

# Comment configurer WinEdt pour L<sup>A</sup>T<sub>E</sub>X?

Christophe BONTEMPS\*

12 décembre 2001

**Abstract:** Cette note se veut un guide pratique post-installation de l'éditeur WinEdt, dans le but d'utiliser L<sup>A</sup>T<sub>E</sub>X. En particulier les raccourcis, les dictionnaires et les menus dont les abords peuvent s'avérer abscons. Puis dans une deuxième partie, nous donnerons quelques petits trucs dans le style F.A.Q. Les templates, macros et options à administrer avec précaution, seront enfin détaillées dans une dernière section.

## Contents

<b>1</b>	<b>Mise en route</b>	<b>2</b>
1.1	Test de latex . . . . .	2
1.2	Liaison DVI - L <sup>A</sup> T <sub>E</sub> X (inverse search) . . . . .	2
1.3	Pour fermer la fenêtre en fin d'exécution de L <sup>A</sup> T <sub>E</sub> X, de Yap et des autres programmes . . . . .	2
1.4	Raccourcis . . . . .	3
1.5	Dictionnaires . . . . .	4
<b>2</b>	<b>Ces petits trucs qui font gagner du temps</b>	<b>4</b>
2.1	Pour compléter automatiquement une commande . . . . .	4
2.2	Pour vérifier automatiquement les parenthèses, et autres " <i>Begin-End</i> " . . . . .	4
2.3	Pour afficher le fichier des erreurs (Log) . . . . .	5
2.4	Pour afficher la liste des labels et références diverses . . . . .	5
2.5	La barre du "bas" . . . . .	5
2.6	Et si l'on repartait à zéro... . . . . .	5
2.7	Enregistrer le numéro de licence . . . . .	6
2.8	A l'aide ! . . . . .	6
<b>3</b>	<b>Les "Templates"</b>	<b>6</b>
3.1	Comment créer un fichier de templates . . . . .	6
3.2	Insérer un raccourci pour utiliser son template . . . . .	7

---

\*INRA, Chemin de Borde Rouge, B.P. 27- 31 326 Castanet Tolosan Cedex. bontemps@toulouse.inra.fr.  
Merci à Fabian !

# 1 Mise en route

**Prérequis** : Je supposerai ici que vous disposez d'un ordinateur (en état de marche) sur lequel est déjà installé une distribution  $\text{\LaTeX}$ . Par défaut, je supposerai qu'il s'agit de Miktex, mais évidemment cela peut être une autre distribution.

La première mise en route de WinEdt commence par le `configuration wizard` auquel on peut prêter attention, mais qui n'est pas exhaustif. On y trouve toutefois des indications sur la *mise en forme* des documents dans WinEdt (Wrapping), sur les *sauvegardes* (qui sont assez intelligemment pensées), les *options*, les *dictionnaires* et sur *les menus*. Je vous conseille d'y prêter attention, vous pouvez toutefois retrouver ce "sorcier" (wizard), dans le menu `Help>Configuration~Wizzard`.

## 1.1 Test de latex

Ouvrez un fichier  $\text{\LaTeX}$ , compilez le (bouton  $\text{\LaTeX}$  en haut), visualisez le (bouton DVI). Vous devriez tout voir...

## 1.2 Liaison DVI - $\text{\LaTeX}$ (inverse search)

Depuis le visualisateur DVI, le célèbre programme YAP, on accède au Menu `View>Options`. Cette action ouvre une boîte de dialogue avec différents onglets. Sélectionnez dans l'onglet `Inverse search` la boîte `Program` et affichez `WinEdt` (La ligne de commande dessous s'éclaire d'un superbe :

```
C:\Program Files\WinEdt\WinEdt.exe [Open('%f');SelLine(%1,8)].
```

Désormais, si vous *double-cliquez* dans Yap, vous allez directement, dans WinEdt, à la ligne correspondante.

## 1.3 Pour fermer la fenêtre en fin d'exécution de $\text{\LaTeX}$ , de Yap et des autres programmes

Vous remarquerez que  $\text{\LaTeX}$  laisse une "*fenêtre DOS*" ouverte à la fin de l'exécution. Pour remédier à cela (et éviter les courants d'air), il faut supprimer une commande `PAUSE`, DANS le fichier système exécuté depuis WinEdt.

Ouvrez (avec n'importe quel éditeur, pourquoi pas WinEdt d'ailleurs) le fichier :

```
C:\program files\winEdit\Bin\Miktex\Latex.Bat.
```

Le fichier comporte 3 lignes :

```
@ECHO OFF
```

LATEX.EXE %1 %2 %3 %4 %5 %6 %7 %8 %9

pause

Supprimez la dernière ligne (pause), sauvez et fermez. C'est idem pour DVIPS

C:\program files\winEdit\Bin\Miktex\Dvips.Bat

## 1.4 Raccourcis

La tâche est simple mais demande une bonne compréhension des menus de Winedit. Tout d'abord il faut connaître l'emplacement de la commande dont vous voulez le raccourci dans le menu général. Par exemple, on trouve la commande  $\LaTeX$  sous forme de bouton **mais aussi** dans le menu **Accessories>Latex**.

