

# UN TRANSCRIPTEUR BRAILLE IDÉAL?

PROBLÉMATIQUE, MODÉLISATION, CONCEPTION ET MISE EN ŒUVRE D'UN PROTOTYPE

Bruno MASCRET\*, Alain MILLE\*, Marc OLLIER\*\*

\*Laboratoire d'InfoRmatique en Image et Système d'information (LIRIS),  
UMR 5205, équipe Silex (Supporting Interactions and Learning by Experience),  
Université Claude Bernard de Lyon 1 (France)

\*\* Institut National Supérieur de formation et de recherche pour l'éducation  
des jeunes Handicapés et les Enseignements Adaptés (INS HEA), Suresnes (France)

**Résumé— L'accès à l'information numérique des déficients visuels, malgré les réels efforts de sensibilisation entrepris, reste encore une source de difficultés. Un des principaux facteurs limitant l'accessibilité d'un document numérique aux brailistes provient de la complexité à transcrire des documents composites (expressions scientifiques, brailles littéraire et abrégé, musique...), chaque contenu utilisant des normes particulières et devant être préalablement identifié. Cet article propose une brève introduction à la problématique de la transcription du braille, puis détaille la modélisation, la conception et la mise en œuvre d'un transcrip-teur « idéal » : simplifiant la chaîne de transcription, multimodal, indépendant des systèmes d'exploitation, polymorphe, et suffisamment rapide pour le traitement en temps réel.**

**Mots-clefs— Accessibilité, Braille, transcription, documents composites, adaptation de documents.**

## I. INTRODUCTION

Les nouvelles technologies, notamment la mise en place désormais courante de CMS<sup>1</sup>, plateformes pédagogiques et autres outils de communication par le web ou le courrier numérique, facilitent de plus en plus l'accès aux ressources numériques.

Que ce soit dans le milieu scolaire ou celui de l'entreprise, la quantité d'informations mises en ligne ne cesse d'augmenter. Aussi, le recours à cet outil de publication est-il privilégié.

Mais pour les personnes déficientes visuelles, ceci se traduit par l'émergence de nouvelles situations de handicap : nous allons montrer dans la première partie de cet article que ces nouveaux modes de communication occasionnent une surcharge de travail, principalement temporelle, pour les déficients visuels [5][6][7].

Nous nous attacherons par la suite plus particulièrement aux utilisateurs de braille, et après une présentation rapide des problématiques liées au braille et des outils de transcription existants, nous présenterons le transcrip-teur NAT qui a pour ambition de minimiser les situations de handicap en proposant une nouvelle approche intégrée de la transcription automatisée.

Nous terminerons en essayant de répondre à la question posée par le titre du présent article : « Peut-on envisager un transcrip-teur braille idéal? ».

1 Content Management System : Système de gestion de contenu

## II. PROBLÉMATIQUES LIÉES AUX DOCUMENTS NUMÉRIQUES ET AU BRAILLE

L'objectif de cette partie est de permettre, sans entrer dans les détails, de familiariser le lecteur avec le monde du handicap et de la déficience visuels.

Nous insistons sur la différence<sup>2</sup> entre la déficience, caractéristique objective, physique, neuro-biologique ou cérébrale, (la plupart du temps permanente) d'une personne, et la situation de handicap qui est contextuelle et peut disparaître grâce à une modification de l'environnement.

La « situation de handicap » est caractérisée par un début, une durée, et éventuellement une fin. Par exemple, si cet article était rédigé en khmer, et que vous deviez le lire, vous seriez en situation de handicap si vous ne connaissez pas cette langue, et le resteriez jusqu'à ce qu'il soit traduit (ou que vous ayez appris la langue, trouvé un interprète, etc.).

### 1. Introduction à la problématique des déficients visuels

#### 1.1 L'utilisateur non-voyant sur poste informatique

Si les avantages des outils de communication numérique sont nombreux pour la plupart des utilisateurs (économie de temps pour la diffusion, organisation, réduction de la consommation de papier...), un déficient visuel en revanche va être confronté à une nouvelle forme de situation de handicap.

En effet, même si l'accessibilité de ces moyens de communication est de mieux en mieux prise en compte par les adaptations logicielles spécifiques aux déficients de la vue et le développement du « design for all<sup>3</sup> », elle s'attache essentiellement à restituer la structure, le contenu des sites ou des applications, et leur mise en forme, sans chercher à diminuer le temps nécessaire pour pouvoir consulter l'information ou le document recherchés [7][8].

