

Entrer facilement « œ », « æ », « Æ », « Œ » et « Ç » sous WinEdt

Pascal Kockaert
Université libre de Bruxelles,
MonPrénom.MonNom (ad) ulb.ac.be

22 juin 2002

Motivation

Sur la proposition de Nicolas LE NOVÈRE, son gestionnaire précédent, j'ai accepté de reprendre la maintenance du dictionnaire français sous WinEdt.

Puisqu'il m'est offert d'y participer activement, je vais essayer de faire profiter les utilisateurs de WinEdt de quelques améliorations que j'ai apportées sur mon installation personnelle, à savoir l'usage des caractères quasiment inaccessibles — sur un clavier ordinaire dans un programme ordinaire — que sont « œ », « æ », « Æ », « Œ » et « Ç ».

Si tous les utilisateurs ont un accès aisé à ces signes, il devient évident que le dictionnaire doit inclure les mots qui contiennent ces signes avec leur orthographe correcte : « cœur » et non « coeur ». Bien sûr, je ne vais pas imposer ce changement et je proposerai donc deux versions du dictionnaire. Je générerai le dictionnaire le plus simple au départ du plus exact, ce qui me semble logique.

J'ENCOURAGE NÉANMOINS FORTEMENT L'UTILISATION DE L'ORTHOGRAPHE EXACTE DE CES MOTS¹.

Philosophie

Les caractères « œ », « æ », « Æ », « Œ » et « Ç » peuvent être entrés de manière très laborieuse en tapant leur code directement au clavier, ce qui se fait en maintenant la touche ALT de gauche enfoncée pendant qu'on utilise le pavé numérique pour taper le code précédé d'un « 0 ». Avec le temps, cette gymnastique peut devenir automatique, mais elle reste désagréable. Si l'on doit se contenter d'un clavier d'ordinateur portable, l'exercice devient plus que rébarbatif.

Je propose donc d'utiliser le remplacement automatique de WinEdt pour composer ces caractères. Le tout est de savoir ce que l'on veut remplacer par « œ », « æ », « Æ », « Œ » et « Ç » de sorte que la combinaison paraisse naturelle. Après mûre réflexion, j'ai mis au point un système que j'utilise depuis deux ans avec succès et que j'ai déjà eu l'occasion de partager autour de moi et sur `fr.comp.text.tex`. Il s'agit d'opérer les substitutions résumées dans le tableau ci-dessous.

Combinaison	Substitution	ALT + 0xxx
ôe	œ	156
âe	æ	230
ÂE, Âe	Æ	198
ÔE, Ôe	Œ	140
ˆC	Ç	199
<<_	« _ »	171
_>>	_»	187

L'idée qui prédomine lors de ce choix est que nous sommes déjà habitués à penser que la touche ^/ˆ modifie le caractère qui suit, lorsqu'il s'agit d'une voyelle. Il devient vite naturel d'étendre cela aux combinaisons de voyelles. La touche morte modifie le caractère qui suit. Ce caractère « ô » ou « â » n'étant jamais suivi d'un « e » en français, on peut systématiquement opérer la substitution des quatre premières lignes du tableau. Certains d'entre nous auront tendance à lâcher la touche « shift » avant de taper le « e ». D'autres le feront après. Pour que ce choix ne soit pas déterminant, les deux possibilités mènent à la construction du même caractère.

¹Contrairement aux ligatures « ct », « st », « fi », « ff », etc. qui sont ornementales et facultatives, les ligatures « œ » et « æ » font partie de l'orthographe et sont obligatoires. Ainsi, on écrit « cœur » mais « coexister » ; « cæcum » mais « paella ». La prononciation suit l'orthographe.

