

Documentation sur le module frenchb de Babel

Daniel FLIPO

Version 1.6e – 28 janvier 2004

Résumé

Le module de francisation de Babel a été complètement réécrit pour produire une typographie de qualité, conforme aux recommandations de l’Imprimerie nationale. Il s’appelle maintenant `frenchb.ldf` (noter le ‘b’ indiquant que ce fichier fait partie de Babel) et remplace définitivement l’ancien fichier `français.ldf` dans la distribution Babel depuis 1996.

Le fichier `frenchb.ldf` n’est pas distribué tel quel, on trouvera sur le serveur de GUTenberg ou sur CTAN deux fichiers `frenchb.dtx` et `frenchb.ins`, le premier contient le code commenté et la documentation complète en anglais, le second permet d’en extraire `frenchb.ldf` (voir comment section 4).

1 Appel de l’extension Babel

Babel est installé en standard dans toutes distributions $\text{\LaTeX} 2_{\epsilon}$, pour disposer des langues française et anglaise¹, il suffit d’ajouter

```
\usepackage[english,frenchb]{babel} (frenchb ‘b’ comme ‘Babel’) ou bien  
\usepackage[english,français]{babel}
```

dans le préambule du document (entre `\documentclass` et `\begin{document}`).

Ces deux commandes sont équivalentes et font appel au même module `frenchb.ldf` inclus dans Babel. La dernière langue chargée (le français dans l’exemple ci-dessus) est active au début du document.

Attention : la variante `\usepackage[english,french]{babel}` n’est équivalente aux deux commandes précédentes que depuis la version 3.7j de Babel (livrée avec TeXLive 2003). Avec les versions antérieures, son effet *dépend du système* : si l’une des extensions `frenchle` ou `frenchPro` de Bernard GAULLE est installée c’est elle qui est activée, sinon c’est le module `frenchb` de Babel. Le fichier `.log` indique bien quel est le fichier effectivement chargé (`frenchle.ldf`, `french.ldf` ou `frenchb.ldf`), mais qui le lit vraiment ?

L’option `french` étant ambiguë, il vaut mieux, dans un souci de portabilité, utiliser soit `frenchb` (ou `français`) soit `frenchle` ou `frenchPRO` selon l’extension désirée. Ceci évitera des surprises en cas de compilation d’un même source sur des installations différentes.

Remarque importante : les fontes utilisées par défaut par \LaTeX (appelées CM ou *Computer Modern* et correspondant au codage OT1 à 128 caractères) ne permettent pas d’obtenir des césures correctes des mots contenant des signes diacritiques (accents, cédilles, etc.).

1. en fait américaine (US-english), il existe une variante `british` pour l’anglais « britannique ».

Pour travailler en français, le recours à Babel/frenchb ne suffit pas, il faut en plus utiliser des fontes contenant tous les caractères spéciaux du français (â, é, ï, ù, ç, etc.), ce qui peut se faire en ajoutant les deux commandes

```
\usepackage[T1]{fontenc}
```

```
\usepackage{lmodern}
```

dans le préambule du document. Ceci impose l'utilisation des fontes LM ou *Latin Modern* qui conviennent pour toutes les langues de l'Europe de l'ouest (latines, anglo-saxonnes et scandinaves). Si les fontes LM ne sont pas installées (leur sortie date de l'été 2003, elles sont sur les CD TeXLive 2003), on pourra remplacer les deux commandes ci-dessous par

```
\usepackage{aequill}
```

ce qui revient à faire appel à des fontes virtuelles en codage T1 dérivées des CM.

L'utilisation de fontes de la famille CM/LM n'est pas du tout indispensable, toute fonte de codage T1 convient, ainsi on pourra remplacer `lmodern`, au choix par `txfonts` (Times), `pxfonts` (Palatino), `fourier` (Utopia), etc.

Pour passer d'une langue à l'autre en cours de document on utilise la commande standard de Babel `\selectlanguage{langue}`, par exemple `\selectlanguage{french}` (ses équivalents `\selectlanguage{frenchb}` et `\selectlanguage{français}` conservés pour des raisons de compatibilité ne devraient plus être utilisés) et `\selectlanguage{english}`. Pour passer *localement* dans une autre langue on peut utiliser l'environnement

```
\begin{otherlanguage}{langue}
```

```
  texte...
```

```
\end{otherlanguage}
```

ou pour une courte citation dans un paragraphe

```
\foreignlanguage{langue}{texte... }.
```

2 Description de la francisation par Babel/frenchb

La commande `\selectlanguage{french}` produit les effets suivants :

