Abdel-Malek Boualem, Stéphane Harié et Jean Véronis

Laboratoire Parole et Langage
Université de Provence & CNRS
29, avenue Robert Schuman, 13621 Aix-en-Provence Cedex 1, France
e-mail: *multext@univ-aix.fr*

Mots-clés. Texte multilingue, caractère, saisie, norme de codage, édition, échange de données textuelles.

Résumé. Cet article présente un éditeur de textes multilingues, **MtScript**, qui permet d'utiliser de nombreux types d'écritures dans un même document, y compris des écritures en sens opposés. De plus, **MtScript** permet d'associer à chaque langue des règles de saisie au clavier. **MtScript** a été développé dans un environnement portable (UNIX, X-Window, C, Tcl/Tk) et une première version peut être téléchargée sur le WWW.

Abstract. This paper describes the multilingual text editor **MtScript**, which enables the use of several different writing systems in the same document, including both right-to-left and left-to-right systems. In addition, customized input rules can be associated with each language in the text. **MtScript** is based on a portable environment (UNIX, X-Window, C, Tcl/Tk) and a first version can be downloaded from the WWW.

1. Introduction

Dans un précédent article [BOUA95a], nous présentions les difficultés de conception et de réalisation d'outils pour l'édition et le traitement de textes multilingues. Nous mentionnions que si des solutions commençaient à se mettre en place pour des langues européennes, la conception d'outils pour d'autres familles de langues était encore à un stade peu avancé. Nous présentions le prototype d'un éditeur multilingue sur lequel nous avions précédemment travaillé [BOUA90] et que nous avions intégré à un environnement de traduction automatique du français vers l'arabe [BOUA93]. Cependant, cet éditeur présentait des faiblesses au niveau du codage des caractères et des documents, de l'incompatibilité des formats d'échanges des données textuelles et au niveau de l'environnement logiciel non portable.

Cet article présente l'éditeur de textes multilingues **MtScript** développé dans le contexte du projet MULTEXT [MUL96]. **MtScript** permet de mélanger de nombreux types d'écritures dans un même document: latin, arabe, cyrillique, grec, hébreu, chinois, japonais, coréen, etc. (voir figure 1). Les fonctions d'édition de **MtScript** permettent d'insérer ou de supprimer des zones de texte même en écritures de sens opposés. De plus, **MtScript** permet d'identifier les langues utilisées dans un texte multilingue, de leur associer des règles de saisie au clavier et de traiter différents types de codage des caractères (sur un ou plusieurs octets). Enfin, **MtScript** a été développé dans un environnement portable (UNIX, X-Window, C, Tcl/Tk) et est basé sur les normes internationales de codage. La version 1.1 de **MtScript** a été mise en libre accès sur le WWW, à l'URL:

`http://www.lpl.univ-aix.fr/projects/multext/MtScript/`

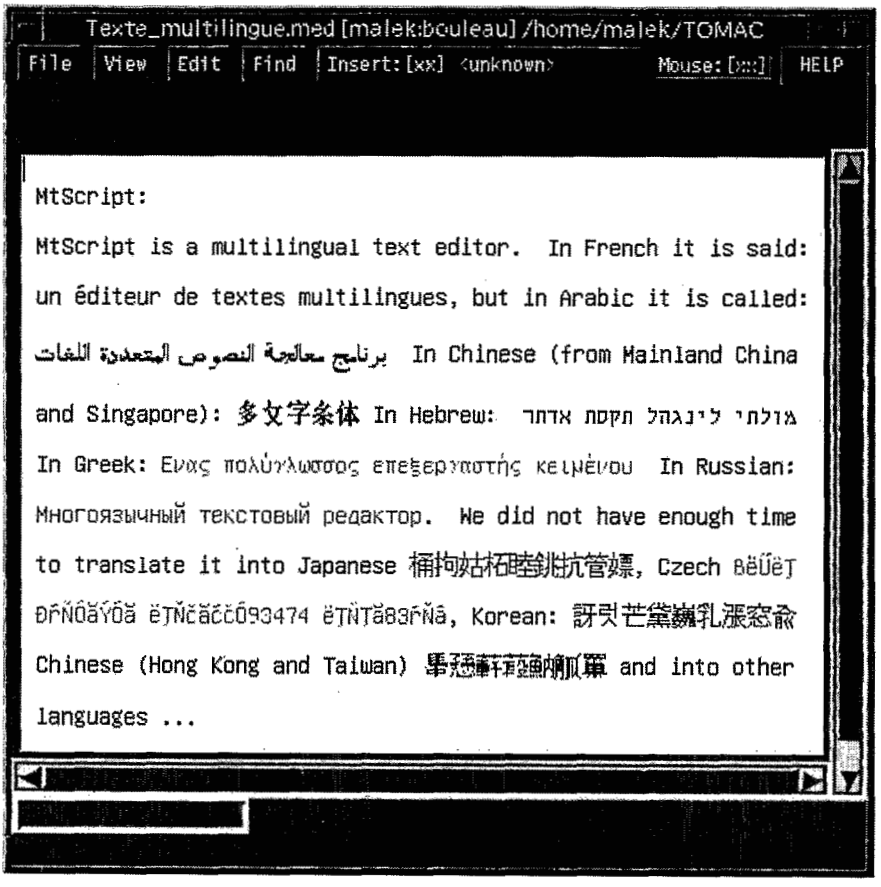

Figure 1. Un écran de MtScript

2. Difficultés de l'édition de textes multilingues

Il existe un nombre de plus en plus important d'outils nécessitant des modules d'édition de textes multilingues tels que le traitement de texte, les bases de données et la micro-édition. Dans le domaine de la traduction automatique ou assistée par ordinateur, un éditeur de textes multilingues est un outil fondamental pour les phases de pré-édition des textes sources et de post-édition des textes cibles [BENT91]. D'autre part, les éditeurs de textes multilingues sont d'une grande utilité dans le domaine de l'internationalisation et de la localisation des logiciels et de leur documentation.

Toutefois, le traitement d'autres langues non basées sur l'écriture latine pose un certain nombre de difficultés. Par exemple, l'arabe s'écrit de la droite vers la gauche; le chinois contient des milliers d'idéogrammes qui rend impossible leur représentation sur un seul octet; en thai et dans certaines langues indiennes, la succession des caractères dans le mot ne correspond pas à leur succession phonétique, un caractère peut même en entourer d'autres; en coréen, les caractères s'agglutinent en syllabes. Les difficultés de réalisation d'éditeurs de textes multilingues se situent à différents niveaux: saisie, codage, édition, impression et échange de données.

2.1. Saisie

Si beaucoup de claviers ne représentent que les caractères graphiques de l'ASCII (ou ISO 646), certains claviers localisés (ou adaptés) comportent des touches pour des caractères accentués ou des caractères spéciaux. Par exemple, les claviers français comportent généralement les touches correspondant aux caractères "à ç é è ù", mais les caractères comportant un accent circonflexe ou un tréma sont saisis au moyen de deux frappes successives et il n'y a généralement pas de touche unique permettant de réaliser sur un clavier français des caractères existants dans d'autres langues européennes, tels que "ñ" ou "ò". Dans un contexte fortement multilingue, on ne peut d'ailleurs guère imaginer un clavier qui contienne tous les caractères possibles. L'adjonction de langues comme le chinois (plus de 6000 idéogrammes) ou l'arabe (environ 4 fois 28 lettres + 10 voyelles) rend nécessaire la mise au point de règles et de programmes spécifiques de saisie.

Les solutions proposées par les constructeurs d'ordinateurs sont souvent hétérogènes. Il existe en théorie une norme de saisie pour les claviers à 48 touches (ISO/IEC 9995-3) au moins pour l'écriture latine, mais elle n'est guère respecté. Un ensemble de méthode de saisie des caractères du répertoire de l'ISO 10646 a été récemment proposé [LABO95]: entrée par code hexadécimal, par composition, etc. Toutefois, ces méthodes imposent une mémorisation quasi-impossible des codes par l'utilisateur et/ou un nombre important de frappes pour un même caractère. Il est donc nécessaire de développer des méthodes de saisie plus intuitives et minimisant si possible le nombre de frappes par l'utilisateur.

2.2. Codage

2.2.1. Caractères

Les constructeurs d'ordinateurs et les concepteurs de logiciels utilisent de nombreux codes de caractères spécifiques et non compatibles (*MS-Windows character set for Western Europe MS CP1252, Dec Multinational Character Set, International IBM PC character set IBM CP850, Macintosh Extended Roman character set, Hewlett-Packard ROMAN8,* etc.). Cependant, des versions successives de normes de codage de caractères ont été élaborées au niveau international. Elles sont déjà utilisées sur certaines plateformes. En particulier, la série de normes ISO 8859 propose des jeux de caractères pour les alphabets latin (6 variantes adaptées à des régions différentes), cyrillique, arabe, grec et hébreu. Plus récemment (1993), la norme ISO 10646 (*Universal multiple-octet coded character set* ou UCS) a proposé un jeu de caractères "universel" regroupant tous les jeux de caractères de l'ISO 8859, ainsi que le chinois, le coréen, le japonais, l'alphabet phonétique international (API), etc. Dans sa forme présente (ISO 10646-1), l'UCS utilise un codage sur 16 bits, qui correspond en réalité à la norme UNICODE et sera étendu à 32 bits dans les additions futures, ce qui permet un codage quasi illimité de caractères [JAMG95]. Toutefois, les environnements ne sont pas encore prêts à l'implémentation de jeux de caractères sur plusieurs octets, bien que la situation évolue rapidement (WINDOWS-NT, AT&T Bell Plan 9 et Apple QuickDraw GX).