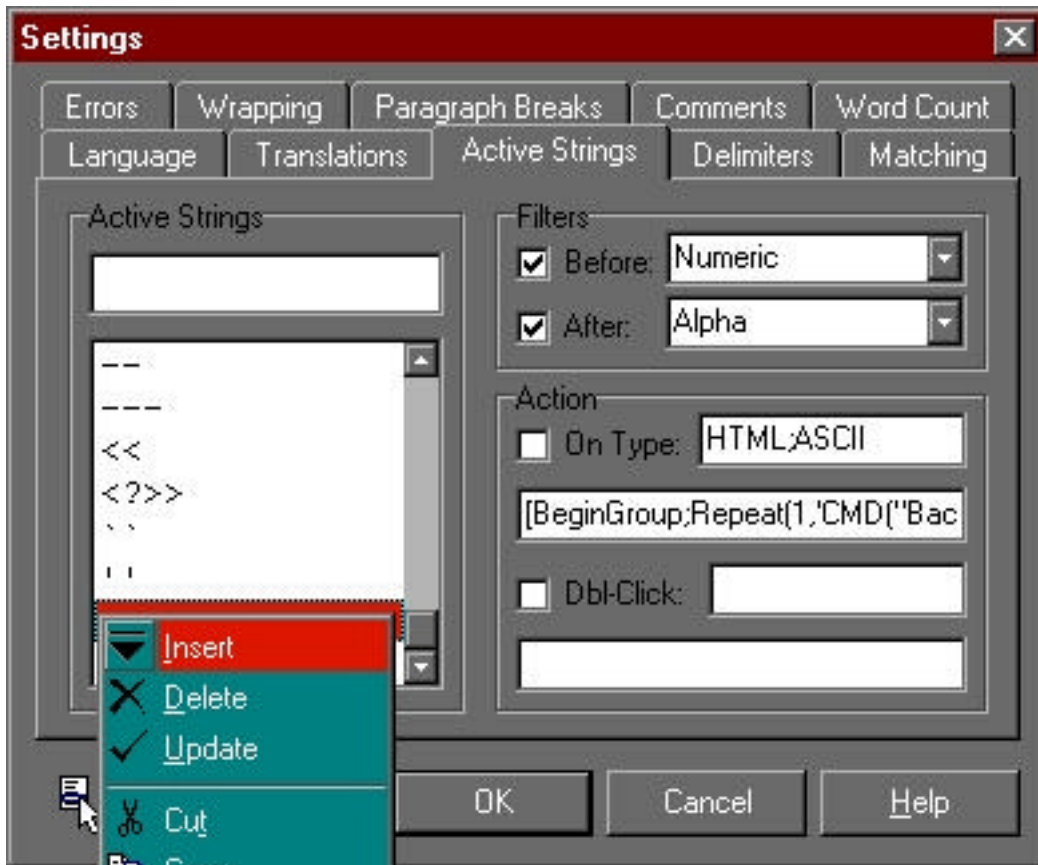


FIG. 1 – Fenêtre dans laquelle il faut effectuer les changements.

Pour le « Ç », j'aurais pu utiliser le même schéma et choisir « ^C » comme combinaison. Ce faisant, j'aurais encouru le risque d'être irrité si un jour j'avais choisi d'utiliser « C » comme exposant dans un développement mathématique : $\$a^C+b^C+c^C\neq(a+b+c)^C\$$. L'expression « C » m'apparaît par contre peu susceptible d'être rencontrée. De plus, pour entrer le « C » majuscule, il faut d'office enfoncer la touche « shift » et donc le tréma s'utilise assez naturellement.

Pour ce qui est des combinaison « << » et « >> », je n'invente rien. \LaTeX les utilise depuis belle lurette. On pourrait donc se contenter d'utiliser le codage classique. Néanmoins, pour tous ceux qui sont passés de « french » à « babel » et qui ont l'habitude de taper << french >> plutôt que <<~babel~>>, la seule solution pour éviter le retour à la ligne après les guillemets d'ouverture ou après les guillemets de fermeture est de rendre les guillemets actifs, ce qui nécessite qu'ils soient effectivement codés par un seul caractère plutôt que par deux, ce qui justifie de demander ce travail à WinEdt plutôt qu'à \LaTeX .

En pratique

Pour que WinEdt prenne en compte nos volontés, il faut aller dans le menu « Options », suivi de « Settings », puis « Active Strings » (voir Fig 1). Dans cette fenêtre, il faut se placer à la fin de la liste en dessous de « Active String » et « cliquer » du bouton droit pour insérer un élément.

