

# Complètement de Commande avec T<sub>E</sub>XShop

Herbert Schulz

[herbs2@mac.com](mailto:herbs2@mac.com)

30 janvier 2010

## Introduction

Depuis la version V1.34, T<sub>E</sub>XShop offrait un service de complètement de commande (*Command Completion facility*) qui était raisonnablement puissante, si sous-utilisé. Le complètement de commande de T<sub>E</sub>XShop permet les achèvements (compléments) ou les substitutions (abréviations) pour un ensemble de caractères, délimités à gauche par un caractère de début de mot (*Word Boundary Character*)<sup>1</sup> et déclenchés par la touche d'échappement (⌘).

Avec l'aide des contributeurs de la liste mél de Mac OS X T<sub>E</sub>X<sup>2</sup>, une mise à jour du fichier `CommandCompletion.txt` a été créée en même temps que des macros Applescript associées pour tirer profit de ce service. Les compléments et abréviations fournis contiennent souvent des gros points, « • », appelés repères (*Marks*)<sup>3</sup>, utilisés comme espaces réservés aux arguments de commande ou pour accéder facilement à la fin d'un environnement. Le saut à ces repères, vers l'avant ou vers l'arrière, a été obtenu en utilisant des macros<sup>4</sup>; les macros sautent aux repères, les sélectionnent ou les suppriment. La plupart des abréviations sont inspirées de celles employées dans le système FasT<sub>E</sub>X<sup>5</sup> utilisé avec Typelt4Me<sup>6</sup>.

La version 2.30 et les suivantes de T<sub>E</sub>XShop intègrent une version améliorée du complètement de commande inspirée de Hugh Neary et Will Robertson. Les macros Applescript ne sont plus nécessaires et les arguments des compléments peuvent comporter un bref aide-mémoire.

---

1. Ces caractères sont des espaces, onglets, retour à la ligne (interligne), points, virgule, point-virgule, deux points, {, }, (, ) ou \ (en réalité le caractère de commande de T<sub>E</sub>X qui peut varier dans différentes réalisations). Le { et \ font aussi partie de l'expansion.

2. Souscrite en envoyant un mél à <[MacOSX-TeX-on@email.esm.psu.edu](mailto:MacOSX-TeX-on@email.esm.psu.edu)>.

3. Précédemment appelés *Tags*.

4. Les fichiers originaux `CommandCompletion.txt`, les macros et la documentation sont encore disponibles sous le titre `CommandCompletion.zip` sur le site <<http://homepage.mac.com/herbs2/>>.

5. FasT<sub>E</sub>X a été développé par Filip G. Machi, Jerrold E. Marsden et Wendy G. McKay. Pour de plus amples renseignements, consultez la page Web FasT<sub>E</sub>X, <<http://www.cds.caltech.edu/~fastex/>>.

6. Typelt4Me, par Riccardo Ettore, qui est actuellement un panneau de préférences, permet de remplacer une abréviation dans la plupart des programmes OS X avec « des dictionnaires » qui peuvent dépendre d'une application. Voir la page web Typelt4Me, <<http://www.typeit4me.com/>>, pour plus d'informations.

## Installation

Placer simplement cette version de  $\text{T}\text{E}\text{X}$ Shop dans `/Applications/ $\text{T}\text{E}\text{X}$ /`. Si vous avez déjà utilisé une précédente version de  $\text{T}\text{E}\text{X}$ Shop (antérieure à 2.31) suivez également les instructions de la sous-section suivante.

## CommandCompletion.txt

Par défaut, une nouvelle version de `CommandCompletion.txt` est ancré dans cette nouvelle version de  $\text{T}\text{E}\text{X}$ Shop. Elle sera installée, avec d'autres fichiers, à la première exécution de  $\text{T}\text{E}\text{X}$ Shop, sauf si vous aviez utilisé  $\text{T}\text{E}\text{X}$ Shop auparavant. Dans ce cas, vous devez déplacer le répertoire `~/Library/ $\text{T}\text{E}\text{X}$ Shop/CommandCompletion/` (`~` est votre répertoire personnel) sur votre bureau et démarrer la nouvelle version de  $\text{T}\text{E}\text{X}$ Shop; un dossier de remplacement avec la nouvelle version de `CommandCompletion.txt` sera créé.

