

# De pdfLaTeX à LuaLaTeX

Daniel Flipo

18 avril 2025

Passer de pdfLaTeX à LuaLaTeX est simplissime : rien à changer dans le corps du document, la seule chose à modifier dans le préambule est l'appel des fontes.

Sauf à lancer la compilation en ligne de commande, il faudra indiquer à votre éditeur que la compilation doit être faite en LuaLaTeX plutôt qu'en pdfLaTeX. Comment ? une petite recherche dans les menus devrait donner la réponse. Des solutions sont proposées en annexe section [B p. 15](#) pour quelques éditeurs courants.

LuaLaTeX impose que le code source soit codé en UTF-8 qui est d'ailleurs le codage par défaut des documents pdfLaTeX récents. Si une conversion de codage est nécessaire, elle peut se faire sous un « bon » éditeur (Emacs par exemple) ou avec un utilitaire du genre `recode` ou `i conv`, etc. Pour plus de détails sur ces conversions consulter l'annexe section [A p. 15](#).

Ce court tutoriel, vise à faciliter la transition. Une fois le pas franchi, la lecture des documents proposés [p. 14](#) s'imposera pour aller plus loin.

## 1 Pourquoi passer à LuaLaTeX ?

pdfLaTeX utilise en général des fontes « type 1 » codées par plans de 256 caractères maximum. Outre le fait que ce format a été abandonné par les éditeurs de fontes au profit du format « OpenType<sup>1</sup> » (codage Unicode par plans de 65536 caractères maximum), les fontes « type 1 » nécessitent pour fonctionner avec pdfLaTeX une kyrielle de fichiers auxiliaires (`.map`, `.fd`, `.tfm`, `.vf`) qu'il faut créer (et placer « aux bons endroits » de l'arborescence TEXMF) si on veut utiliser une fonte non disponible dans la distribution TeX utilisée.

Le gros avantage de LuaLaTeX est de donner immédiatement accès à toutes les fontes système de la machine (Cambria sous Windows, Hoefler Text sur Mac, etc.) en plus des

---

1. Ce format regroupe les fontes de type TrueType (suffixe `.ttf`) et les fontes CFF (suffixe `.otf`). Il représente un réel progrès au niveau des possibilités offertes (variantes contextuelles, etc.).

fontes incluses dans la distribution TeX. Mais surtout, toute fonte achetée ou trouvée sur les sites spécialisés est immédiatement utilisable, il suffit de la placer dans TEXMF-HOME/fonts/opentype<sup>2</sup> (pour un usage personnel) ou TEXMFLOCAL/fonts/opentype (pour la rendre accessible à tous les utilisateurs) mais dans ce dernier cas il faudra mettre à jour la base de données (commande `mktexlsr` ou équivalent).

Autre avantage pour nous francophones, LuaLaTeX permet de gérer la ponctuation haute sans recours aux caractères actifs, source de multiples conflits et incompatibilités en pdfLaTeX. De plus, le langage de script intégré (lua) permet une gestion plus fine du deux-points : pas d'espace parasite dans les ratios (1:1) ou les URL (`http://mon-site.fr`) ceci sans intervention de l'utilisateur, ce qui n'est pas le cas avec pdfLaTeX ou XeLaTeX. Enfin, LuaLaTeX permet des raffinements supplémentaires : guillemets de second rang en début de ligne, affichage des défauts typographiques (veuves, orphelines, voir `lua-typo.sty`).

## 2 Choix des fontes

### 2.1 Par défaut : « Latin Modern »

Il suffit de remplacer dans le préambule *toutes les commandes relatives aux fontes*

```
\usepackage[utf8]{inputenc} % codage par défaut pour LaTeX
\usepackage[T1]{fontenc}    % ou [T2A,T1]
\usepackage{textcomp}      % si présent
\usepackage{lmodern}       % choix de la fonte.
```

par

```
\usepackage{fontspec}
```

ou, si on veut utiliser aussi des fontes OpenType pour les maths<sup>3</sup> (recommandé) par

```
\usepackage{unicode-math}
```

Si on ne fait que ça, le document compilé en LuaLaTeX utilisera les fontes OpenType « Latin Modern » pour le texte et les maths et aura donc un aspect quasi-identique<sup>4</sup> au document initial compilé en pdfLaTeX avec le préambule précédent.

L'encadré de la page suivante montre un exemple de document minimal (avec des maths) pour LuaLaTeX.