**Remarque :**

Il y a déjà un raccourci par défaut (CTRL+SHIFT+L).

Le menu **Option>Menu Setup** ouvre une boîte de dialogue comprenant des *onglets*. Dans **main Menu**, une liste de commandes apparaît, si l'on double-clique sur l'une de ces commandes une deuxième boîte de dialogue apparaît (en sur-impression).

Double cliquez sur **&Accessories** une deuxième boîte apparaît, dans laquelle vous pouvez spécifier des tas de choses (dont le raccourci). Placez vous sur **&Latex** dans le **Menu Item** vous observez bien, à droite dans la boîte **shortcut** que le raccourci par défaut est CTRL+SHIFT+L.

**Remarque :**

La logique est de suivre le chemin du menu déroulant dans laquelle figure la commande; la commande  $\LaTeX$  figure dans le menu **Accessories>Latex** on suit donc la succession de boîtes **&Accessories >&Latex**.

Placez vous dans cette fenêtre. Il suffit d'appuyer sur les touches de votre clavier pour voir apparaître le nouveau raccourci (par exemple **Ctrl+L**).

Vous pouvez alors vérifier que ce raccourci n'est pas déjà utilisé en cliquant sur le petit icône de droite **Context Shortcut Menu** et en optant pour **?is used** une fenêtre vous indique alors si le raccourci est déjà utilisé. Un simple **OK** valide l'opération.

**Remarque :**

les raccourcis utilisés précédemment sous PFE ne posent pas de problème à savoir **Ctrl+L** pour  $\LaTeX$  ; **Ctrl+Y** pour YAP, et **Ctrl+P** pour DVIPS, mais attention de ne pas perdre des raccourcis utiles comme **Ctrl+C** =COPY, **Ctrl+V**= PASTE, **Ctrl+S**= SAVE, **Ctrl+F12** utilisé par WinEdt (voir plus loin).

## 1.5 Dictionnaires

Par défaut WinEdit "*travaille*" avec un dictionnaire anglais et "*comprend*" les symboles L<sup>A</sup>T<sub>E</sub>X, qu'il écarte de la vérification orthographique. Il existe un dictionnaire français `fr.dic`, que je mettrai en libre service. Placer ce dictionnaire dans `C:\program files\winEdit\Dict\FR\`.

Pour pouvoir utiliser ce dico, il faut aller le déclarer dans le gestionnaire de dictionnaires (outil complexe mais très puissant). Le Menu `Options>Dictionary` propose une boîte de dialogue assez ésotérique.

- Dans l'onglet `Dictionaries`, boîte `Dictionaries` placez vous en dernière ligne blanche de la liste des dictionnaires déjà existants: `English`; `USER`, `LATEX`
- Bouton droit `Insert`, dans l'onglet supérieur (là où clignote votre curseur) tapez le nom de votre dictionnaire (ce que vous voulez), celui-ci apparaît dans la liste de dictionnaires
- Bouton droit sur ce nom, `Browse`, indiquez le chemin de `fr.dic` et cochez `Enabled`

Comme vous le remarquerez, chaque dictionnaire a ses options à cocher à droite de la boîte, vous pouvez donc choisir le ou les dicos par défaut, l'activer ou le désactiver, etc...

Les autres options, dans les autres onglets sont assez faciles à comprendre et ne sont pas détaillées ici ..

## 2 Ces petits trucs qui font gagner du temps

### 2.1 Pour compléter automatiquement une commande

Tapez le début de la commande, par exemple `\subs`, puis `Ctrl+Enter` et une boîte vous propose différentes solutions pour compléter votre commande.

### 2.2 Pour vérifier automatiquement les parenthèses, et autres "*Begin-End*"

Dans une formule, ou dans le corps d'un document, il arrive souvent que l'on se perde dans les parenthèses, ou que l'on ait la flemme de fermer les commandes ouvertes. Pas mal d'options vous sont proposées dans WinEdit (voir le menu `Search>Delimiters`). Dont :

- `CTRL+F12` qui **analyse** et **reporte** les erreurs
- `Alt+F12` qui **analyse** et **répare automatiquement** les erreurs en ajoutant ce qu'il faut, ou il faut, y compris les `\end{bidule}`

## 2.3 Pour afficher le fichier des erreurs (Log)

Menu `Search>Errors` (ou `CTRL+E` par défaut) vous affiche le fichier Log (en petit en bas).

## 2.4 Pour afficher la liste des labels et références diverses

Il suffit de taper `\cite{}` ou `\ref{}`. La fermeture de l'accolade déclenche alors l'affichage de toutes les références du document dans une fenêtre additionnelle en bas, dans laquelle on peut faire son choix sans risque d'ambiguïté.