Si vous avez déjà ajouté des éléments au dossier original, vous serez en mesure de les fusionner dans le nouveau fichier après sa création. Vous pouvez ouvrir la nouvelle version de `CommandCompletion.txt` en cliquant sur l'élément final du menu `Source` → `Complètement` → `Ouvrir le Fichier "Complètement"...` L'ancien fichier doit être ouvert avec le codage Unicode (UTF-8) : utilisez la commande du menu `Fichier` → `Ouvrir...` ( $\text{⌘}$ -O) et assurez-vous d'avoir sélectionné le format de codage Unicode (UTF-8) avant d'ouvrir votre ancienne version dans  $\text{T}\text{E}\text{X}$ Shop.

## Nouveautés dans $\text{T}\text{E}\text{X}$ Shop

Quatre changements interconnectés à  $\text{T}\text{E}\text{X}$ Shop sont des plus importants : un ajout à la façon dont  $\text{T}\text{E}\text{X}$ Shop traite les compléments à partir du fichier `CommandCompletion.txt` ; un nouveau menu qui a des commandes pour rechercher et sélectionner les *Marks* (Repères) dans les compléments ; la possibilité de joindre des aide-mémoire, aux *Marks* ; un nouveau fichier `CommandCompletion.txt` qui acquiert un certain avantage des trois précédents changements. Les quatre sous-sections suivantes traitent de chacun de ces changements plus en détail. Quelques changements mineures, au niveau du menu, sont examinés séparément.

## Traitement du complètement

Les compléments (dans le fichier `CommandCompletion.txt`) dans les versions précédentes de  $\text{T}\text{E}\text{X}$ Shop ne pouvaient contenir qu'une seule commande `#INS#` pour positionner le point d'insertion en leur sein. Cette version  $\text{T}\text{E}\text{X}$ Shop permet aux compléments d'avoir deux copies de `#INS#` entre lesquelles le texte sera sélectionné. Un seul `#INS#` se comporte de la même façon qu'auparavant ; il y a une totale compatibilité ascendante avec les versions précédentes de  $\text{T}\text{E}\text{X}$ Shop.

## Nouveau menu Source → Complètement → Repères

Le nouveau menu Source → Complètement → Repères contient des commandes pour rechercher, aller au niveau des repères et des aide-mémoire, et les sélectionner. Les commandes sont indiquées dans le tableau 1 page 3. Les versions (Ôter) (*Del*) des commandes de recherche ne sont visibles dans le menu que si la touche option ( $\surd$ ) est appuyée et la commande Insérer un aide-mémoire ne s'affiche que lorsque vous maintenez appuyée la touche Contrôle ( $\wedge$ ). La commande Insérer un repère est ajoutée puisque l'utilisation de la fonction d'auto-complètement de T<sub>E</sub>XShop (raccourci clavier) va insérer un `\bullet` dans le document lorsque la combinaison de touches qui, normalement, insère un « • » ( $\surd$ -8 dans la correspondance du clavier américain) est employée. La version par défaut du menu Repères est montrée dans la figure 1, page 4.

### Aide-mémoire

Les modifications mentionnées dans les sous-sections précédentes permettent aux compléments de contenir des aide-mémoire – petits « pense-bêtes » – qui donnent des indications sur ce que doit contenir un argument donné. Les aide-mémoire, entourés de « •< » et « > »<sup>7</sup>, sont placés à l'intérieur les arguments ; si le premier argument contient un aide-mémoire, celui-ci doit être entouré de deux #INS# afin qu'il soit sélectionné d'abord.