Il faut alors entrer le groupe à substituer — « ôe » dans l'exemple — et cocher les cases appropriées : « On Type » suivi du type « TeX » et de la ligne de commande :

```
[BeginGroup;Repeat(2,'CMD("Backspace)');Ins("œ");EndGroup]
```

Cette ligne de commande peut être copiée, ce qui vous évitera des fautes de frappe. Néanmoins, certains logiciels peuvent remplacer le signe « œ » par « oe » ! Il faut donc entrer ce caractère transformé en utilisant la table de codes qui a été fournie (ALT+0xxx). Il est normal que certains caractères apparaissent comme de petits rectangles noirs dans la ligne de commande de WinEdt.

Il faut réitérer l'opération précédente pour les autres substitutions en utilisant les lignes de commande du tableau qui suit. Vous pouvez utiliser le « copier/coller ». Un fichier WinEdt-OE.txt contenant les lignes à copier est créé lors de la compilation de WinEdt-OE.tex.

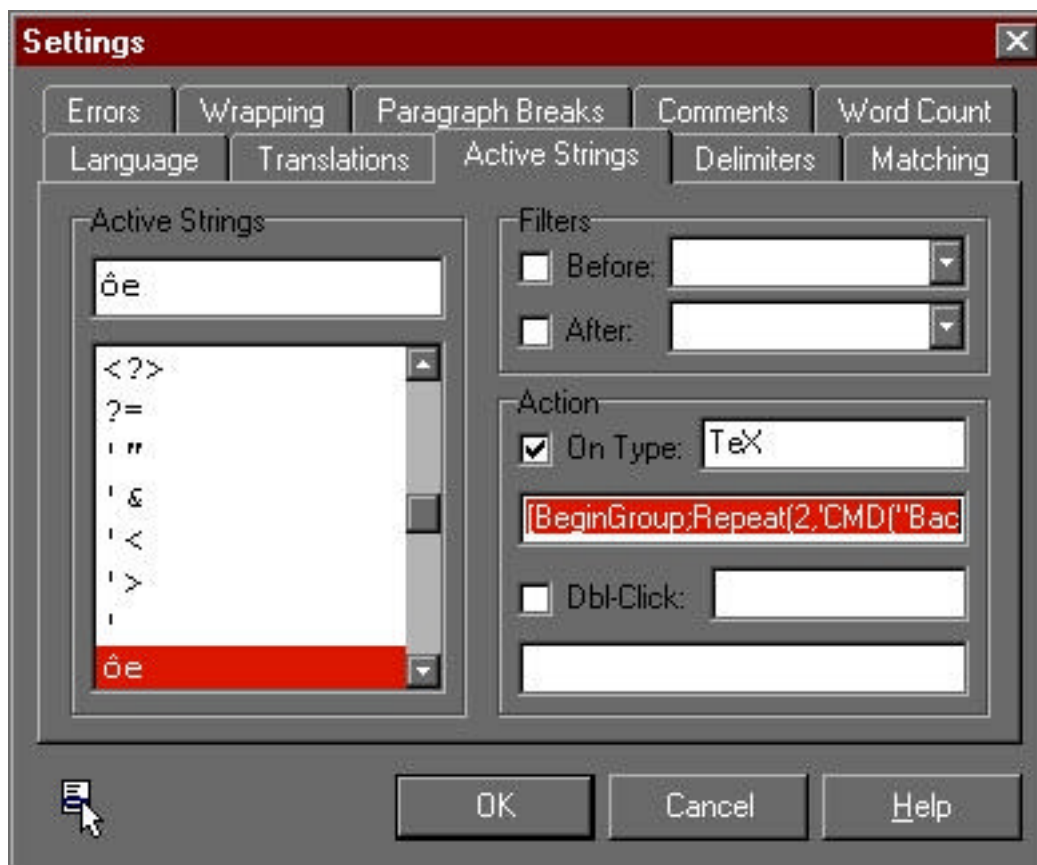


FIG. 2 – Exemple pour « œ ».