L'accès aux informations pour un non-voyant est séquentiel (lecture vocale, plage de caractères brailles) et ne permet pas de se positionner directement sur les éléments intéressants ou d'appréhender un document dans sa globalité.

L'accessibilité des contenus numériques par ceux qui les ont générés (mis en ligne ou envoyés par mél) n'est que peu souvent<sup>4</sup> envisagée : ainsi, dans le cas particulier des déficients visuels utilisateurs de braille, les documents numériques portant sur une écriture de type scientifique, ne sont pas

2 Comme le propose la CIDIH (Classification Internationale des Déficiences, Incapacités et Handicaps) utilisée par l'Organisation Mondiale de la Santé (OMS).

3 Conception pour tous : l'objectif de cette philosophie est d'aboutir à une production utilisable par un public le plus large possible.

4 Malgré la norme du WAI, les textes législatifs et l'action de l'association Braillet en France.

directement utilisables et nécessitent un traitement d'adaptation en braille.

Les synthèses vocales et les lecteurs d'écran offrent bien entendu une première idée du document, mais elle ne permettent pas toujours une lecture fine et précise, et sont totalement inefficaces pour transcrire entre autres les écritures mathématiques, musicales, et encore moins les graphiques.

### 1.2 Le producteur de braille papier

L'embossage d'un document, c'est-à-dire son impression sous forme de braille papier, nécessite un travail de préparation en amont : mise en page de l'original, transcription en braille à 6 points, encodage, etc. (voir la suite).

L'utilisation de logiciels spécialisés d'adaptation de documents et de production braille augmente là encore le temps de traitement des informations, parfois le coût déjà élevé des équipements, et complexifie la chaîne de transcription.

Voici par exemple une procédure utilisée pour obtenir un embossage correct d'un document word contenant des équations et du texte :

- conversion des équations en MathML avec mathtype ;
- enregistrement du fichier word au format texte ;
- utilisation du logiciel BraMaNet<sup>5</sup> pour transcrire les contenus mathématiques ; enregistrement du résultat ;
- utilisation du logiciel Duxbury pour traiter le texte ;
- embossage.

Là où la production d'un document pour un voyant ne nécessite qu'un éditeur d'équations et une impression à partir d'un fichier unique, celle pour un non-voyant oblige à utiliser quatre logiciels différents et la manipulation de trois fichiers au minimum.

De plus, il est contraint d'utiliser des configurations logicielles imposées et coûteuses :

- licence d'utilisation de mathtype 6 : env. 120€ ;
- licence d'utilisation de microsoft office 2007 : env. 150€ ;
- licence d'utilisation de duxbury 10.6 : 465€ ;
- licence d'utilisation de windows vista : env. 100€.

BraMaNet étant un logiciel libre et gratuit, l'ensemble de l'équipement logiciel revient à plus de 800€.

### 2. Introduction à la problématique du braille

Contrairement aux idées reçues, la transcription en braille n'est pas un simple transcodage de caractères.

Le braille imprimé est composé de cellules de 6 points comme l'illustre la figure 1.

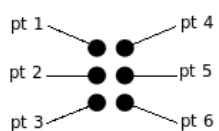


Fig. 1. Une cellule braille, matrice 3x2, avec les noms de chacun des six points (pt 1 à pt 3 pour la première colonne, pt 4 à pt 6 pour la deuxième) et le code braille pts 135 présentant les points 1, 3 et 5 levés pour coder la lettre 'o' en braille.

code braille pts 135 présentant les points 1, 3 et 5 levés pour coder la lettre 'o' en braille.

<sup>5</sup> Logiciel libre développé par la Mission Handicap de l'Université Claude Bernard Lyon 1. <http://handy.univ-lyon1.fr/>

Un document non transcrit est appelé document « en noir » par opposition au document « en braille ».

Le braille informatique (braille 8 points), utilisé avec les plages ou les terminaux tactiles, comporte 2 points supplémentaires (matrice 4x2) ajoutés en bas de chaque colonne (pt 7 et pt 8). Ce braille n'est pas destiné à être imprimé et a pour objectif de lever une partie des ambiguïtés générées par les afficheurs qui ne transcrivent pas et réalisent uniquement un transcodage bijectif.

Par exemple, la lettre A (majuscule) doit normalement être transcrite par deux caractères braille : pts 46 (indicateur de majuscule) et pt 1 (lettre a minuscule). En braille huit points, on va se servir du point 7 pour indiquer la majuscule : pts 17 car il n'est pas possible d'associer plus d'un code braille à un caractère noir.