- Les motifs de césures françaises sont activés.
- Les caractères de double ponctuation (; : ! ?) sont rendus actifs pour que les espaces précédant ces caractères soient insécables et conformes aux usages français².
- la définition de `\dots` est modifiée pour supprimer l'espace qui suit (cet espace est gênant devant une parenthèse par exemple...).
- La commande `\today` donne la date en français.
- Les titres (« caption names » en anglais) sont traduits en français, ainsi par exemple la commande `\chapter` imprimera « Chapitre » au lieu de « Chapter ».
- Les légendes des figures et des tables utilisent un tiret double en français à la place du ':', on obtient 'Fig. 1 – Légende' au lieu de 'Fig 1: Légende'. Il est possible de choisir un autre séparateur : par exemple pour remplacer le tiret double '–' par un tiret triple '—', ajouter dans le préambule

```
\addto\captionfrench{%
```

```
  \def\CaptionSeparator{\space\textemdash\space}}.
```

Les légendes des figures et des tables sont laissées intactes si la classe utilisée ou une extension (`caption.sty` par ex.) a modifié la définition standard de la commande

2. Voir section 3 en cas de problème dû aux caractères actifs.

`\@makecaption.`

Il est également possible de modifier les noms ‘Fig.’ et ‘Tab.’ :

```
\addto\captionsfrench{%
  \def\figurename{\scshape Figure}%
  \def\tablename{\scshape Table}}.
```

- Le premier paragraphe de chaque section est mis en retrait³, comme tous les autres paragraphes, (ce n’est pas le cas en anglais).
- Listes « `itemize` » :
 - les marqueurs traditionnels du type ‘•’ sont remplacés par des tirets doubles ‘–’, ou par un autre marqueur choisi par l’utilisateur : par exemple `\renewcommand{\FrenchLabelItem}{\textemdash}` remplacera le tiret double par un tiret triple (ce choix est global, il affecte tous les niveaux de liste comme c’est l’usage en français). Il est également possible de choisir des marqueurs différents pour chaque niveau en utilisant la syntaxe standard de \LaTeX : `\renewcommand{\labelitemii}{\textbullet}`. Si une telle redéfinition est faite lorsque la langue courante est le français, elle sera sauvegardée en cas de changement de langue (elle n’affecte pas les listes des autres langues) et sera réactivée lors du retour en français.
 - les espaces verticaux ajoutés par \LaTeX sont supprimés en français ; il est possible de revenir aux réglages standard de \LaTeX en ajoutant dans le préambule la commande `\FrenchItemizeSpacingfalse`.
- Par défaut les espacements verticaux de *toutes* les listes standard ou créées par l’utilisateur sont réduits, il est possible de revenir aux réglages standard de \LaTeX en ajoutant dans le préambule la commande `\FrenchListSpacingfalse`, cette commande est indépendante de celle opérant sur les listes « `itemize` ».

La commande `\selectlanguage{english}` ramène au comportement standard de \LaTeX (typographie anglo-américaine).

À partir de la version 1.6a, deux commandes ont été ajoutées afin de permettre de présenter les notes de bas de pages selon les recommandations de l’Imprimerie nationale :

- `\AddThinSpaceBeforeFootnotes`, qui doit être placée dans le préambule ou dans le fichier `frenchb.cfg`, ajoute une espace fine insécable lors de l’appel de chaque note comme ceci⁴ au lieu de ceci⁵.
- `\FrenchFootnotes` modifie la présentation de la note elle-même : comparer la note 4 (standard) et la note 5 (à la française). La commande `\FrenchFootnotes` peut s’utiliser avant ou après le `\begin{document}`, il est possible de revenir à tout moment à la présentation standard des notes de bas de page grâce à la commande `\StandardFootnotes` ; ceci peut être utile dans les environnements `minipage` où les notes sont référencées par des lettres. La commande `\FrenchFootnotes` n’a aucun effet sur les notes référencées par des symboles (comme celles produites par la commande `\thanks` utilisée par `\maketitle`).

Aucune de ces commandes n’est active par défaut, et toutes deux agissent sur *toutes* les notes de bas de page que la langue courante soit le français ou non.

3. Les trois points suivants affectent la présentation globale du document (la *maquette*), certains auteurs préfèrent que les changements de langue n’affectent pas cette présentation, ceci est possible : voir les commandes `\StandardLayout` et `\FrenchLayout` ci-dessous.

⁴Une note de bas de page standard.

⁵Une note de bas de page à la française (avec `\FrenchFootnotes`).