TABLEAU 1 – Commandes du menu Source → Complètement → Repères.

Élément du menu	Raccourci	Connexion interne
Repère suivant	$\wedge$ -⌘-F	Va au prochain repère ( <i>Mark</i> ) ou aide-mémoire et le sélectionne.
Repère suivant (Ôter)	$\wedge$ - $\surd$ -⌘-F	Va au prochain repère ou aide-mémoire, le sélectionne et supprime le repère : c'est surtout utile lorsque vous avez des environnements imbriqués pour supprimer automatiquement un repère situé à la fin d'un environnement interne.
Repère précédent	$\wedge$ -⌘-G	Comme pour Repère suivant mais en sens inverse.
Repère précédent (Ôter)	$\wedge$ - $\surd$ -⌘-G	Comme pour Repère suivant (Ôter) mais en sens inverse.
Insérer un repère	⌘-8	Place un repère au point d'insertion : utilisé pour repérer des arguments, etc., lors de la création de nouveaux compléments dans le fichier CommandCompletion.txt.
Insérer un aide-mémoire	$\wedge$ -⌘-8	Place l'ossature d'un aide-mémoire « •< » avec le point d'insertion situé avant le « > » : pratique pour créer des commentaires dans le fichier CommandCompletion.txt.

7. Notez que « < » et « > » sont des guillemets simples et non les glyphes « < » et « > ».

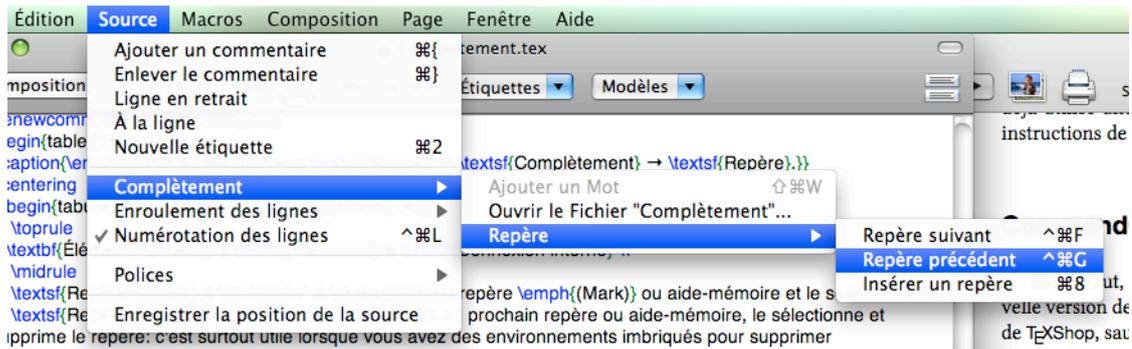


FIGURE 1 – Menu, par défaut, de Source → complètement → Repères.

## Nouveau fichier `CommandCompletion.txt`

Par rapport à la version originale, il y a quelques changements dans le nouveau fichier `CommandCompletion.txt` qui vient avec cette version de TeXShop :

- Il y a quelques corrections, mineures, de bugs, et quelques environnements et commandes supplémentaires ;
- toutes les commandes `#INS#` isolées sont remplacées par des `#INS#•#INS#` afin que le repère initial soit sélectionné, ■. Il s’agit d’un souci de cohérence entre l’apparence et le comportement lorsque vous utilisez les commandes Repère suivant/précédent ;
- il y a un peu (trop peu ?) d’exemples d’aide-mémoire dans les commandes et les environnements.