2.2.2. Systèmes d'écriture

Dans un texte multilingue il convient de coder non seulement les caractères individuels, mais aussi les systèmes d'écriture (notion plus générale que celle des langues ou des scripts). Dans le cas d'un codage sur un octet (par exemple la série ISO 8859-1), il faut marquer le passage d'un jeu à l'autre (par exemple, passage de grec à cyrillique). Ce codage peut être fait à l'aide d'un codage tel que celui proposé par la norme ISO 2022 qui fournit des séquences d'échappement <SI> (shift-in) et <SO> (shift-out) qui permettent le passage entre "jeu principal" et "jeu complémentaire". Ces mécanismes sont toutefois limités et des difficultés se posent, en particulier lors du mixage dans un même document de caractères codés sur un octet (par exemple les jeux ISO 8859) ou sur deux (par exemple gb2312-80 ou big5-0 pour le Chinois, jisx0208-1983-0 pour le Japonais ou ksc5601-1987-0 pour le Coréen).

Unifiant tous ces jeux de caractères en un jeu unique, l'UCS résoud une partie du problème, puisqu'il n'est plus nécessaire d'implémenter des mécanismes de changement de jeux. Toutefois, le problème n'est pas totalement résolu car l'UCS ne fournit pas de codage explicite des systèmes d'écriture, qui est nécessaire en particulier pour déterminer la direction d'écriture.

2.2.3. Langues

Les traitements linguistiques d'un texte multilingue (segmentation, analyse morpho-lexicale, etc.) nécessitent l'identification des langues utilisées. La connaissance du jeu de caractères ou du système d'écriture ne suffisent pas à identifier la langue dans laquelle est écrite une portion de texte: un document codé en ISO 8859-1, par exemple, peut aussi bien être écrit en français, en anglais, en espagnol ou même dans une combinaison de ces langues.

Il existe des normes de codification des noms des langues:

- ISO 639-1988: code à 2 lettres alphabétiques pour 140 langues (par exemple, "en" pour English, "fr" pour French, etc.),
- ISO 639-2: code à 3 lettres alphabétiques, alpha-3, en cours de développement ("eng" pour English, "fre" ou "fra" pour French, etc.).

Toutefois, dans le codage interne d'un document, ces codes ne peuvent pas être utilisés tels quels et il n'y a pas à l'heure actuelle de norme établie pour des séquences d'échappement qui permettraient de représenter le passage d'une langue à l'autre, bien qu'une proposition existe consistant à utiliser des codes de séquences de contrôle ISO/IEC 6429 avec une conversion numérique des codes alphabétiques ci-dessus [LANG93]. Le marquage des langues est également en cours de définition dans le langage HTML utilisé sur le *World Wide Web* [YERG95].

2.3. Édition

La plupart des langues s'écrivent horizontalement de la gauche vers la droite. Certaines langues comme l'arabe ou l'hébreu s'écrivent de la droite vers la gauche. D'autres langues comme le chinois ou le japonais peuvent même s'écrire du haut vers le bas (textes anciens). La cohabitation de langues à écritures opposées dans un même document et particulièrement dans une même ligne de texte pose des difficultés lors de l'insertion ou la suppresssion de zones de texte. À l'extrême, l'exemple évoqué par J.Becker (figure 2) montre que le ré-arrangement des mots est nécessaire pour maintenir la cohérence sémantique de la phrase. Cet aspect représente l'une des grandes difficultés dans la conception d'éditeurs de textes multilingues.