Des commandes ont été prévues pour faciliter la saisie :

- Les guillemets peuvent être saisis grâce aux commandes `\og` et `\fg`⁶, ce qui donne selon la langue de travail « guillemets français » ou “english quotes”.
- Une commande `\up` facilite la saisie des exposants en mode texte : `M\up{me}` imprime M^{me} , `1\up{er}` donne 1^{er} ; on dispose aussi de `\ier \iere \iers \ieres \ieme \iemes` pour 1^{er} , 1^{re} , 1^{ers} , 1^{res} , 2^{e} , 2^{es} .
- L’usage français veut que les noms propres soient imprimés en petites capitales et ne soient pas coupés en bout de ligne, la commande `\bsc` (*boxed small caps*) en facilite la saisie : on saisira par exemple `Leslie~\bsc{Lamport}` pour obtenir Leslie LAMPORT sans risque de coupure LAM-PORT en bout de ligne.
- les commandes `\primo`, `\secundo`, `\tertio` et `\quarto` peuvent être utilisées dans les énumérations ; elles donnent 1° , 2° , 3° , 4° .
- Les abréviations de « numéro », N° et n° , sont obtenues en tapant `\No` et `\no` ;
- Le caractère $^{\circ}$ (à ne pas confondre avec le petit ‘o’ de N°) est saisi `\degre`, mais comme les espaces avant et après ce caractère dépendent fortement de la police utilisée (PostScript ou non), on emploiera `\degres` pour saisir les températures « 20°C » (codé `20~\degres C` avec espace insécable) ou les titres alcooliques « 45° » (codé `45\degres` sans espace). Lorsque l’extension `textcomp.sty` est chargée (elle donne accès aux « T_EX Companion fonts » qui contiennent un vrai symbole degré), `\degres` utilise celui-ci (`\textdegre`).
- En mode mathématique, la virgule est toujours suivie d’un espace car elle est considérée comme une ponctuation et non comme une virgule décimale. La commande `\DecimalMathComma` supprime cet espace tandis que `\StandardMathComma` permet de revenir au comportement par défaut. Remarquer qu’après une commande `\DecimalMathComma`, il est nécessaire de saisir un espace explicite dans les listes et les intervalles par exemple `$(x, \ y)$` et `$$[0, \ 1]$$`. Ces deux commandes sont utilisables n’importe où dans le document.
- Une commande `\nombre` (utilisable indifféremment en mode texte et en mode mathématique) formate automatiquement les nombres entiers ou décimaux, les tranches de trois chiffres étant séparées par des espaces en français et des virgules (usage anglo-saxon) sinon : `\nombre{1234,567890}` donne 1 234,567 890 en français et 1,234.567,890 en anglais.
En français, l’espace entre les tranches de trois chiffres est par défaut une espace mot, si on préfère une espace fine, ajouter dans le préambule la commande `\ThinSpaceInFrenchNumbers`.
- Deux commandes `\FrenchLayout` et `\StandardLayout` (qui ne peuvent être utilisées qu’avant le `\begin{document}`) ont été ajoutées : elles ont pour objet d’uniformiser la présentation des listes et des premiers paragraphes des sections (à la française ou à l’américaine) pour tout le document, indépendamment des changements de langue. `\StandardLayout` peut également être utile aux développeurs de classes qui ne souhaitent pas que `frenchb` interfère avec leur maquette.

Il est recommandé de profiter des avantages offerts par l’extension `xspace` (il suffit d’ajouter `\usepackage{xspace}` dans le préambule) : les espaces suivant les commandes `\ier \ieres \ieme ... \iemes \fg` et `\dots` seront respectés sans avoir à les faire suivre de `{}` ou `_`.

6. Si les caractères « et » sont accessibles au clavier, ils peuvent être utilisés pour saisir les guillemets, voir la section 5.

3 Problèmes avec les caractères actifs ; : ! ?

Normalement, le nécessaire est fait par Babel pour que les caractères rendus actifs ne perturbent pas les autres extensions... mais il y a hélas des exceptions (`xypic`, `xcolor`, `arabtex` par exemple).

Il est possible de désactiver un ou plusieurs caractères actifs grâce à la commande `\shorthandoff` de Babel. Le mieux est de le faire *localement* comme ceci :

```
{\shorthandoff{;:!?}%  
Partie ne supportant pas les caractères actifs de frenchb...  
}
```

4 Installation

Pour la création de formats adaptés aux langues utilisées, consultez la mise à jour (mars 2001) de l'article paru dans le cahier GUTenberg n° 32, mise à jour disponible à l'URL <http://math.univ-lille1.fr/~flipo/frenchb/format-LaTeX.pdf>.

Pour vérifier que votre format LaTeX fonctionne correctement au niveau des césures, au moins en français et en anglais, téléchargez le fichier de test <http://math.univ-lille1.fr/~flipo/frenchb/frenchb-cesures.tex>.

