa Dan Pustaka, Kuala Lumpur, 667 p.

Haralambous Y. (2004). Fontes & codages. O'Reilly France.

Kind, I. (1997). A Glossary of Computer Oriented Abbreviations and Acronyms, Version 97B.

Mangeot M. (2001). Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Thèse de nouveau doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, 27 septembre, 280 pages.

Mangeot M., Sérasset G., Lafourcade M. (2003). Construction collaborative de données lexicales multilingues : le projet Papillon. Revue TAL, Vol. 44:2/2003, pages 151-176.

Mangeot M. (2009). Projet Mot à mot : élaboration d'un système lexical multilingue par le biais de dictionnaires bilingues. Actes des journées scientifiques LTT 2009, Lisbonne, Portugal, 15-17 octobre, 12 pages.

Mijinguini, A. (2003). Dictionnaire élémentaire hausa-français.

Modi, I. (2007). Les caractères tifinagh dans Unicode. Actes du colloque international "le libyco-berbère ou le tifinagh : de l'authenticité à l'usage pratique", pages 241-254, ed. Haut Commissariat à l'amazighité (HCA), pages 21-22, mars, Alger.

Arrêté 212-99 de la République du Niger. (1999). Alphabet haoussa.

Arrêté 213-99 de la République du Niger. (1999). Alphabet kanouri.

Arrêté 214-99 de la République du Niger. (1999). Alphabet tamajaq.

Programme Décennal du Développement de l'Éducation (PDDE), Niger. (2003). Enseignement bilingue, pages 121-132.

Richer, D. (2007) Dictionnaire français-khmer (en phonétique), Cambodge : Édition DR, 696 p.

Romary L., Salmon-Alt S., Francopoulo G. (2004). Standards going concrete: from LMF to Morphalou. Proceedings of the COLING Workshop on Enhancing and Using Electronic Dictionaries. ElectricDict'04, pages 22–28, Stroudsburg, PA, USA: Association for Computational Linguistics.

Programme de soutien à l'éducation de base (Soutéba). (2004). Dictionnaire kanouri-français destiné pour le cycle de base 1.

Programme de soutien à l'éducation de base (Soutéba). (2007). Dictionnaire tamajaq-français destiné à l'enseignement du cycle de base 1.

Streiter O., Scannell K., Stuflesser M. (2006). Implementing NLP projects for non-central languages : Instructions for funding bodies, strategies for developers. Machine Translation, volume 20.

Text Encoding and Interchange P4. Guidelines for Electronic Text Encoding and Interchange. 2004. www.tei-c.org/release/doc/tei-p4-doc/html/

Umaru, I. A. (1997). Zarma ciine - kaamuusu kayna. Editions Alpha.

Unicode. (2005).The Unicode Standard 4.1. Tifinagh, range 2D30-2D7F.

Unicode. (2012). About Versions of the Unicode Standard, 2012. www.unicode.org/versions/, consulté le 5 juillet 2012.

Adresse des auteurs :

Mathieu Mangeot
GETALP-LIG
41 rue des mathématiques
BP 53
38041 Grenoble Cedex 9
France

Chantal Enguehard
LINA
2, rue de la Houssinière
BP 92208
44322 Nantes Cedex 03
France