Active string	Action
ôe	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("œ");EndGroup]
âe	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("æ");EndGroup]
ÂE	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("Æ");EndGroup]
Âe	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("æ");EndGroup]
ÔE	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("Œ");EndGroup]
Ôe	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("œ");EndGroup]
ˆC	[BeginGroup;Repeat(2,'CMD("Backspace")');Ins("Ç");EndGroup]
<<	[BeginGroup;Repeat(3,'CMD("Backspace")');Ins("<<");EndGroup]
>>	[BeginGroup;Repeat(3,'CMD("Backspace")');Ins(">>");EndGroup]

Vous pourrez remarquer que les deux dernières lignes remplacent trois caractères. Bien qu'elles ne se voient pas, il y a effectivement une espace après les guillemets d'ouverture et une avant ceux de fermeture dans la description de l'« Active string ».

Bien entendu, vous êtes libres de modifier ces raccourcis comme bon vous semble.

Inconvénients

Ils ne sont pas nombreux et sont évidents. Par exemple, il devient impossible de coder simplement la suite de caractères « ôe ». Il existe néanmoins plusieurs moyens d'y parvenir. Dans un document T_EX, le plus naturel est peut-être de taper `ô{ }e`. si vous voulez uniquement ces deux caractères, il faut éviter de taper un « e » juste derrière le « ô ». Le remplacement ne se fait que quand on tape un caractère, pas quand on en efface un. La solution est donc d'entrer « ô e » puis d'effacer l'espace entre ces deux caractères. Ce n'est pas très pratique, mais ne devrait vraiment pas être fréquent en français.

Vous pouvez également désactiver temporairement le remplacement en supprimant la sélection « On Type » dans la fenêtre de définition du remplacement considéré.

Compatibilité avec d'autres systèmes d'exploitation

« \TeX me permettait de passer d'un système à l'autre sans encombre. Avec les changements qui viennent d'être fait, n'ai-je pas perdu cette belle universalité ? »

Ma réponse est « non ! » sans quoi je n'aurais pas proposé cette solution. Il faut savoir que depuis un certain temps, l'extension « inputenc » fait partie de la base de \TeX . Cette extension résout les problèmes posés par le fait que chacun des systèmes d'exploitation est aussi un monde dans lequel le numéro associé à un caractère accentué est choisi de manière indépendante de ce qui se fait dans les autres mondes.

En passant de DOS à Windows, Microsoft a rapproché sa table de caractères pour l'Europe occidentale de celle du codage « latin1 » sous UNIX (et clones). C'est pourquoi une majorité de gens utilisent

```
\usepackage[latin1]{inputenc}
```

dans le préambule d'un texte entré sous Windows. Ceci ne devrait pas être fait pour la bonne raison que le caractère « œ » est défini dans le codage Windows et n'existe pas dans « latin1 » (raccourci pour « iso-8859-1 »). Il faut donc utiliser le préambule adapté à Windows qui est

```
\usepackage[ansinew]{inputenc},
```

du moins pour les gens qui utilisent le codage « Western », alias « CP1252 ».²

En résumé de ce qui précède, moyennant l'usage correct de l'extension « inputenc », un fichier créé sous windows devrait pouvoir être compilé sous UNIX ou MAC. Je n'ai pas dit par là qu'il pourrait être facilement édité. En effet, un éditeur innocent pris sur un de ces systèmes n'affichera pas correctement notre caractère « œ ». Si l'on veut la possibilité d'éditer un même fichier dans plusieurs environnements, la solution la plus robuste, c'est de n'utiliser que 128 caractères sur les 256 disponibles, lorsque l'on sauve un fichier. Cela peut se faire en demandant à son éditeur préféré (WinEdt) de remplacer tous les caractères dont la position dans la table des caractères n'est pas universelle par une séquence de contrôle \TeX . Par exemple, « œ » sera sauvé sous la forme « $\{\backslash\text{\oe}\}$ ». Dans ce cas, on peut supprimer l'utilisation de l'extension « inputenc ».