Si le braille 8 points offre un plus grand nombre de combinaisons possibles ( $2^8$  c'est-à-dire 2 exposant 8, soit 256) que le braille 6 points ( $2^6$  c'est-à-dire 2 exposant 6, soit 64), il est en revanche plus difficile à lire pour certains utilisateurs, les codes brailles nécessitant pour eux un temps d'appréhension plus important au toucher. C'est pourquoi on peut lui préférer le braille 6 points.

Le braille est utilisé essentiellement par des non-voyants ou des transcribers. Un transcriber n'est pas nécessairement un professionnel de la transcription, il peut s'agir par exemple d'un enseignant souhaitant réaliser un document pour un élève déficient visuel en intégration.

Il existe différents types de braille :

- le braille littéraire intégral : destiné à transcrire chaque caractère du document d'origine par un ou plusieurs codes brailles ;
- le braille littéraire abrégé : destiné à réduire le nombre de codes brailles, il décrit des règles de sténographie complexes réparties en deux ensembles : les exceptions et le cas général ;
- le braille mathématique : destiné à retranscrire les expressions scientifiques [2] ;
- le braille musical : destiné à retranscrire la notation musicale.

Chaque langue écrite ne disposant pas du même jeu de caractères (pas d'accents en anglais par exemple), le braille est spécifique à une langue précise [1]. Parfois, il existe même des différences entre normes braille d'une même langue suivant le pays (le braille britannique n'est pas identique au braille américain).

Seule la norme musicale est universelle.

On peut trouver également des normes de transcription pour les textes imprimés, précisant la manière de mettre en page les documents et les structures (tableau, notes, etc.).

Pour finir, chaque langue dispose de tables de codage, qui permettent d'associer un caractère noir à un code braille. Il ne s'agit pas là d'un mécanisme de transcription, mais de représentation, utilisé par les embosseuses, les terminaux brailles ou les fontes de caractères. L'encodage des fichiers est donc un facteur déterminant pour s'assurer de la bonne représentation de la transcription.

Les tables de représentation permettent un affichage correct sur les dispositifs tactiles ou lors de l'embossage, mais dépendent également de l'encodage des fichiers (ISO-8859-1, UTF-8, etc.).

| Nom de la table | Représentation de<br>⠠ (pts 256) | Représentation de<br>⠨ (pt 3) |
|-----------------|----------------------------------|-------------------------------|
| TBFr2007        | . (point)                        | ' (apostrophe)                |
| CBISF           | / (barre oblique)                | . (point)                     |

Fig. 2. Tableau illustrant les différences de normes pour le codage des caractères . et ' avec différentes tables ; il peut donc exister des confusions (cf. représentation de pt 3) suivant les tables utilisées pour la lecture ou la transcription.

### 3. Problématique de la transcription

Les normes sont en général suffisamment détaillées et précises pour réaliser une transcription humaine correcte des contenus.

En revanche, les techniques automatisées rencontrent encore de nombreuses difficultés : passage d'une expression à deux dimensions à un contenu linéaire comme dans le cas des formules mathématiques [2], complexité des règles d'abréviation, ambiguïtés devant prendre en compte le contexte pour être résolues...

Si un transcripateur professionnel est capable d'analyser et de corriger un texte transcrit, les non-professionnels ne disposent pas toujours des connaissances en face de cette complexité.

Pour les cas simples, la lecture d'une transcription comportant des erreurs sera malgré tout réalisable. Lorsque le document comporte en revanche des contenus hétérogènes et plus complexes (images, formules, tableaux), la lecture devient très difficile, voire impossible.

Le passage par un centre de transcription, est à la fois, un gage de qualité grâce à l'expertise du travail, mais aussi source de lenteur par l'aspect communication et délai demandé. Une transcription automatisée offrirait un gain de temps appréciable.

Si la chaîne de transcription est trop longue, ou que les outils ne sont pas accessibles, cela va occasionner une surcharge importante de travail que les personnes déficientes visuelles ne peuvent pas toujours gérer.

En plus d'être performant, un transcripateur automatique doit donc être à la fois simple d'utilisation pour les non-professionnels, accessibles pour les non-voyants, et suffisamment paramétrable pour intéresser les transcripateurs professionnels qui rechercheront avant tout un gain de temps.

### III. CRITIQUE DES SOLUTIONS DE TRANSCRIPTION AUTOMATISÉE EXISTANTES

L'objectif de cet article n'est pas de faire un état de l'art de la transcription automatisée francophone<sup>6</sup>. Nous nous contenterons de proposer une critique générale des solutions actuelles.