La *seule* commande impliquant les fontes est en rouge. Le résultat de sa compilation en LuaLaTeX est visible *ici* (premier cadre). Une remarque toutefois : le résultat présenté

---

2. Ou même dans le répertoire courant pour un usage très ponctuel...

3. Noter que `unicode-math` charge automatiquement `fontspec`.

4. Les métriques des fontes type 1 utilisées par pdfLaTeX peuvent présenter de légères différences avec leurs homologues OpenType.

est obtenu en ajoutant la ligne

```
\setmainfont{Latin Modern Roman}[SmallCapsFont=lmromancaps10-regular]
```

ce qui permet d'avoir à la fois les chiffres elzéviens et de vraies petites capitales (elles sont fournies dans un fichier distinct il faut en spécifier le nom). Si on charge la fonte OpenTypes sans l'option – la fonte *n'a pas* de petites capitales intégrées – LaTeX signale cette absence dans le fichier .log :

```
LaTeX Font Warning: Font shape `TU/LatinModernRoman(0)/m/sc' undefined
(Font)                using `TU/LatinModernRoman(0)/m/n' instead ...
```

et les remplace par des minuscules normales...

```
\documentclass[a5paper,english,french]{article}
\usepackage{babel}\frenchsetup{og=«, fg=»}
\usepackage{unicode-math}
\begin{document}
Si les «bons» guillemets, l'euro €, les lettres grecques
ne peuvent pas être saisis \emph{directement} au clavier,
il reste la possibilité du codage traditionnel \og, \fg,
\texteuro, \zeta, \Gamma...

$$[\zeta(z)= \sum_{n=1}^{\infty} \frac{1}{n^z} \\
= \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t-1} dt]$$

\textbf{Du gras}, \textit{\textbf{du gras italique}}.\par
Chiffres standard 0123456789 ou elzéviens
\oldstylenums{0123456789} ?
\end{document}
```

Notez que les lettres grecques  $\zeta$  et  $\Gamma$  ont été saisies littéralement<sup>5</sup> dans la formule mathématique, ce qui en facilite la lecture.

## 2.2 Une alternative : « New Computer Modern »

« New Computer Modern » est un autre avatar des polices « Computer Modern » de Knuth qui peut avantageusement remplacer les « Latin Modern » : les polices de texte disposent des petites capitales et des chiffres elzéviens, elles sont en outre enrichies des alphabets cyrillique, grec, hébreu et phonétique (IPA), la police mathématique est Unicode complète.

Enfin une variante plus grasse –Book complète les –Regular. C'est cette variante –Book qui est présentée [ici](#) (cadre 3). Elle se charge de la façon suivante :

---

5. Pour Emacs la méthode est indiquée en annexe, à la fin de la section [B.4](#) p. 16

```
\usepackage{newcomputermodern} % variante -Book
%\usepackage[regular]{newcomputermodern} % variante -Regular
```

## 2.3 Choisir d'autres fontes

Avec pdfLaTeX changer les fontes de base revient le plus souvent à charger une extension : `fourier`, `kpfonts`, `txfonts`, etc. La même chose est possible avec LuaLaTeX mais pas toujours...

Voyons comment choisir d'autres fontes que celles chargées par défaut par `fontspec`. La référence est le manuel de `fontspec` [3], celui-ci étant un peu touffu voici quelques pistes pour débayer le terrain.

À chacune famille, romaine (`\rmfamily`), sans empattements (`\sf family`) et à chasse fixe (`\ttfamily`), correspond une commande de `fontspec` :

```
\setmainfont{ nom }[ options ] % fonte romaine
\setsansfont{ nom }[ options ] % fonte sans empattements
\setmonofont{ nom }[ options ] % fonte à chasse fixe
```

Pour les fontes mathématiques, `unicode-math` définit la commande

```
\setmathfont{ nom }[ options ]
```

Noter que, contrairement à `\usepackage[...]{...}` les options se placent *après* le nom et que si une ou plusieurs de ces commandes sont omises, la fonte par défaut correspondante est utilisée (Latin Modern Roman, Sans ou Mono).

### 2.3.1 Quelques exemples

Si avec pdfLaTeX les fontes de type Computer Modern vous donnent satisfaction vous pouvez vous limiter à ce qui est proposé à la section 2.1 et sauter le reste de la section 2.

Les exemples qui suivent montrent comment composer le document minimal ci-dessus avec d'autres fontes, toutes OpenType pour le texte comme pour les maths; elles sont libres et font toutes partie de la distribution TeXLive.

Pour chacun d'eux, il faudra remplacer, dans le code de la page 3, la ligne

```
\usepackage{unicode-math}
```

par la ou les lignes en rouge. Pour voir le résultat de leur compilation, cliquer sur le nom de fonte (en vert).