Pour installer une mise à jour de frenchb, procédez de la manière suivante :

- récupérez la dernière version de `frenchb.dtx` ainsi que le fichier `frenchb.ins` à l'URL : <http://math.univ-lille1.fr/~flipo/frenchb/> ;
- lancez L^AT_EX sur le fichier `frenchb.ins`, ceci produit deux nouveaux fichiers `frenchb.ldf` et `frenchb.cfg`⁷ ;
- remplacez l'ancienne version de `frenchb.ldf` par la nouvelle (elle se trouve généralement dans le répertoire `texmf/tex/generic/babel`), si Babel est déjà correctement installé, il est inutile de refaire les formats.

5 Personnalisation

Comme il est d'usage avec Babel, le comportement de `frenchb` peut être personnalisé grâce à un fichier de configuration `frenchb.cfg`. Cette possibilité ne devrait être utilisée *qu'avec circonspection* pour ne pas aboutir à des versions disparates et incompatibles d'une installation à une autre.

Un fichier type `frenchb.cfg` est généré automatiquement en même temps que `frenchb.ldf` à partir des fichiers `frenchb.dtx` et `frenchb.ins` (voir section 4).

On y trouvera quelques possibilités de personnalisation, toutes sont mises en commentaires, il suffit de supprimer les `%` en début de ligne pour les activer. Ce fichier est donné *à titre d'exemple*, le mieux, pour préserver la portabilité des documents, est d'en extraire les options utiles et de les placer dans le préambule de chaque document, entre

7. Pour l'utilisation de `frenchb.cfg`, voir section 5.

`\makeatletter` et `\makeatother` si nécessaire (c'est-à-dire si des @ figurent dans les noms de commandes).

Une de ces possibilités mérite d'être décrite ici : normalement, les auteurs devraient saisir un espace avant chaque signe de double ponctuation en français. Malheureusement, il existe des auteurs distraits, aussi par défaut `frenchb` ajoute l'espace adéquat dans le texte imprimé, qu'un espace soit saisi ou non. Ceci est souvent confortable, mais a des inconvénients : dans certains cas on ne souhaite pas d'espace avant un ':' par exemple dans un chemin MS-DOS ou une URL. On peut toujours dans ces cas-là, soit passer localement en anglais, soit utiliser la commande `\string` et saisir `C\string:TEX` ou `http\string://www.ens.fr`.

Ceux qui sont sûrs de leur saisie peuvent souhaiter débrayer cette fonction : il leur suffit d'ajouter `\NoAutoSpaceBeforeFDP` dans le préambule de leurs documents.

Il est également possible de placer la commande `\NoAutoSpaceBeforeFDP` n'importe où dans le texte source, puis de revenir au comportement par défaut de `frenchb` en tapant `\AutoSpaceBeforeFDP`.

On trouvera aussi dans le fichier `frenchb.cfg`, une possibilité de saisie simplifiée des guillemets français utilisant les symboles « et » existant sur certains claviers.

6 Compatibilité avec `frenchPro`

Il est souhaitable qu'un texte saisi avec `frenchPro` de Bernard GAULLE puisse être compilé avec un minimum de modifications sur une machine utilisant Babel et `frenchb` et réciproquement.

Des efforts ont été faits en ce sens, le fichier `frenchb.cfg` contient (en commentaires) des définitions facilitant la compilation sous Babel d'un texte écrit pour `frenchPro`.

En ce qui concerne les guillemets français, `frenchPro` rend actifs les caractères < et > afin de saisir les guillemets sous la forme << et >> tandis que `frenchb` s'y refuse et utilise en interne des macros `\og` et `\fg`.

Lorsqu'on travaille en codage T1 avec `frenchb`, l'existence des ligatures << et >> permet de saisir <<~guillemets français~>>, mais les espaces insécables sont *indispensables*. Les commandes `\og` et `\fg` (ou leurs alias « et », à condition que ceux-ci aient été activés dans `frenchb.cfg` ou dans le préambule) sont préférables : elles introduisent automatiquement des espaces insécables et plus fines pour un meilleur rendu typographique.

Les points de suspensions sont saisis ... avec `frenchPro` et `\dots` avec `frenchb`.

Pour ceux qui éditent leurs sources L^AT_EX avec emacs, une fonction Lisp `french2b` opère une adaptation *partielle* d'un fichier conçu pour `frenchPro` facilitant sa compilation avec `frenchb`. L'appel à `frenchPro` est remplacé en un appel à `babel/frenchb`, les guillemets << et >> sont convertis en `\og` et `\fg` et les ... en `\dots`, enfin quelques commandes spécifiques à `frenchPro` sont ajoutées dans le préambule.

Cette fonction Lisp est disponible sur <http://math.univ-lille1.fr/~flipo/frenchb/french2b.el>. Il suffit de l'ajouter à un `.emacs` et de l'exécuter par `Esc x french2b` sur le fichier à convertir.