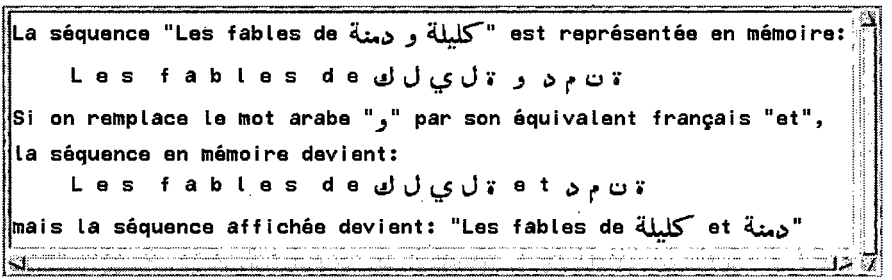

Figure 2. Edition dans un texte mixte

2.4. Impression

Les nouvelles techniques d'impression incluant des représentations Bitmap ou PostScript ne sont pas tout à fait généralisées aux caractères non latins. Toutefois, des travaux sont menés dans ce domaine comme ceux de C.Bigelow et K.Holmes [CBKH95] dans la création d'une police de caractères *Unicode Lucida Sans* pour visualiser et imprimer des documents électroniques multilingues. La production de polices de caractères (PostScript, etc.) pour les nouvelles normes de codage permettrait sans doute une impression multilingue de qualité.

2.5. Échange d'informations

Avec l'explosion d'Internet, l'échange des documents multilingues sous un format électronique est de plus en plus fréquent. Jusqu'à une date récente, seuls les caractères graphiques invariants de

l'ISO 646-IRV (ASCII) permettaient de véhiculer et restituer des textes électroniques "sans corruption" et il était donc nécessaire d'utiliser des mécanismes de recodage tels qu'*uuencode* ou *binhex*. La situation est cependant en voie d'amélioration: des normes ont été adoptées pour Internet qui permettent de transporter les 8 bits des caractères de manière "clean" dans le protocole TCP/IP (par exemple des applications comme *telnet* et *ftp* sont "8-bit clean"). De plus l'extension MIME (*Multi-purpose Internet Mail Extensions*: RFC-1521 and RFC-1522) permet l'échange de données proprement en toutes circonstances par compactage et décompactage appropriés. Toutefois, cette "norme" n'est pas encore implémentée de façon universelle et quelques problèmes de transmission subsistent. Enfin, notons que la garantie de l'échange des octets sans corruption (sans perte du 8ème bit en particulier) n'est pas suffisante pour le transfert de données multilingues; il est nécessaire que les deux acteurs de la transaction, l'expéditeur et le destinataire, aient les mêmes mécanismes de codage des caractères, des systèmes d'écriture et des langues.

3. Les outils existants

Les travaux pour l'élaboration d'outils d'édition de textes multilingues ont souvent été menés sous forme d'études expérimentales isolées aboutissant à des produits parfois incompatibles, difficiles à exploiter et non conformes aux normes de codage. En outre, les solutions proposées ne concernent, dans la plupart des cas, que les langues à alphabet latin et ne peuvent être adaptés à d'autres familles de langues.

Les premières approches dans la conception d'éditeurs de textes multilingues furent proposées par Xerox (1ère et de 2ème génération d'outils "Star" [BECK84] et "ViewPoint Documenter" [BECK87]). D'autres outils se sont spécialisés dans des domaines particuliers comme la traduction assistée par ordinateur, par exemple les logiciels "TED" de Ink-Languages, "IDOS-A/II" de la société Integro et "TSS" de la société Alps. Ces logiciels ont étré conçus de manière limitée à quelques langues européennes.

Parmi les outils récents de traitement de textes multilingues, nous trouvons le logiciel "Universal Word" développé par Wysiwyg Corporation [UNIW96] qui intégre un grand nombre de langues et le logiciel "WinText" développé sous l'environnement Apple Macintosh par la société WinSoft. Celui-ci permet la fusion de plusieurs langues dans le même document même en écritures opposées. Dans l'environnement PC-Windows, nous trouvons le logiciel de traitement de texte "Word" de Microsoft utilisant les interfaces multilingues TwinLink et TwinBridge. Enfin, dans le monde des stations de travail, l'environnement TEX et son extension multilingue (ArabTeX, etc.) est quasi présent dans le milieu scientifique. Un grand nombre de polices de caractères au format TEX dans tous les systèmes d'écriture ont été conçues. L'inconvénient de cet environnement est qu'il n'est pas WYSIWYG. Les textes sources sont codés à l'aide de balises de structure, ils doivent être compilés pour générer la version finale visualisable au format PostScript.

Parmi les projets à vocation universelle, OMEGA comprend un certain nombre d'extensions de TEX qui améliorent ses possibilités de traitement multilingue [YHJP95]. Il utilise la norme ISO 10646/UNICODE comme base de codage des caractères, mais accepte d'autres types de codage car il inclut un mécanisme de conversion de textes vers cette norme. Des algorithmes puissants permettent d'interpréter la composition ou la translitération des caractères non latins (interface-utilisateur), à manipuler des codes de caractères différents (échange d'information) et à générer les variantes graphiques correctes telles que les ligatures ou les forms contextuelles des caractères (typographie).