1. Sous pdfLaTeX vous aimez faire appel à `fourier`? la transition est facile : dans l'exemple précédent il suffit de remplacer la ligne

```
\usepackage{unicode-math}
```

par

```
\usepackage{fourier-otf}.
```

En effet, `fourier-otf` (cadre 2) charge lui-même `unicode-math` et donc `fontspec`.

`fourier-otf` fait appel pour le texte à la fonte Erewhon créée par Michael Sharpe à partir de la police Utopia. Erewhon dispose de vraies petites capitales et des chiffres

elzéviens, présents uniquement dans les versions commerciales d’Utopia (option expert de fourier). Pour les maths, `fourier-otf` fait appel à la fonte Erewhon-Math<sup>6</sup>, version OpenType des polices mathématiques fourier créées par Michel Bovani.

Ni `fourier` ni `fourier-otf` ne modifient les fontes sans empattement ou à chasse fixe : dans les deux cas les fontes Latin Modern (Sans et Mono) sont utilisées par défaut.

Les options de `fourier-otf` diffèrent de celles de `fourier`, il faudra consulter la documentation, fichier `Erewhon-Math.pdf` de votre distribution TeX.

2. Sous pdfLaTeX vous utilisez Times ou Palatino ? essayez [Termes](#) (cadre 4) ou [Pagella](#) (cadre 5) :

```
\usepackage{unicode-math}
\setmainfont{TeX Gyre Termes X}
\setsansfont{TeX Gyre Heros}
\setmonofont{TeX Gyre Cursor}
\setmathfont{TeX Gyre Termes Math}
```

pour Times ; remplacez « Termes » par « Pagella » pour Palatino.

Pour charger les polices TeX Gyre OpenType *de texte*, il existe des extensions (dus à Herbert Voß), par exemple `\usepackage[...]{termes-otf}` pour TeX Gyre Termes, ou `\usepackage[...]{pagella-otf}` pour TeX Gyre Pagella, consulter la documentation pour avoir la liste des options disponibles.

La commande `\setmathfont{TeX Gyre ... Math}` reste nécessaire pour charger la police math associée.

Notez l’existence, pour accompagner la police de texte Pagella de la police [Asana Math](#) (cadre 6) qui est Unicode complète.

3. L’équipe TeX Gyre propose également des avatars de Bookman et New Century Schoolbook [Bonum](#) et [Schola](#) (cadres 1 et 2) :

```
\usepackage{unicode-math}
\setmainfont{TeX Gyre Bonum}
\setmathfont{TeX Gyre Bonum Math}
```

pour Bookman ; remplacez « Bonum » par « Schola » pour New Century Schoolbook.

Pour charger la police de texte TeX Gyre Schola, il existe une extension `schola-otf` : `\usepackage[...]{schola-otf}`, consulter la documentation pour avoir la liste des options disponibles. La commande `\setmathfont{TeX Gyre Schola Math}` reste nécessaire pour charger la police math associée.

4. Vous avez besoin d’écrire en grec ou en russe de temps en temps ? essayez Linux Libertine, ou plutôt son avatar [Libertinus](#) (cadre 3) qui se charge en une ligne :

```
\usepackage{libertinus-otf}
```

---

6. Par rapport à Utopia, Erewhon comme Erewhon-Math sont réduites à 94% contre 92% pour `fourier`.

ou bien **EBGaramond** (cadre 4) :

```
\usepackage{ebgaramond}
\usepackage{unicode-math}
\setmathfont{Garamond-Math}
```

L'une comme l'autre disposent d'options pour accéder aux les ligatures anciennes et et ainsi qu'au Q-long, voir section 3 p. 11.

5. Sous pdfLaTeX vous utilisez les KPfonts (`\usepackage{kpfonts}`)? elles sont maintenant disponibles aussi en OpenType, essayez **KPfonts** (cadre 3) :

```
\usepackage{kpfonts-otf}
```

De multiples options sont proposées (variantes « light », ligatures anciennes, Q-long, etc.). Consulter la documentation en français `Kpfonts-OTF-Doc-French.pdf` jointe à la distribution.