## 2.5 La barre du "bas"

La barre du bas, ou "*status line*", vous donne de nombreuses indications et permet d'activer/désactiver de nombreuses fonctions. En particulier vous y trouvez, de gauche à droite:

- Une aide sur la barre du bas
- La position du curseur ligne:colone ; un click permet de chercher une ligne
- Le nombre de mots tapés ; un click vous donne les statistiques sur le document
- Une indication de l'état du document (`modified` s'il n'est pas sauvé, rien sinon); Un click vous donne des infos sur le document en cours
- Un selecteur de formatage de colone `Wrap` signifie un retour automatique forcé à la ligne; un click active ou désactive cette fonction (**très utile !**)
- `INS` permet de changer le mode d'insertion du curseur
- `LINE/Block` je ne sais pas !
- `Spell` permet d'activer ou de désactiver la correction haurtografique automatique (**très agréable!**)
- Plus loin, un click sur `WinEdt` permet d'ouvrir un nouveau fichier.

Enfin un click sur le bouton droit fait apparaitre un menu fort utile..

## 2.6 Et si l'on repartait à zéro...

Il existe une manière de rétablir tous les paramètres par défaut, dans le menu `options>configurations>virgin start`

## 2.7 Enregistrer le numéro de licence

Menu `Help>Register` dans l'onglet `Nom` tapez l'identifiant un truc comme

`INRA - LEERNA (10-user Site License)` et dans la case `code` tapez les chiffres de votre licence à la place de `Unregistered`

## 2.8 A l'aide !

Il existe pas mal de façon de trouver de l'aide dans WinEdt.

- Un classique "help" (dans le Menu `Help`)
- Un manuel toujours dans ce même menu, qui donne de bonnes idées
- dans le répertoire `C:\Program Files\WinEdt\Doc` vous trouverez un ensemble de fichiers explicatifs des fonctions de WinEdt
- Le site <http://www.winedt.org/Macros/index.html> propose pas mal de macros et outils utiles pour l'utilisateur
- Enfin, vous pouvez toujours m'appeler...

## 3 Les "Templates"

On désigne par ce terme "*Template*" des petits bouts de commandes que l'on souhaite écrire souvent ou répéter. WinEdt en propose par défaut quelques uns, par exemple dans le menu `insert` l'on dispose de plusieurs "templates" comme `Equation` qui donne :

```
\begin{equation}\label{}  
{}  
\end{equation}
```

La procédure ci-dessous vous indiquera comment en créer par vous même.

### 3.1 Comment créer un fichier de templates

Un "template" désigne un fichier comportant des commandes que l'on souhaite pouvoir insérer directement dans un document `LATEX`, par exemple parce que l'on s'en sert souvent, on peut donc y mettre n'importe quoi, même si ce n'est pas du `LATEX`.

Sous `C:\program files\WinEdt\templates` créer son propre fichier(avec n'importe quel éditeur) et lui donner l'extension `.ltx`, en s'inspirant des fichiers déjà créés.

## 3.2 Insérer un raccourci pour utiliser son template

Le principe est simple mais sa réalisation au départ un peu "obscur". L'idée est de **créer** un raccourci, puis de le **lier** au fichier contenant le "template", via une **action**.

Dans `Option\Menu Setup` double click sur `&insert` (c'est là que sont définis les menu, macros et autres éléments à insérer) Vous trouverez d'ailleurs là exactement les éléments du menu déroulant `insert`. Placez vous dans la rubrique que vous préférez (par exemple `&Latex`.

Cliquez sur le bouton droit `insert` donnez un nom à votre commande par exemple `Truc`, Vous avez **créé** votre raccourci. Il faut ensuite indiquer dans la ligne `Macro` l'action à réaliser c'est à dire insérer les lignes du fichier template (`truc.ltx`). Là point de salut sans copier coller les instructions des autres raccourcis et de modifier le nom du fichier, c'est ainsi que l'on définit l'**action** à effectuer (ici l'insertion).

Par exemple, en recopiant depuis la fenêtre `Macro` la commande de `&Equation` et en changeant le nom du fichier contenant le template de `&Equation`, on a :

```
[SetSel(0);Mark(9);ReadFile(%B\Templates\LaTeX\Truc.ltx,9);InsLine;CMD(Go To Beginning Of Line);InsText(%!9);Goto(9);CMD(Next Bullet)]. Ceci réalise la liaison entre le raccourci et le fichier, et indique également que l'on souhaite insérer ce fichier.
```

Cette commande placée dans le champ `macro` correspondant au raccourci `truc` fera très bien l'affaire pour insérer le fichier template `Truc.ltx` dont le contenu est :

```
\begin{truc}
truc
\end{truc}
```

On peut facilement par le menu proposé, associer un icône ou une séquence de touches de raccourci. Bien sûr, cela semble ésotérique, mais c'est assez puissant puisque l'on peut sûrement faire (programmer) d'autres actions que l'insertion!!