Le principal défaut réside souvent dans la complexité de la chaîne de traitement, et dans le décalage entre les temps nécessaires à un voyant et un non-voyant pour l'accès à un contenu numérique de qualité équivalente.

Certaines réalisations offrent des services de bibliothèques spécialisées via le web<sup>7</sup>, mais ne permettent pas une

<sup>6</sup> Lire plutôt à ce sujet [4].

<sup>7</sup> Comme par exemple le serveur Hélène (<http://www.bibliotheque-helene.org/>) de l'association Brailletnet,

transcription en temps réel, et imposent des configurations matérielles précises.

Le passage obligatoire par certaines solutions logicielles et/ou matérielles n'est pas à négliger : l'accès à des transcriptions de qualité se traduit par l'obligation d'utiliser un certain type de matériel ou de logiciel, aux coûts souvent élevés. Une autre forme d'exclusion peut être générée<sup>8</sup>.

Du point de vue des interfaces, les solutions ne sont pas polymorphes dans leurs utilisations : soit via le web, soit logicielle, pas de fonctionnement de type service ou processus. Certaines ne sont pas très simples à utiliser<sup>9</sup>.

Elles se basent la plupart du temps sur des dictionnaires, assistés parfois de règles de transformations, difficiles à maintenir lors des évolutions du braille francophone<sup>10</sup>.

### IV. VERS LA RÉALISATION D'UN TRANSCRIPTEUR « IDÉAL »

Le transcripateur NAT a vu le jour dans le cadre d'un projet technique de Master<sup>11</sup> en 2005. Le développement du logiciel s'est poursuivi pendant plus d'un an sans financement avant de recevoir en Juillet 2007 le soutien du fond social européen par l'intermédiaire du GIP handicap et compétence. Le projet est maintenant suivi par le LIRIS. Un projet de convention partenariale est à l'étude avec l'INS HEA.

L'objectif principal de ce projet est de répondre aux différents problèmes décrits précédemment, en offrant une solution intégrée, d'utilisation polymorphe, accessible à tous, et indépendante de configurations particulières.

La principale motivation n'est pas d'entrer en concurrence avec les centres de transcription, dont les compétences restent de loin supérieures à toute solution automatisée, mais au contraire de leur fournir un outil capable d'augmenter efficacement leur capacité de production, leur offrant la possibilité de répondre plus rapidement aux demandes de leurs utilisateurs.

#### 1. Architecture générale

La prise en compte des contraintes a abouti à une organisation modulaire, basée sur un comportement adaptable à chaque type de document (format, contenus mixtes, encodages, etc.).

L'architecture propose trois modules principaux : la conversion, la transcription et le post-traitement. Dans l'idéal, l'utilisateur transmet au système un fichier dans un certain format ; le module de conversion s'adapte au type de document et réalise un fichier au format interne ; Le module de transcription transcrit ce fichier interne à l'aide de filtres choisis ; le module de post-traitement gère la présentation, l'exportation ou l'impression.

Les rôles spécifiques de chaque module permettent une indépendance dans le processus de développement : si un nouveau format doit être pris en compte, il suffit de lui associer

<sup>8</sup> De nombreux transcripateurs occasionnels et potentiels renoncent souvent à s'équiper et reposent entièrement sur les centres de transcription, dans le cas de la scolarisation intégrée par exemple.

<sup>9</sup> C'est par exemple le cas du logiciel Winbraille de la société Index, <http://www.indexbraille.com>, réputé chez les transcripateurs pour sa configuration et sa prise en main particulièrement complexes.

<sup>10</sup> Le braille francophone est en plein processus d'unification : les évolutions sont fréquentes, la dernière date de 2007 pour les Brailles littéraire et scientifique. <http://www.inja.fr/rubriqueCN.E.A.>

<sup>11</sup> Master handi Paris VIII, <http://ufr6.univ-paris8.fr/desshandi/>

un convertisseur particulier. De même, le choix des filtres de transcription est indépendant du format d'origine.

L'originalité du mécanisme de transcription repose sur l'interopérabilité des différents filtres et de leur spécialisation. Leur implémentation est basée sur l'utilisation de règles et non plus de dictionnaires, afin de s'approcher le plus possible d'un raisonnement humain quant à l'utilisation des différents processus de transcription.

L'interopérabilité de ces filtres permet également la réalisation dynamique d'un scénario de transcription propre à chaque document : utilisation de l'abrégé ou de l'intégral, encodages, tables brailles à utiliser, transcription des mathématiques ou non.