6. Sous pdfLaTeX vous utilisez les fontes Vera de Bitstream (`\usepackage{bera}`)? essayez les fontes **DejaVu** (cadre 4) qui en dérivent mais sachez qu'elles n'ont ni petites capitales ni chiffres elzéviens et que la police mathématique chasse plus que les autres :

```
\usepackage{unicode-math}
\setmainfont{DejaVu Serif}
\setsansfont{DejaVu Sans}
\setmonofont{DejaVu Sans Mono}
\setmathfont{DejaVu Math TeX Gyre}
```

7. Vous aimez les fontes Bitstream Charter? essayez **XCharter** (cadre 5) :

```
\usepackage{unicode-math}
\setmainfont{XCharter}
\setmonofont{Latin Modern Mono}[Scale = MatchLowercase]
\setmathfont{XCharter Math}
```

8. Vous aimez les polices du début xx<sup>e</sup> siècle? essayez **OldStandard** (cadre 6) :

```
\usepackage{unicode-math}
\setmainfont{OldStandard}
\setmonofont{Latin Modern Mono}[Scale = MatchLowercase]
\setmathfont{OldStandard Math}
```

9. Vous avez besoin de symboles mathématiques exotiques et vous n'aimez pas New Computer Modern? la fonte math **STIX Two** (cadre 5) est également Unicode complète, à charger comme ceci :

```
\usepackage{unicode-math}
\setmainfont{STIX Two Text}
\setmathfont{STIX Two Math}
```

Un autre, presque aussi complète, [Cambria Math](#), (cadre 6) est fournie avec Windows mais elle n'est pas libre... on peut la charger comme ceci :

```
\usepackage{unicode-math}
\setmainfont{Cambria}
\setmathfont{Cambria Math}
```

10. La police mathématique Concrete-Math est maintenant disponible en OpenType, l'extension [concmath-otf](#) (cadre 2) vise à remplacer `concmath` (police math Concrete en Metafont); le chargement se fait comme ceci :

```
\usepackage{concmath-otf}
```

11. Enfin, le couple Concrete / AMS-Euler existe en OpenType; Concrete fait partie des `cm-unicode`, Euler est maintenant dans TeXLive; le couple [Concrete & Euler](#) (cadre 1) est chargé comme ceci :

```
\usepackage{unicode-math}
\setmainfont{CMU Concrete}
\setmathfont{Euler Math}
```

On peut aussi remplacer Concrete par Mint Spirit comme ceci :

```
\setmainfont{Mint Spirit}, voir MintSpirit & Euler (cadre 1).
```

12. Vous aimez les fontes sans empattements? essayez [Noto](#) (cadre 1), notez qu'il vous faudra récupérer la version 3 de NotoSansMath [ici](#).

```
\setmainfont{NotoSans}[%
  Extension = .ttf,
  UprightFont = *-Regular,
  BoldFont = *-Bold,
  ItalicFont = *-Italic,
  BoldItalicFont = *-BoldItalic
]
\setmonofont{NotoSansMono-Regular.ttf}
\setmathfont{NotoSansMath-Regular.otf}
```

ou [Lato](#) (cadre 2)

```
\usepackage{lete-sans-math}
\setmainfont{Lato}[%
  Extension = .ttf,
  UprightFont = *-Regular,
  BoldFont = *-Bold,
  ItalicFont = *-Italic,
  BoldItalicFont = *-BoldItalic ]
\setmonofont{Fira Mono}[Scale = MatchLowercase]
```

ou **Fira** (cadre 3) :

```
\usepackage{unicode-math}
\setmainfont{Fira Sans}
\setmonofont{Fira Mono}[Scale = MatchLowercase]
\setmathfont{Fira Math}
```

ou les KPfonts sans-serif **KP Sans** (cadre 4) :

```
\usepackage[]{}{kpfonts-otf}
```

Voir aussi **Arsenal** (cadre 5), **GFSNeohelleni** (cadre 6) et **MintSpirit** (cadre 1).

13. Les polices **Luciole** (cadre 2) ont été spécialement conçues pour les malvoyants, voir le site dédié [ici](#), elles sont désormais disponibles sur CTAN et dans la distribution TeXLive 2025 :

```
\setmainfont{luciole}
\usepackage{luciole-math}
```

Des exemples plus complets sont donnés (en anglais) ici :

<https://tex.stackexchange.com/a/425099/18401>

Voir également la présentation d’Ulrik Vieth à TUG-2023[5].

**Remarque** : pour les documents sans maths, il est possible de remplacer

`\usepackage{unicode-math}` par `\usepackage{fontspec}`  
en supprimant évidemment la ligne `\setmathfont{...}`.

Le choix de fontes de texte est bien plus large si on n’impose pas l’existence d’une fonte mathématique associée : un catalogue présentant une centaine de fontes OpenType est disponible [ici](#).

D’autre part les polices mathématiques présentées ci-dessus peuvent aussi se marier avec d’autres polices de texte... à essayer donc.