## Révisions mineures du système de menu

Il y a trois changements mineurs au système de menu :

1. L’élément du menu : Source → Numérotation des lignes est maintenant attribué au raccourci `^⌘L` de telle sorte que la numérotation des lignes peut être rapidement activée ou désactivée. Cela ne change pas le comportement par défaut défini dans les TeXShop → Préférences.
2. Le raccourci clavier `^2` est attribué à Fenêtre → Diviser la fenêtre. Cela fonctionne à la fois dans les fenêtres du Source et de l’Aperçu ; si elles sont actives.
3. Il y a en outre un item « caché », dans le menu Source → Enroulement des lignes « sous » la commande Enrouler, appelé Soft Wrap qui est attribué à `⌘⇧H`. Presser la touche `⌘` lorsque le sous-menu Source → Enroulement des lignes est ouvert. Utilisez cette possibilité avec précaution car elle combinera plusieurs paragraphes en un seul s’ils sont tous sélectionnés, et si vous utilisez cette commande sans sélection, *tous* les sauts de lignes du document seront supprimés.

# Utilisation

## Complètement de commande

Une utilisation typique du complètement de commande est d'installer des environnements. Pour ce faire, tapez `\b` puis  $\varnothing$  ; ce qui devrait vous donner `\begin{`. Ensuite, commencez à taper le nom de l'environnement, par exemple, `eq` puis  $\varnothing$  ce qui donnera :

```
\begin{equation}
```

•

```
\end{equation}•
```

tandis que l' $\varnothing$  suivant donnera `eqnarray` suivi de sa variante étoilée (\*). Après avoir entré le texte de votre équation au niveau du curseur exécutez la commande Source  $\rightarrow$  Complètement  $\rightarrow$  Repères  $\rightarrow$  Repère suivant et le curseur sélectionnera (ou supprimera, si Repère suivant (Ôter) est utilisé) le « • » suivant et vous pourrez commencer à taper le texte qui suit.

Les macros sont également très pratiques pour les commandes qui prennent plusieurs arguments. Par exemple, pour créer une nouvelle commande avec un argument optionnel tapez `\new` or `\newc` puis trois fois  $\varnothing$  pour obtenir :

```
\newcommand{•}[•][•]{•}
```

avec le premier repère sélectionné. Après avoir entré le nom de la nouvelle commande utilisez la commande Repère suivant pour passer à l'argument suivant, etc.

## Abréviations

En plus du complètement de commande, il existe aussi de nombreuses abréviations pour les commandes. La principale différence est que les abréviations ne sont pas seulement le début d'un nom de commande. Par exemple, taper `benu`, puis  $\varnothing$  au début d'une ligne<sup>8</sup> produira l'environnement complet d'une énumération :

```
\begin{enumerate}
```

```
\item
```

•

```
\end{enumerate}•
```

comme vous pouvez vous en douter. De telles abréviations existent pour de nombreux environnements, ainsi que pour des commandes de sectionnement. D'autres versions de commande avec une ou plusieurs options ou des variantes étoilées (\*) ont des noms qui se terminent respectivement par « o » (un ou plusieurs) ou « s » : par exemple, `sec` et deux pressions sur  $\varnothing$  ou `secs` et une seule pression sur  $\varnothing$  au début d'une nouvelle ligne donnent `\section*{•}`. Par ailleurs, après avoir tapé le texte pour le premier item, taper `it` puis  $\varnothing$  sur une nouvelle ligne va générer un autre `\item` avec un repère sélectionné sur la ligne du dessous ; des pressions successives sur la touche  $\varnothing$  donneront `\item[•]` avec un repère sur la ligne suivante, `\textit{•}` et enfin `\itshape` avant de retourner au `it` d'origine.