## 2. Choix techniques

Ce type d'organisation se base sur l'utilisation, pour l'opération de transcription, d'un format de fichier interne. Ce format doit fournir un niveau de granularité assez fin pour l'analyse des expressions à transcrire, tout en restant suffisamment souple pour évoluer facilement si on doit lui ajouter de nouveaux types de contenus (musique par exemple).

Les langages de programmation doivent également pouvoir s'exécuter sur différentes plateformes et dans différents modes (via le web, sous forme d'application interfacée ou de services).

Nous avons retenu une solution basée sur java et l'API JAXB<sup>12</sup>, facilitant la manipulation de fichiers XML et de filtres XSL<sup>13</sup> pour la transformation des documents. Le format interne<sup>14</sup> a une base assez simple, permettant l'intégration d'autres standards de notation comme MathML : l'élément le plus fin est le mot ou la ponctuation, les différents types de contenus (mathématiques ou littéraires) sont organisés dans des phrases formant elles-mêmes le document.

Pour l'instant, les propriétés des différents éléments sont représentées à l'aide d'attributs des balises, et non comme balises elles-mêmes (différent en ce sens de HTML par exemple), afin de conserver une structure simple de document.

L'utilisation de formats ouverts et standardisés en entrée (open document, texte simple, MathML) garantit la compatibilité en amont avec les logiciels respectant ces normes internationales, et l'indépendance de Nat vis-à-vis de distributions logicielles particulières (un utilisateur pourra par exemple utiliser open office, staroffice, koffice... afin de réaliser un document au format ODF<sup>15</sup>).

## 3. Implémentation des règles par des feuilles XSL

Chaque type de contenu présente des difficultés différentes de transcription : ainsi, un texte littéraire est composé de

<sup>12</sup> Java Api for XML Processing, <http://java.sun.com/>

<sup>13</sup> eXtensible Markup Language et eXtensible Stylesheet Language

<sup>14</sup> On peut se demander s'il n'aurait pas été plus judicieux d'utiliser un format déjà existant plutôt que d'en spécifier un nouveau : les principales faiblesses des autres formats viennent du niveau de granularité trop grossier, dont l'atome de base est souvent la phrase et non le mot (par exemple, les formats Daisy-dtbook (<http://www.daisy.org/>) ou HTML). De plus, ces formats offrent de nombreuses fonctionnalités non pertinentes pour une transcription qui risqueraient d'en complexifier le processus.

<sup>15</sup> Open Document Format, format certifié ISO 26300

nombreux mots différents, mais conserve une structure assez simple à transcrire ; en revanche, un texte mathématique ne comporte pas autant de variété de mots, mais sa structure est plus complexe.

Les traitements associés seront donc très différents : on trouvera plus de manipulation de chaînes de caractères avec les brailles littéraire ou abrégé et plus de règles nodales avec le braille mathématique ou la musique.

### 3.1. Transcription de texte en braille littéraire

L'algorithme de la feuille de transformation pour le braille littéraire français se compose de différentes fonctions (templates) et se base essentiellement sur la récursivité.

L'objectif est d'aboutir le plus rapidement possible à un état terminal, c'est-à-dire à la transcription du mot.

Les différentes fonctions sont appelées dans un ordre précis au départ, et déterminent l'état dans lequel elles transmettent le mot en cours de transcription à la fonction suivante.

Certaines fonctions ont besoin de déterminer le contexte dans lequel on leur demande de travailler (cas des préfixes majuscules actifs par exemple), et de connaître la nature des éléments précédents et suivants.

L'organisation peut être représentée par le diagramme d'état machine de la figure 3.