D'autres travaux plus récents dans le domaine du multilinguisme sont également en pleine effervescence. Nous citerons les activités du laboratoire CRL [CRL96] pour le développement d'outils dans une variété de domaines (traduction multilingue, extraction de textes, dictionnaires multilingues, etc.), les activités de la société Accent [ACC96] pour le développement de navigateurs WWW multilingues et enfin les activités de la société LangBox International [LANG96] dans l'internationalisation et la localisation du système Unix et de ses applications.

Contrairement à la plupart des travaux existants dans le domaine des éditeurs multilingues, **MtScript** a le mérite, d'une part d'être distribué gratuitement, et d'autre part d'avoir été développé dans un environnement (Unix, X, C, Tcl/Tk, WorkStation) paramétrable, évolutif et portable (PC/Windows et Macintosh).

4. Description de l'éditeur MtScript

4.1. Caractéristiques

L'éditeur **MtScript** a été développé dans l'environnement UNIX, X, C, Tcl/Tk, qui présente les avantages suivants:

- langage de script (les commandes sont interprétées interactivement),
- manipulation de données textuelles (caractères, fontes, mots, etc.),
- possibilité d'affecter des "attributs" aux caractères,
- possibilité de gérer des évenements de X-Window (souris, clavier, etc.),
- contrôle des bitmaps,
- convivialité de l'interface (X-Window, boutons, widgets, etc.)
- bonne portabilité sur plusieurs autres environnements (Windows, etc.).

MtScript est un éditeur de texte incluant toutes les caractéristiques d'un éditeur monolingue standard ainsi que des caractéristiques multilingues:

- mixage d'écritures en sens opposés sur la même ligne de texte,
- marquage de la langue d'une portion donnée de texte,
- insertion et suppression de caractères dans les deux sens d'écriture,
- fonctions d'édition de texte (copier / couper / coller),
- etc.

MtScript est indépendant des langues. Les langues traitées sont des paramètres externes représentés par des fichiers de règles d'écriture et de règles de translitération et des polices de caractères. L'adaptation de l'éditeur à une nouvelle langue nécessite "simplement" de fournir les polices de caractères adéquates ainsi que les règles d'écriture et de translitération.

4.2. Représentation interne

Dans sa version actuelle, **MtScript** utilise les jeux de caractères suivants:

- iso8859-1, 2, 3 et 4 (Alphabets latins)
- iso8859-5 (Cyrillique)
- iso8859-6 (Arabe)
- iso8859-7 (Grec)
- iso8859-8 (Hebreu)
- gb2312-80 et big5-0 (Chinois)
- jisx0208-1983-0 (Japonais)
- ksc5601-1987-0 (Coréen)

Dans les prochaines versions, nous envisageons d'adopter l'UCS (ISO 10646) qui inclut d'autres systèmes d'écriture et un grand nombre de caractères absents dans les autres normes (par exemple, les ligatures typographiques, voire linguistique, œ, Œ, qui sont considérées en Français comme étant des éléments textuels distincts des lettres qui les composent).

L'étiquetage des jeux de caractères et des langues est effectué à l'aide d'un fichier de style associé à chaque texte multilingue et contenant une instanciation des attributs des caractères. Ces attributs décrivent pour chaque portion du texte la langue, la police de caractères, le jeu de caractère, le style, la taille, la couleur, les tabulations, etc. Ces attributs sont associés à une position dans le texte exprimée par des numéros de lignes et de caractères. La figure 3 donne une partie de la "feuille de style" associée au texte de la figure 1. Nous développons actuellement un format d'échange HTML qui utilisera la balise <LANG> proposée par la norme HTML 3.0.

4.3. Saisie

MtScript utilise des modules de saisie basés sur les caractères qui se retrouvent sur la quasi-totalité des claviers, à savoir ceux de l'ISO 646-IRV. Les modules de saisie sont de 2 types:

- **module de saisie alphabétique** pour les langues alphabétiques (français, arabe, russe, etc.),
- **module de saisie phonétique** pour les langues à idéogrammes (chinois, etc.). Ce deuxième module est en cours de développement.

```
{mtscript_version 1.1}
{default_style
  { -width 80}
  { -height 40}
  { -tabs {52.0 104.0 156.0 208.0 260.0 312.0 364.0 416.0 468.0 520.0 572.0 624.0 676.0
728.0 780.0}}
  { -wrap char}}
{xx
  {-foreground IndianRed3 -font iso_8859_1}
  {23.0 23.1}}
{ar
  {-foreground PaleGreen4 -font arabic}
  {8.1 8.40}}
{bg
  {-foreground DeepPink1 -font iso_8859_5}
  {}}
{zh_CN
  {-foreground brown -font gb2312_1980}
  {10.17 10.27 10.39 10.40}}
{zh_TW
  {-foreground MediumPurple4 -font big5_0}
  {20.32 20.46}}
{cs
  {-foreground DarkGoldenrod -font iso_8859_2}
  {16.59 18.34}}
{nl
  {-foreground black -font iso_8859_1}
  {}}
{en
  {-foreground black -font iso_8859_1}
  {1.1 4.41 6.35 8.1 8.40 10.17 10.27 10.39 10.40 10.41 10.64 12.11 12.51 14.1 14.32 16.31
16.51 16.59 18.34 18.44 18.69 20.32 20.46 23.0}}

...
```

Figure 3. Représentation interne des documents (feuille de style)

4.3.1. Saisie alphabétique

Le module de saisie alphabétique est un programme unique pour toutes les langues alphabétiques. Il utilise un fichier de règles d'écriture et un fichier de règles de translitération par langue.

Chaque fichier de règles d'écriture contient:

- une **classification des caractères en** fonction de leur comportement, par exemple:
  - minuscules accentuables,
  - minuscules non accentuables
  - majuscules accentuables
  - caractères invariants utilisés pour la saisie des accents,
  - chiffres,
  - etc.
- **des règles d'écriture** spécifiques à la langue, par exemple:
  - français: **e + ' => é ; c +, => ç** ; etc.
  - grec: sigma => σ (début et milieu de mot) ou ς (fin de mot)
  - allemand: **s + s => ß**
  - etc.

Les règles d'écriture des différentes langues sont exprimées dans un formalisme intuitif basé sur un mécanisme d'automates à états finis. Les fichiers de règles sont ensuite compilés et convertis sous forme de tables exploitables par les programmes de saisie.

Des règles sont fournies par défaut pour chaque langue, mais elles peuvent être redéfinies à volonté par l'utilisateur, en fonction d'habitudes ou de besoins spécifiques, ou de particularités de certains claviers. Les règles par défaut sont basées sur les principes suivants:

- Les **caractères accentués** sont saisis par deux caractères, selon les règles déclarées pour la langue. Ainsi, dans les règles du français, e + ' donne é, mais la même combinaison donne e' en anglais. La génération de e' en français (qui est une séquence beaucoup moins fréquente que é), passe par une séquence d'échappement: e + **ESC** + ' => e'.
- Les **caractères non latins** sont saisis en respectant le plus possible les habitudes et conventions de langues concernées et des normes de translittération quand elles existent. Les tables de translitération sont représentées dans un fichier de règles de translitération assicié à chaque langue (par exemple ISO 233-1984/1993 pour l'arabe, ISO 259-1984 pour l'hébreu, ISO/R 843-1968 pour le grec, etc.). Ainsi, on tapera a pour α, b pour β, s pour س, etc.
- Les **formes variantes** telles que les deux sigmas du grec (σ en début et milieu de mot ou ς en fin de mot), le double s allemand (β) ou les variantes positionnelles des lettres arabes sont générées de façon dynamique en fonction du contexte sans que l'utilisateur ait à intervenir. Le cas de la langue arabe est particulièrement intéressant [BOUA95b]: l'alphabet contient 28 lettres, dont la plupart s'écrivent sous 4 formes différentes selon leurs position dans le mot (figure 4). **MtScript** gère totalement l'affichage des variantes positionnelles, y compris lors des éditions (insertions, suppressions, etc.).

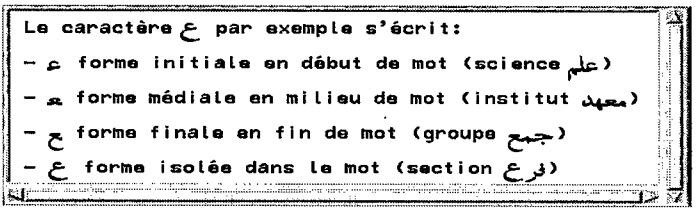

Figure 4. Variantes positionnelles de l'arabe.