Vous devez vous souvenir qu'il faut un des caractères de début de mot, avant d'utiliser les abréviations ; sinon, la substitution ne fonctionnera pas correctement. Ce n'est pas un

---

8. Ou après tout autre caractère de début de mot.

problème avec les environnements et les commandes de sectionnement, puisque vous commencez, généralement, sur une nouvelle ligne ; mais, cela peut l'être pour d'autres abréviations. C'est pourquoi de nombreuses abréviations ont également une version avec « \ » ; par exemple, 't t puis  $\varnothing$  ne se développera pas correctement, puisque « ' » n'est pas un caractère de début de mot tandis que '\tt puis  $\varnothing$  donnera '\texttt{•} et qu'une deuxième pression de la touche  $\varnothing$  donnera la déclaration '\ttfamily<sup>9</sup>

Bon nombre des caractères grecs et leurs versions mathématiques, en ligne, ont des abréviations dont les règles sont les suivantes :

1. Les abréviations des caractères grecs commencent toutes par un « x » suivi d'une notation pour les caractères : par exemple, xa ou \xa<sup>10</sup> puis  $\varnothing$  et donnera \alpha.
2. La version var de plusieurs caractères grecs commencent par « xv » suivi de la notation pour le caractère : par exemple, xth donne \theta tandis que xvth puis  $\varnothing$  donne \vartheta.
3. Pour obtenir des lettres capitales utilisez un « xc » : par exemple, xg donne \gamma tandis que xcg donne \Gamma.
4. Enfin, l'abréviation précédée d'un d présente le caractère grec comme dans une équation mathématique en ligne : par exemple, dxc donne \(\Delta\).

Les abréviations seront complétées et bouclées par le biais des correspondances tout comme les compléments de commande : par exemple, l'abréviation newcoo (notez le « oo » à la fin de l'abréviation) suivi d'une pression sur  $\varnothing$  ou newc suivi de trois pressions sur la touche  $\varnothing$ , les deux placées sur une nouvelle ligne, donneront \newcommand{•}[•][•]{•} : le \newcommand avec deux arguments optionnels. Il existe des abréviations de remplacement pour certaines commandes : par exemple, ncm donne le même résultat que newc.

Je vous suggère de lire le fichier CommandCompletion.txt pour voir les abréviations disponibles ; toutes les lignes avec « : = » sont des abréviations. Naturellement, vous pouvez les modifier selon vos besoins, en ajouter ou en supprimer.

## Aide-mémoire

S'il est facile de se souvenir des arguments des commandes utilisées assez souvent, ce n'est pas le cas de celles qui le sont rarement. Ces dernières sont de parfaites candidates pour les aide-mémoire. L'ordre des arguments de la commande \rule en est un exemple ; tapez \rul et deux fois  $\varnothing$  pour obtenir \rule[•<lift>]{•<width>}{•<height>} : la version avec l'argument optionnel<sup>11</sup>. Un autre exemple est l'environnement wrapfigure de l'extension wrapfig, qui dispose de plusieurs versions avec différents numéros et différents positionnements des arguments facultatifs. Pour voir les variations avec les aide-mémoire, tapez bwr sur une ligne vide et appuyez sur  $\varnothing$  pour obtenir :

```
\wrapfigure{•<placement: r,R,l,L,i,I,o,O>}{•<width>}
•
\end{wrapfigure}•
```

9. Des abréviations similaires existent pour les bf, sf, sc, etc. Les versions pour les maths sont précédées par un m, par exemple, mbf puis  $\varnothing$ , donnera \mathbf{•}.

10. Toutes les abréviations des caractères grecs ont une version \.

11. C'est-à-dire, \rule[#INS#•<lift>#INS#]{•<width>}{•<height>} dans le fichier CommandCompletion.txt.

pour obtenir les versions comportant des arguments optionnels appuyez successivement sur  $\mathfrak{D}$ .

## Autres environnements

Les environnements non prédéfinis dans le fichier `CommandCompletion.txt` peuvent, toujours, être ajoutés si vous les utilisez beaucoup, mais il y a une alternative pour une utilisation occasionnelle. Les environnements peuvent être construits à l'intérieur même de l'algorithme de complétement. Tapez d'abord `\b` et appuyez sur  $\mathfrak{D}$  pour obtenir `\begin{`, entrez ensuite le nom d'environnement et l'accolade de fermeture `}`, appuyez de nouveau sur  $\mathfrak{D}$  et la commande de clôture `\end{}` avec le nom correspondant de l'environnement sera générée sur une ligne distincte.