L'utilisateur qui reçoit ce fichier et l'ouvre sur un éditeur simple lira par exemple « $\text{c}\{\backslash\text{\oe}\}\text{ur}$ », ce qui n'est pas très agréable, mais compréhensible. Si son éditeur est plus perfectionné, il remplacera systématiquement « $\{\backslash\text{\oe}\}$ » par le caractère « œ » (qui peut avoir un autre numéro d'ordre que le caractère « œ » initial).

Pour obtenir cette conversion de caractères, il faut se rendre dans le menu Options – Settings – Translations. En sélectionnant « read » et « write » dans la colonne « tables », on voit les conversions qu'effectue WinEdt lors de la lecture d'un fichier et de son écriture. Pour qu'une table soit prise en compte, il faut que la case « Enabled for » soit cochée et qu'un mode soit entré dans la case voisine. Ce mode peut s'appeler « UNIX », par exemple. Voir les figures 3 et 4.

Cela veut dire que si le mode du document est « $\text{\TeX}:\text{FR}:\text{UNIX}$ », à la sauvegarde du fichier, une conversion de caractères sera effectuée. La conversion inverse sera effectuée à l'ouverture et l'utilisateur ne se rendra compte de rien. Néanmoins, si l'on ouvre le fichier \TeX dans le bloc note, par exemple, on verra une différence entre deux fichiers de mêmes contenus, mais de types « $\text{\TeX}:\text{FR}$ » et « $\text{\TeX}:\text{FR}:\text{UNIX}$ ».

La possibilité de modifier le sous-mode UNIX alors que le fichier est en mémoire permet facilement de convertir un document UNIX au format pour windows et vice-versa, sans avoir à modifier les options de conversions de tables de caractères. Il suffit de cliquer sur « $\text{\TeX}:\text{FR}$ » dans la ligne d'affichage du bas de la fenêtre de WinEdt pour passer de « $\text{\TeX}:\text{FR}$ » à « $\text{\TeX}:\text{FR}:\text{UNIX}$ » afin de sauver un document en codage universel (seulement 128 caractères sur les 256 disponibles sont utilisés).

Idéalement, le mode UNIX devrait être défini sous la forme « $:\text{UNIX}|\ast.\ast;\ast.\text{unix}$ » dans le menu Preferences-Modes. On peut entrer cette ligne juste avant la ligne « $:\text{FR}|\ast.\ast;\ast.\text{fr}$ » par exemple (cf. Fig. 5). Ce faisant, un fichier qui aurait l'extension « .unix » sera d'office ouvert en mode « $:\text{UNIX}$ » et la conversion de caractères se fera toute seule. Si on le sauve sans cette extension et qu'on change le mode en « $\text{\TeX}:\text{FR}$ », on obtient un fichier au format Windows. Celui-ci peut être ouvert dans le bloc-notes ou passé dans un correcteur orthographique commercial, par exemple.

Pour les caractères dont il est question dans ce document, il faut que les lignes suivantes soient présentes dans Options-Preferences-Translations-read (cf. Fig. 3) :

- " $\{\backslash\text{\AE}\}$ " -> " \AE " ;
- " $\{\backslash\text{\OE}\}$ " -> " \OE " ;
- " $\{\backslash\text{\ae}\}$ " -> " \ae " ;

²Au passage, il est à noter que la table « iso-8859-1 » ne contient pas le logo de l'euro. Une nouvelle table doit donc être utilisée sous UNIX. Elle se nomme « iso-8859-15 » ou « latin15 », voire aussi « latino ». Ce passage à l'euro a permis d'inclure, dans la foulée, le caractère « œ » dans la table « iso-8859-15 ». Il se pourrait donc qu'un jour « ansinew » et « latin1 » s'appellent tous deux « latino » et que les utilisateurs de systèmes UNIX et Windows utilisent le même préambule.