4.3.2. Saisie phonétique

Certaines langues, comme le chinois, sont basées sur un nombre important d'idéogrammes. Le chinois traditionnel, par exemple, inclut plus de 60000 idéogrammes représentant chacun un concept particulier. Dans les années 80, des versions d'idéogrammes simplifiées et différentes ont été adoptées par la Chine Populaire d'une part, et par Taiwan et Hong Kong d'autre part. Diverses méthodes de saisie existent, telles que la saisie par le code du caractère à deux octets (par exemple, code GB-2312-80) ou bien la saisie Pinyin consistant en un codage phonétique des idéogrammes en caractères latins (420 syllabes accompagnées de tons différents, jusqu'à 5 tons par syllabe). **MtScript** intégrera dans sa prochaine version un module de saisie des idéogrammes chinois basé sur leur transcription phonétique en Pinyin.

4.4. Affichage et restitution

Comme il a été dit plus haut, le problème fondamental de l'affichage de textes multilingues est la co-existence d'écritures en sens opposés sur la même ligne de texte. Les fonctions d'insertion et de suppression de caractères doivent tenir compte de leur sens d'écriture, selon des règles parfois complexes. Dans **MtScript**, l'utilisateur choisit le sens d'écriture principal pour une zone de texte, gauche-droite ou droite-gauche. L'autre sens est appelé sens secondaire. Le curseur se déplace alors seulement dans le sens principal. Lorsqu'une séquence de caractères est entrée dans une langue utilisant le sens secondaire, le curseur reste fixe et les caractères s'écrivent en mode insertion (figure 5).

Les textes multilingues sont stockés en mémoire sous forme de séquences d'octets représentant des codes de caractères. Afin d'obtenir une restitution fidèle des textes multilingues mémorisés en ce qui concerne les variantes positionnelles, combinaisons de sens d'écritures, etc., leur affichage à

l'écran se fait à travers les modules de saisie, c'est-à-dire que les caractères sont affichés comme s'ils provenaient du clavier. Cependant, les aspects de justification de textes multilingues incluant des écritures en sens opposés ne sont pour l'instant pas suffisamment abordés.

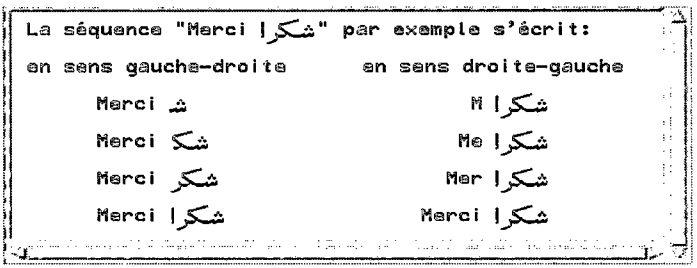

Figure 5. Directions d'écriture mixtes

5. Développements futurs

A l'heure actuelle un prototype existe, il inclut la plupart des fonctionnalités visées. Cependant, deux modules sont en cours de développement, comme il a été mentionné au passage dans le texte (ces modules seront très vraisemblablement terminés lors de la présentation à CARI'96):

- Sortie HTML pour échange de données,
- Saisie de type phonétique pour les langues idéographiques.

Parmi les développements futurs (outre l'adjonction de nouvelles langues); nous envisageons:

- l'intégration des fonctionnalités de représentation de HTML qui permettraient l'enrichissement des textes et l'utilisation de **MtScript** en association avec un navigateur WWW;
- la voyellation de l'arabe: bien que les textes arabes diffusés dans la presse et les journaux ne soient généralement pas voyellés, les voyelles jouent un rôle important dans le traitement automatique de l'arabe (figure 6).

La version actuelle de **MtScript** utilise une police de caractères non voyellés issue du code Metafont présenté par Y.Haralambous à partir du format TEX. Mais nous intégrons actuellement une version plus complète de cette fonte contenant la plupart des caractères requis par UNICODE et incluant les voyelles ainsi que les formes contextuelles des caractères.

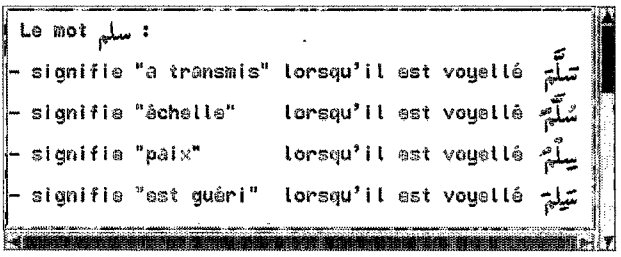

Figure 6. Voyelles en arabe

6. Conclusion

On assiste aujourd'hui à une évolution des technologies de l'information entraînant un besoin croissant en outils multilingues. Les échanges d'informations entre partenaires de langues

différentes sont de plus en plus fréquents. L'éditeur de textes multilingues est un élément d'articulation dans cette internationalisation des supports de l'information. **MtScript** a été développé dans la perspective de répondre à des besoins en outils de codage et de traitement de documents multilingues, tant pour des langues européennes que des langues non européennes. Cet outil permet de saisir, éditer, mémoriser et échanger des textes multilingues. Il ouvre des perspectives sur un certain nombre d'applications nécessitant des traitements de textes multilingues telles que la segmentation et l'analyse morphologique de textes, la traduction automatique, les dictionnaires multilingues ainsi que la localisation des logiciels (et de leur documentation) dans différentes langues.