## Abréviations du fichier `CommandCompletion.txt`

Cette section contient, nous l'espérons, la liste complète des abréviations fournies dans `CommandCompletion.txt`. La liste a été partagée entre, les environnements, les commandes et les déclarations, et les lettres grecques. Si vous donnez le début d'une abréviation la recherche va débiter à la première abréviation similaire à votre proposition et à chaque pression de  $\mathfrak{D}$  elle va descendre dans la liste jusqu'à ce qu'il n'y ait plus aucune identité : par exemple, si vous tapez `be` des pressions successives de  $\mathfrak{D}$  correspondront à `benu`, `benuo`, `bequ`, `bequs`, `beqn` et `beqns` avant de retourner à l'original `be`. Ajouter plus de lettres à l'abréviation peut vous permettre d'arriver au résultat escompté en utilisant moins souvent la touche  $\mathfrak{D}$ . En réalité, certaines des commandes et déclarations sont éparpillées entre les environnements dans le fichier `CommandCompletion.txt` si bien qu'il pourrait y avoir, parfois, des éléments en plus. Les tableaux ne présentent pas les compléments standards ni les versions « `\` » des abréviations.

REMARQUE. – La liste peut sembler un peu intimidante. Il n'est pas nécessaire de « mémoriser » toutes ces abréviations ; apprenez-en un nombre minimum, celles dont vous avez besoin, et utilisez la touche  $\mathfrak{D}$  pour obtenir les variations.

## Abréviations des environnements

Le tableau 2, page 9, contient une liste des abréviations, pour différents environnements, fournies dans le fichier `CommandCompletion.txt`. Plusieurs environnements voisins verticalement portent le même nom. Ils correspondent à des variations dans le nombre et la répartition des arguments optionnels possibles ou à des variantes étoilées (\*). Il se peut qu'il y ait, aussi, plus d'une abréviation pour le même environnement.

## Commandes et déclarations

Comme pour les environnements il y a beaucoup de variations, en options et variantes étoilées (\*), ainsi que de nombreuses abréviations correspondant à la même commande : voir le tableau 3, page 10.

## Lettres grecques

Les abréviations des lettres grecques apparaissent dans le tableau 4, page 11. Les versions destinées aux équations en ligne, c'est-à-dire, celles précédées d'un « d », ne sont pas présentées.

## Ajouts au fichier **CommandCompletion.txt**

Si vous ajoutez des éléments au fichier `CommandCompletion.txt`, il y a certaines choses à connaître concernant sa structure :

- Chaque environnement possède trois entrées : un complètement qui supprime le `\begin` de tête, c'est-à-dire, qu'il commence avec une accolade « { » et le nom de l'environnement ; deux abréviations qui ont un nom d'abréviation sans barre oblique inverse (`\`) et l'abréviation même avec la barre oblique inverse. Les commandes peuvent avoir plusieurs formes ; la commande intégrale ainsi qu'une ou plusieurs abréviations, toutes avec ou sans la barre oblique inverse `\`.
- Pour les abréviations, vous devez ajouter toutes les variantes de terminaisons légèrement différentes. J'utilise un « o », à la fin d'une abréviation, pour un argument optionnel ; un « oo » pour deux arguments facultatifs ; un « s » pour des commandes de formes étoilées, etc.
- L'ordre d'éléments similaires, dans le fichier, ne fait pas une différence spectaculaire dans l'ordre dans lequel ils sont trouvés : ceux placés après, seront trouvés plus tôt (l'exploitation du fichier est ascendante). Par exemple, l'ordre des éléments obtenus lorsque vous tapez `\b` puis  $\mathfrak{S}$  dépend purement de l'ordre des correspondances dans le fichier `CommandCompletion.txt`.
- Pour un maximum de confort, placez un repère <sup>12</sup> dans chacun des arguments des commandes. Entourez le tout premier argument avec deux commandes `#INS#` afin qu'il en ressorte sélectionné. Si vous voulez avoir un aide-mémoire dans tous les arguments, insérez la structure nécessaire et remplissez-la.