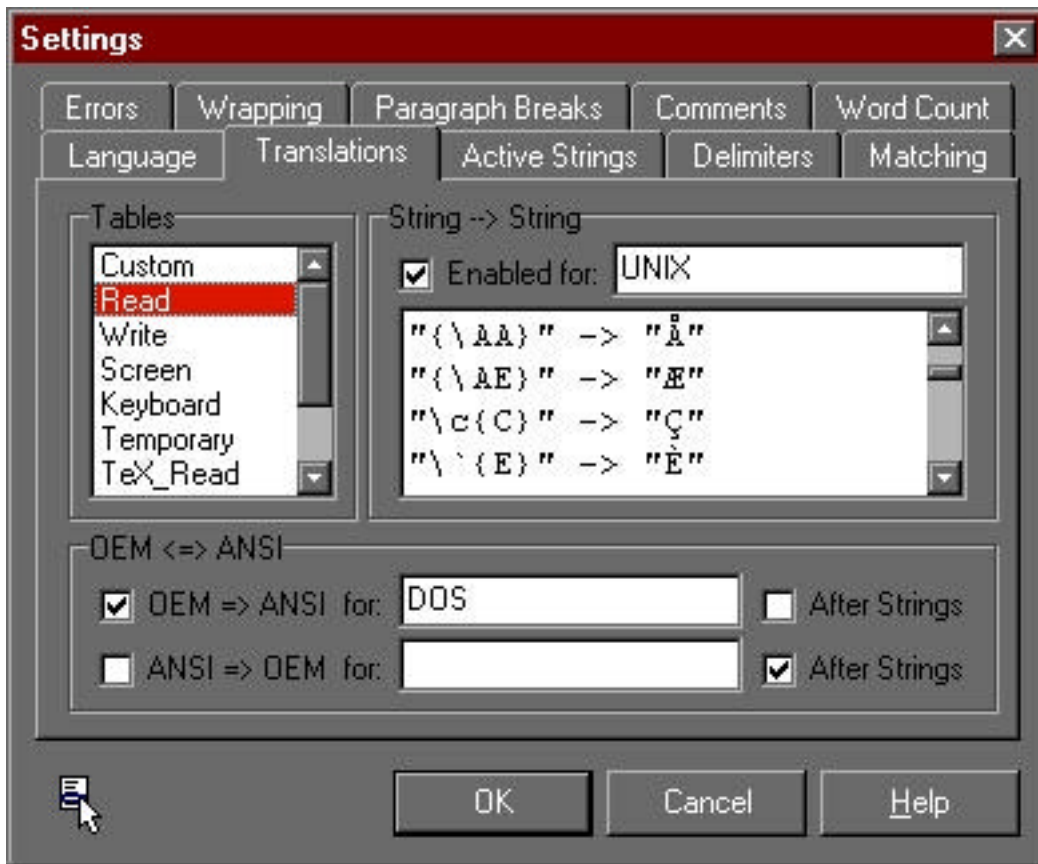


FIG. 3 – Conversions appliquées lors de la lecture d'un fichier.