Remerciements

Ce travail a bénéficié du financement de la Commission Européenne dans le cadre du projet MULTEXT. Diverses personnes ont contribué à l'amélioration du logiciel. Les auteurs tiennent en particulier à remercier Greg Priest-Dorman et *Langbox International* pour leur tests approfondis, Emmanuel Flachaire pour la compilation sous Linux (Intel) et Nancy Ide pour son aide sur la documentation anglaise. Nous remercions Mark Leisher du laboratoire CRL (New Mexico State University) pour ses commentaires et sa collaboration sur l'arabe, ainsi que les relecteurs anonymes pour leurs commentaires détaillés.

Bibliographie

[ACC96] `http://www. accentsoft. com`

[BECK84] J. Becker, "The multilingual word processing", *Pour La Science*, Septembre 1984, 66-67.

[BECK87] J. Becker, "Arabic word processing", *Communications of the ACM*, volume 30, number 7, Juillet 1987, 600-610.

[BENT91] P. M. Benton, "The Multilingual Edge", *BYTE*, Mars 1991, 124-132.

[BOPI95] L. Bourbeau, F. Pinard, "Normalisation et internationalisation: inventaire et prospective des normes clefs pour le traitement informatique du français". *Progiciels BPI.* , Montréal, Canada, 1995.

[BOUA90] A. M. Boualem, "The multilingual terminal", rapport de recherche, INRIA Sophia Antipolis, Janvier 1990, 1-4.

[BOUA93] A. M. Boualem, "ML-TASC: Système de traduction automatique multilingue dans un environnement à syntaxe contrôlée", SS'93, *7th annual High Performance Computing Conference*, Alberta, Canada, Juin 1993, 537-544

[BOUA95a] A. M. Boualem, "Multilingual text editing", *SNLP'95, The 2nd Symposium on Natural Language Processing*, NECTEC, C&C, Bangkok, Août 1995, 336-342.

[BOUA95b] A. M. Boualem, "Arabic Language Processing", *SNLP'95, The 2nd Symposium on Natural Language Processing*, NECTEC, C&C, Bangkok, Août 1995, 95-102.

[CBKH95] C. Bigelow, K. Holmes, "The design of a UNICODE font", version française dans le *Cahier GUTenberg* n°20, Mai 1995, 81-102.

[CRL96] http://crl. nmsu. edu

[LANG93] *Language Coding Using ISO/IEC 6429.* Draft circulated in January 1993 by the European Standardization Organization CEN technical committee TC304. Available electronically at: `http://www. stonehand. com/unicode/standard/tc304. html/`

[IDVE95] N. Ide, J. Véronis, *The Text Encoding Intiative: Background and Context*, Kluwer Academic Publishers, Dordrecht, 1995.

[JAMG95] J. André, M. Goossens, "Codage des caractères et multi-linguisme: de l'ASCII à UNICODE et ISO/IEC-10646", *Cahier GUTenberg* n°20, Mai 1995, 1-54.

[LABO95] A. Labonté, "Input methods to enter characters from the repertoire of ISO/IEC 10646 with a keyboard or other input devices". *ISO/CEI JTC1/SC18/GT9 Working Draft*, Février 1995. `ftp://ftp. funet. fi/pub/doc/charsets/ucs-input-methods`

[LANG96] http://www. spartacus. com/langbox/

[MUL96] `http://www. lpl. univ-aix. fr/projects/multext/`

[UNIW96] `http://www. wysiwyg. com`

[YERG95] F. Yergeau, G. Nicol, G. Adams, M. Duerst, "Internationalization of the Hypertext Markup Language". *Internet Draft draft-ietf-html-i18n-02*, November 1995. `http://www. ics. uci. edu/pub/html/draft-ietf-html-i18n-02. txt`

[YHJP95] Y. Haralambous, J. Plaice, "Ω, une extension de TEX incluant UNICODE et des filtres de type Lex", *Cahier GUTenberg* n°20, Mai 1995, 55-79.