Je suggère que vous jetiez un œil dans le fichier `CommandCompletion.txt` pour des exemples.

## Bogues

Le fichier `CommandCompletion.txt` est habituellement exploité de façon ascendante, à partir du dernier élément ; mais, dans de rares cas, le sens de la recherche semble s'inverser, si bien que vous n'obtenez pas les correspondances dans l'ordre que vous attendiez. Vous pouvez, généralement, forcer l'exploitation à revenir dans la « bonne » direction, en tapant `---` et en appuyant trois fois sur  $\mathfrak{S}$ , et ensuite en enlevant le `---`. Si cela ne corrige pas le sens de la recherche, vous pouvez utiliser le raccourci `↑- $\mathfrak{S}$`  pour chercher dans l'« autre » direction.

---

12. En utilisant Insérer un repère ( $\mathfrak{S}$ -8) du menu Source → Complètement → Repères.

## Ce qui manque

J'aimerais que les compléments puissent respecter les renforcements (retraits), mais ce n'est pas dans les tablettes pour l'instant. D'autres suggestions sont bienvenues et seront prises en considération afin d'être incluses dans les itérations ultérieures du code de la Commande de complètement.

Essayez la... J'espère que vous l'aimerez.

---

Traduit par René Fritz, 20 février 2010.

TABLEAU 2 – *Abréviations des environnements dans CommandCompletion.txt.*

<b>Abbreviation</b>	<b>Environment</b>	<b>Abbreviation</b>	<b>Environment</b>
barr	array	blett	letter
babs	abstract	blist	list
bali	align	bminp	minipage
balis	align*	bminpo	minipage
baliat	alignat	bmult	multiline
baliats	alignat*	bmults	multiline*
balied	aligned	bpict	picture
baliedat	alignedat	bpmat	pmatrix
baliedato	alignedat	bquot	quotation
bapp	appendix	bquo	quote
bbmat	bmatrix	bsplit	split
bcase	cases	bsubeq	subequations
bcent	center	btabs	tabular
bcenum	compactenum	btabs	tabular*
bcenumo	compactenum	btabsx	tabularx
bcitem	compactitem	btabl	table
bcitemo	compactitem	btablo	table
bdes	description	btabsl	table*
benu	enumerate	btabslo	table*
benuo	enumerate	btbl	table
bequ	equation	btblo	table
bequs	equation*	btbls	table*
beqn	eqnarray	btblso	table*
beqns	eqnarray*	btabb	tabbing
bfig	figure	bbib	thebibliography
bfigo	figure	bindex	theindex
bframe	frame	btheo	theorem
bframeo	frame	btitpg	titlepage
bflalign	flalign	btrivl	trivlist
bflaligns	flalign*	bvarw	varwidth
bfl	flushleft	bverb	verbatim
bflr	flushright	bvers	verse
bgath	gather	bwrap	wrapfigure
bgaths	gather*	bwrapo	wrapfigure
bgathed	gathered	bwrapo2	wrapfigure
bgathedo	gathered	bwrapoo	wrapfigure
bite	itemize		
biteo	itemize		