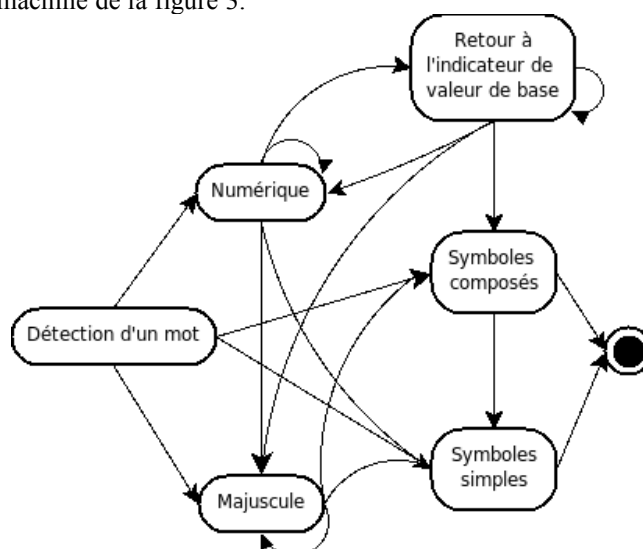


Fig. 3. Diagramme UML d'état machine simplifié de la transcription d'un mot en braille intégral. Cinq fonctions principales sont utilisées : Majuscule, Numérique, Retour à la valeur de base (RVB), Symboles composés et symboles simples. Trois d'entre elles peuvent s'appeler elles-mêmes (Numérique, Majuscule et RVB), deux peuvent amener directement à un état final (Symboles composés et symboles simples).

La transcription du mot « Bonjour » passe ainsi par la fonction Majuscule (à cause du B), puis directement à Symboles simples. Cela donne comme résultat :

```
⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠
```

Le premier code (pts 46) est un indicateur de majuscule, les autres points correspondent à 'bonjour'.

En revanche, le mot « 88Haüy » aura comme parcours : Numérique, RVB (à cause du ü, qui a la même représentation en braille que le chiffre 8), Majuscule, Symboles simples ; on aboutit à la représentation suivante :

```
⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠⠠
```

Le pt 6 indique un mot contenant des chiffres, '88' est transcrit par pts 1256 pts 1256, le code pt 56 indique un retour à la valeur de base (et non plus la valeur numérique), pts 46 indique une majuscule, pts 125 représente le 'h', pt 1 le 'a', pts 1256 le ü et non plus le 8 (retour aux valeurs de base) et pts 13456 le 'y'.

Une telle organisation possède deux grands avantages : d'une part elle autorise la combinaison de règles de manières dynamique (les fonctions sont susceptibles d'être appelées à n'importe quel moment du mécanisme de transcription, et non pas à un moment précis); d'autre part, elle facilite la rapidité de traitement des différents mots, en appelant uniquement les fonctions nécessaires à la transcription de telle ou telle chaîne.

Les fichiers de règles « dictionnaires » du logiciel winbraille ont ainsi ce défaut de ne pouvoir chaîner un ensemble d'opérations, bien qu'elles prévoient l'activation de différents contextes de lecture (présence de majuscules, de contenu numérique, etc.). Cependant, les enchaînements de règles doivent être prévus à l'avance, et générèrent à chaque fois de nouvelles règles redondantes.

Si le nombre de règles à traiter en parallèle augmente, le nombre de combinaisons à représenter peut devenir exponentiel.

```

- Number prefix rules
-----
<digit, maintain_NP;          ; digit ; ""      >
<math_debut                   ;          ; digit ; ""      >
<digit, letters                ;          ; math_debut digit; "" >
<digit, maintain_NP, letters;  ; math_debut digit; "" >
<                               ;          ; math_debut digit; NP_Braille >
<                               ;          ; digit; NP_Braille; "" >
[etc...]
- Capital prefix rules
<(CAPITALPREFIX=CAPITALPREFIX) upperletter
; ; upperletters, lowerletter ;
<(CAPITALPREFIX=CAPITALPREFIX)
[etc...]
-- Symbol and character rules
[etc...]
< ; "ü" ; ; p1256 >

```

Fig. 4. Règles de transformation des majuscules et du caractère 'ü' sous winbraille : les règles sont ordonnées et ne permettent pas le retour arrière. Dans l'exemple, le ü ne pourra plus être transcrit correctement car le contexte « numérique » est perdu ; on lira donc 88ha8y.

### 3.2. La feuille de transformation des mathématiques

Le principe de fonctionnement est cette fois-ci différent : le vocabulaire mathématique est plus simple à gérer que les écritures littéraires, mais les structures sont plus complexes.

Le format MathML, recommandé par le w3c, a la particularité de représenter assez fidèlement ces structures et d'identifier chaque élément de vocabulaire par des tags spécifiques suivant le rôle de leur contenu : mo (math operator), mi (math identifier), mn (math number), mtext (math text) entre autres. La discrimination des différents symboles est donc facilitée.

MathML offre deux possibilités de codage : le content MathML, destiné à représenter la structure logique d'une expression, et le presentation MathML utilisé pour la représentation graphique d'une expression mathématique.

Le premier format faciliterait grandement le processus de transcription, mais il n'est malheureusement pas utilisé par les logiciels grand public dont l'objectif est surtout de soigner le rendu graphique d'un contenu.