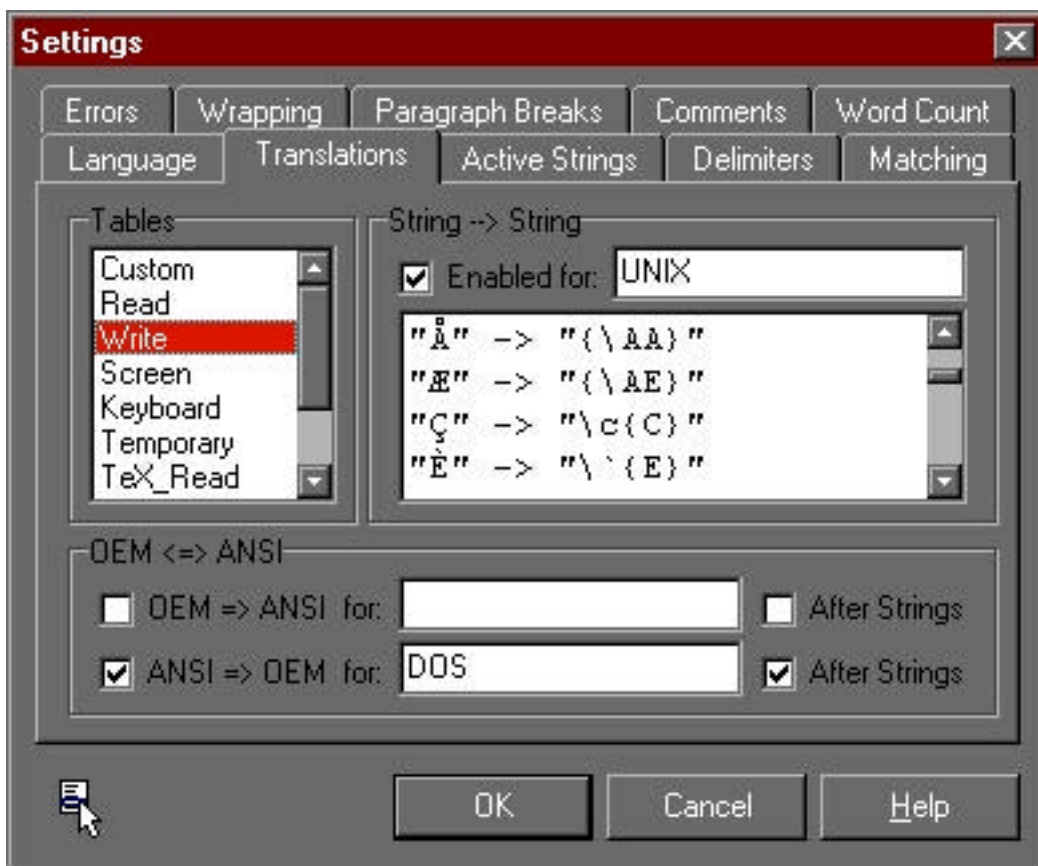


FIG. 4 – Conversions appliquées lors de l'écriture d'un fichier.

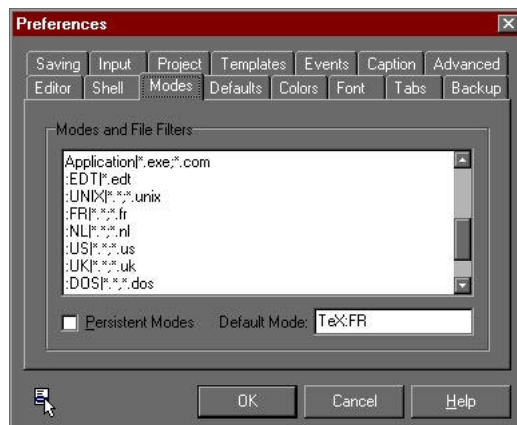


FIG. 5 – Définition des modes et sous-modes.

- "\oe" -> "œ";
- "\c{C}" -> "Ç".

De plus, les lignes suivantes doivent se trouver dans Options-Settings-Translations-write (cf. Fig. 4) :

- "Æ" -> "\AE";
- "Œ" -> "\OE";
- "æ" -> "\ae";
- "œ" -> "\oe";
- "Ç" -> "\c{C}".

Annexe : extrait du fichier « frenchb.cfg »

Le mode français de l'extension « babel » peut être configuré dans le fichier « frenchb.cfg ». Voici les lignes à entrer — d'après le modèle de fichier de configuration fourni avec babel — pour ne pas devoir introduire explicitement les espaces insécables avant ou après les guillemets.

```
%%
%% If French guillemets are available on your keyboard, you can use
%% them instead of the commands \og and \fg: REPLACE << and >> in
%% the following code by your ready-made guillemets (2 occurrences
%% for each) AND uncomment the resulting code.
%% WARNINGS:
%% 1) This will reduce the portability of your source files!
%% 2) This adds two active characters.
%% 3) If you add this stuff to the preamble of a document, you'll
%%    need to wrap it in \makeatletter ... \makeatother.
%%
\ifx\mule@def\undefined
  \initiate@active@char{« }
  \initiate@active@char{» }
  \AtBeginDocument{%
    \def« {\og\ignorespaces}%
    \def »{{\fg}}%
  }%
\else %%% For the CJK package, see MULEenc.sty.
  \mule@def{11}{\og\ignorespaces}
  \mule@def{27}{{\fg}}
\fi
%%
```