TABLEAU 3 – *Commandes et déclarations dans CommandCompletion.txt.*

Abbreviation	Command	Abbreviation	Command	Abbreviation	Command
--	textendash	midr	midrule	renewcomo	renewcommand
---	textemdash	mnorm	mathnormal	renewcomoo	renewcommand
---	textemdash w/sp	msf	mathsf	rncm	renewcommand
adlen	addtolength	mtt	mathtt	rnewc	renewcommand
adcount	addtocounter	mit	mathit	rncmo	renewcommand
bf	textbf	midr	midrule	rnewcoo	renewcommand
bfd	bfseries	mnorm	mathnormal	rncmoo	renewcommand
biblio	bibliography	mdd	mdseries	rnc	rmfamily
bibstyle	bibliographystyle	mbox	mbox	rbox	raisebox
botr	bottomrule	makebox	makebox	rboxo	raisebox
bibitem	bibitem	mboxo	makebox	rboxoo	raisebox
bibitemo	bibitem	makebox	makebox	sec	section
center	centering	mboxoo	makebox	secs	section*
chap	chapter	mpar	marginpar	seco	section
cmidr	cmidrule	multic	multicolumn	ssec	subsection
cmidro	cmidrule	ncol	space & space	ssecs	subsection*
em	emph	ncm	newcommand	sseco	subsection
emd	em	newc	newcommand	sssec	subsubsection
foot	footnote	ncmo	newcommand	sssecs	subsubsection*
frac	frac	newco	newcommand	ssseco	subsubsection
fbox	fbox	ncmoo	newcommand	spar	subparagraph
fboxo	framebox	newcoo	newcommand	spars	subparagraph*
fboxoo	framebox	nct	newcolumnntype	sparo	subparagraph
geometry	geometry	newct	newcolumnntype	setl	setlength
hw	headwidth	newpg	newpage	stcount	stepcounter
hw2tw	headw=tw	npg	newpage	sf	textsf
href	href	nline	newline	sfd	fffamily
item	item	newlin	newline	sc	textsc
ito	item	nlen	newlength	scd	scshape
incg	includegraphics	newlen	newlength	sl	textsl
incgo	includegraphics	nenv	newenvironment	sld	slshape
it	textit	newenv	newenvironment	sqrt	sqrt
itd	itshape	nenvo	newenvironment	sqto	sqrt
latex	LaTeX	newenvo	newenvironment	tt	texttt
latexs	LaTeX w/sp	nenvoo	newenvironment	ttd	ttfamily
latexe	LaTeXe	newenvoo	newenvironment	tw	textwidth
latexes	LaTeXe w/sp	pgref	pageref	tex	TeX
label	label	par	paragraph	texs	TeX w/sp
lbl	label	pars	paragraph*	tilde	textasciitilde
lettrine	lettrine	paro	paragraph	topr	toprule
lettrineo	lettrine	pgs	pagestyle	toc	tableofcontents
listf	listoffigures	parbox	parbox	tableofcontents	tableofcontents
listt	listoftables	parboxo	parbox	tpgs	thispagestyle
rule	rule	parboxoo	parbox	thispagestyle	thispagestyle
ruleo	rule	parboxooo	parbox	up	textup
mbf	mathbf	pbox	parbox	upd	upshape
mrm	mathrm	pboxo	parbox	url	url
mcal	mathcal	pboxoo	parbox	usep	usepackage
msf	mathsf	pboxooo	parbox	usepo	usepackage
mtt	mathtt	ref	ref	verb	verb
mit	mathit	renewcom	renewcommand	verb2	verb

TABLEAU 4 – *Lettres grecques dans CommandCompletion.txt (version d non présentée).*

<u>Abbreviation</u>	<u>Command</u>	<u>Abbreviation</u>	<u>Command</u>
xa	alpha	xph	phi
xb	beta	xcph	Phi
xch	chi	xvph	varphi
xd	delta	xps	psi
xcd	Delta	xcps	Psi
xe	epsilon	xs	sigma
xve	varepsilon	xcs	Sigma
xet	eta	xvs	varsigma
xg	gamma	xz	zeta
xcg	Gamma	xr	rho
xio	iota	xvr	varrho
xl	lambda	xt	tau
xcl	Lambda	xth	theta
xm	mu	xcth	Theta
xn	nu	xvth	vartheta
xo	omega	xu	upsilon
xco	Omega	xcu	Upsilon
xp	pi	xx	xi
xcp	Pi	xcx	Xi
xvp	varpi		