Comme il est également plus facile de transformer du content MathML en presentation MathML que l'inverse, nous avons donc choisi d'utiliser ce dernier dans le format interne de Nat pour la représentation des mathématiques.

Plutôt que d'utiliser une grande quantité de dictionnaires de traduction de différents formats d'entrée (LaTeX, MathML, Notation polonaise, etc.), et un transcripateur pour chacun de ces formats comme c'est le cas pour le transcripateur mathématique du projet Vickie<sup>16</sup> [3], NAT s'appuie sur un premier module de conversion en MathML et intègre ensuite ce format tel quel au format interne.

Comme pour l'intégral, il est alors possible de réaliser des feuilles de transformation XSL capables de s'adapter au contexte lors de la transcription d'une expression.

NAT reprend et améliore légèrement les techniques de transcription du MathML via XSL utilisées avec le logiciel BraMaNet<sup>17</sup>. Les améliorations portent sur l'implémentation de nouvelles entités MathML et sur l'organisation générale de la feuille.

Le projet Vickie utilise lui en fin de traitement (reading module) un dictionnaire de termes associant une représentation unique pour chaque élément; ce système est relativement performant, mais ne permet pas encore une fois de s'adapter au contexte.

Par exemple, un contenu mathématique est normalement précédé de l'indicateur de changement de code : ::X:: (pt6 pt3). Cependant, lorsqu'un contenu est suffisamment simple (i.e. qu'il n'est pas nécessaire d'utiliser des symboles de blocs), il suffit d'activer le contexte mathématique en utilisant uniquement le modificateur mathématique : ::X:: (pt6).

Ainsi, la fraction un demi doit être précédée de pt3 dans le cas [1] et de pt6 pt3 dans le cas [2] car x + 1 doit être placé entre blocs brailles.

$$[1] \frac{1}{2} + 3 \quad [2] \frac{1}{2} * \frac{3}{x+1}$$

Il n'est donc pas possible d'associer une valeur unique en braille au tag racine d'une équation.

Il existe des cas plus complexes où une entité peut, suivant le contexte, avoir différentes significations : par exemple, l'entité « &rarr; » (flèche droite) peut représenter une flèche simple ou un vecteur.

Il est certain que le content MathML serait idéal pour lever facilement ce type d'ambiguïtés, mais il reste encore minoritairement implémenté.

Le code XSL de la figure 5 illustre comment NAT peut différencier dans ce cas ces deux possibilités.

Les travaux actuels devraient permettre d'ici peu la prise en compte directe du content MathML, sans passer par une phase de conversion en presentation MathML.

16 démonstration en ligne : <http://inova.snv.jussieu.fr/math/>  
 17 Frédéric Schwebel, Vivien Guillet et Bruno Mascaret, mission handicap de Lyon 1, <http://handy.univ-lyon1.fr/MH/bramanet/>

```

<xsl:when test=".='&rarr;' or .='\xrightarrow'">
  <xsl:choose>
    <xsl:when test="local-name(parent:*)='mover'">
      <xsl:text>&pt46;&pt25;</xsl:text><!--vecteur-->
    </xsl:when>
    <xsl:otherwise>
      <xsl:text>&pt456;&pt156;</xsl:text><!--fleche-->
    </xsl:otherwise>
  </xsl:choose>
</xsl:when>

```

Fig. 5. Fragment de la feuille XSL de NAT où se fait la distinction entre la flèche droite et le vecteur, représentés de la même manière en MathML 1.

## V. LIMITES ET PERSPECTIVES

La dernière version de NAT gère exhaustivement le Braille intégral et mathématique. La récente mise en place d'un consortium comprenant experts, développeurs et testeurs devrait permettre d'améliorer la qualité du logiciel et d'inclure à court terme les fonctionnalités de transcription en abrégé et en braille musical.

Mais peut-on pour autant parler de transcripateur parfait?

Certains cas restent problématiques et sujets à ambiguïté; si l'intégral reste assez simple à appréhender, l'abrégé en revanche va poser de nombreux problèmes que seule une intelligence humaine serait à l'heure actuelle capable de résoudre sans se tromper.

Nous proposons de terminer cet article par une mise en lumière des principaux facteurs limitant la qualité d'une transcription automatisée, et en envisageant de nouvelles utilisations de cet outil informatique dans le traitement de l'information adaptée.

### 1. Les limites d'un transcripateur automatique « idéal »

Un des aspects particulièrement négligé dans l'accessibilité des contenus numériques est la prise en compte du temps qui sera nécessaire pour atteindre puis prendre connaissance de l'information [7][8]. Si les sommaires, les liens internes et autres outils de navigation cherchent à limiter ce problème, ils ne permettent toujours pas à un non-voyant d'interagir de la même manière qu'un voyant avec un document numérique : l'accès à l'information reste séquentiel et occasionne une perte de temps conséquente.

L'utilisation de sortie papier permet de s'affranchir de la représentation linéaire du braille informatique, et offre des possibilités de spatialisation de l'information plus avancées : représentation de matrice par exemple, de tableaux, mise en page ergonomique, etc. Cependant, on perd également de nombreuses possibilités d'interaction avec le document (corrections, recherches, annotations, etc.).

De nombreux contenus restent également quasiment impossibles à transcrire par un logiciel (images, illustrations, graphiques, etc.).

Notre idée n'est donc pas de prétendre réaliser un transcripateur parfait, mais plus un outil adaptable, facilitant le mécanisme de transcription et suffisamment souple pour offrir à une large gamme d'utilisateurs des fonctionnalités qui à défaut d'être parfaites leur feront gagner du temps.

Ainsi, NAT peut servir de base au travail des transcripateurs professionnels, tout en permettant à un non-brailleiste de produire des documents utilisables par des non-voyants.

Les transcriptions produites par NAT doivent donc être vues comme le premier pas vers l'accessibilité d'un document, et non comme une finalité.

### 5.2. Perspectives d'amélioration et de fonctionnement

Afin d'utiliser pleinement le potentiel de NAT, plusieurs pistes de recherches sont en cours d'exploration.

NAT pouvant facilement s'intégrer dans d'autres logiciels de part son aspect polymorphe, nous sommes en train d'exploiter plusieurs pistes d'intégration du logiciel pour en faciliter l'usage : plugin pour openoffice, fonctionnement via internet, chaînage avec d'autres outils dans un processus complexe de transcription en ligne (projet BraillePost<sup>18</sup>).

L'intégration de NAT en tant que service de transcription à un niveau proche du système d'exploitation pourrait améliorer de manière significative l'utilisation actuelle que les non-voyants ont de leurs logiciels d'accessibilité aux interfaces graphiques.

Nous travaillons également à la mise au point de facilitateurs d'interactions, afin de rendre réellement accessible un document numérique à un non-voyant. Ces recherches réalisées au LIRIS dans l'équipe SILEX devraient également permettre d'élargir notre champ d'action à l'ensemble des déficients visuels.

Il reste également à développer NAT dans d'autres langues que le français : une communauté se constitue actuellement autour du projet et devrait donner le jour à de nouvelles fonctionnalités de transcription prochainement.

## BIBLIOGRAPHIE

- [1] Mackenzie sir C.N., « L'écriture braille dans le monde : rapport sur le progrès accompli dans l'unification des écritures brailles », *International meeting on Braille uniformity*, UNESCO, Paris, 1954.
- [2] Magna F., « Le braille mathématique », *Revue bimestrielle du Groupement des professeurs et éducateurs d'aveugles et amblyopes*, n°183, 2002.
- [3] Moço V., Archambault D., « Automatic translator for Mathematical Braille », *2nd International Conference on Universal Access in Human-Computer Interaction*, Heraklion, Greece, Juin 2003
- [4] Ollier M., « Aides techniques, nouvelles technologies et scolarisation d'élèves déficients de la vue », *La nouvelle revue de l'adaptation et de la scolarisation*, hors série n° 3, « Scolariser les élèves déficients visuels aujourd'hui », 2007.
- [5] Sperandio J.-C., Uzan G., Oltra R., « L'informatique comme barrière d'exclusion ou comme aide technique à l'intégration », *Performances Humaines et Techniques*, n° hors série : « Situation de Handicap », p. 34-40, juin 1999.
- [6] Sperandio J.-C., Uzan G., « Ergonomie des aides techniques informatiques pour personnes handicapées. », *Handicap, revue de sciences humaines et sociales*, mai 2002.
- [7] Uzan G., Temps technologiques, temps individuels, temps sociaux : l'articulation des contraintes temporelles dans l'utilisation de l'informatique par des aveugles, *Colloque Fisaf*, Paris, juin 2003.
- [8] Uzan G., Ergonomie cognitive du handicap visuel : une contribution à la conception d'aides informatiques, Thèse de doctorat, Université René Descartes Paris 5, 9 décembre 2005.

18 <http://www.braillepost.be> : bureau de poste virtuel