### 2.3.2 Notion de « famille » de fontes

Les commandes `\setmainfont{...}`, `\setsansfont{...}` et `\setmonofont{...}` permettent en fait de choisir une « famille » de fontes c’est-à-dire une police de base (romaine, sans-serif ou à chasse fixe) *et ses variantes italique, grasse et grasse italique* ; ceci est indispensable si on veut qu’une commande `\textbf{bla bla}` ou `\textit{toto}` affiche bien le gras ou l’italique correspondant à la police de base.

La plupart du temps (c’est le cas dans tous les exemples présentés ci-dessus), l’utilisateur n’a rien à spécifier car le créateur de la fonte a prévu un fichier de suffixe `.fontspec`, par exemple `XCharter.fontspec` pour la police `XCharter`<sup>7</sup>, et la distribution (TeXLive, MikTeX, etc.) a placé ce fichier dans un répertoire parcouru par LuaTeX.

---

7. Pour la famille Erewhon, le fichier s’appelle `erewhon.fontspec`, sur les systèmes sensibles à la casse comme Linux, il faut coder `\setmainfont{erewhon}` et non `\setmainfont{Erewhon}`.



Si ce n'est pas le cas, il faut préciser les polices à utiliser pour le gras, l'italique etc. en argument optionnel de la commande `\setmainfont{...}[...]`, voir la documentation de fontspec [3] à ce sujet. Autre solution : créer un fichier `<mafonte>.fontspec` en s'inspirant de ceux qui existent, il suffit ensuite de coder `\setmainfont{<mafonte>}`.

Lorsqu'une famille de fontes est chargée grâce à une commande `\usepackage{...}` c'est l'extension qui s'occupe de tout.

### 2.3.3 Comment nommer la famille de fonte choisie ?

Dans les exemples ci-dessus, nous avons utilisé systématiquement le nom de *famille* : « TeX Gyre Termes X », « DejaVu Sans Mono », « Fira Sans », etc. Notez que ces noms peuvent aussi s'écrire sans espace : « TeXGyreTermesX » pour « TeX Gyre Termes X » par exemple.

C'est la solution la plus simple. Sachez toutefois que fontspec propose deux autres méthodes : appel par nom de fichier ou usage d'un fichier de configuration .fontspec, ces méthodes, décrites dans le manuel de fontspec [3] ne seront pas abordées ici.

Mais où trouver ces noms de *famille* ? Quelles sont les fontes disponibles ?

**Les fontes OpenType de la distribution TeXLive** sont regroupées dans deux répertoires TEXMF/fonts/truetype et TEXMF/fonts/opentype<sup>8</sup>.

Les commandes – à adapter à l'installation (année notamment) –

```
ls -R /usr/local/texlive/2020/texmf-dist/fonts/truetype/ et
```

```
ls -R /usr/local/texlive/2020/texmf-dist/fonts/opentype/
```

permettent de les lister à l'écran. Ensuite pour chaque fichier .ttf ou .otf trouvé `otfinfo -a `kpsewhich PlayfairDisplay-Regular.otf`` affiche<sup>9</sup> le nom de la famille : « Playfair Display » dans cet exemple.

**Les fontes système** là tout dépend du système d'exploitation :

- sous Linux ou depuis le terminal en MacOSX, la commande `fc-list | sort` donne la liste complète de toutes les polices présentes, quel que soit le format (otf, ttf, pfb, pfa, etc.), triées par leur chemin d'accès. La liste est longue (plus de 2 500 lignes chez moi!), on peut sélectionner un type par son suffixe et envoyer le résultat dans un fichier, par exemple :

```
fc-list | sort | grep .ttf > TTF.lst
```

Dans chaque ligne, le premier champ (avant le :) donne le nom complet du fichier, le suivant (terminé par une virgule) donne le nom de la famille.

- sous MacOSX, les fontes système sont sous Library/Fonts, l'application *Font Book* (*Livre des polices* en français) en fournit la liste par famille, on y accède par `finder/Applications`;

---

8. En fait, les fontes TrueType font partie des OpenType. Ce qui se trouve sous le répertoire dit opentype sont des fontes CFF; elles se distinguent des TrueType par le type de courbes de Bézier utilisé : degré deux pour les TrueType, degré trois pour les CFF.

9. Ne pas oublier les accents graves!

— sous Windows, les fontes système sont sous `C:\Windows\Fonts`; pour obtenir le nom de famille à partir du nom complet du fichier on peut utiliser `otfinfo` :

```
otfinfo -a C:\Windows\Fonts\chemin\nom_fichier
```

### 2.3.4 Quelles options ?

Les options appelées *font features* en anglais, permettent de spécifier la couleur, de choisir des variantes (les chiffres elzéviens, des ligatures particulières, etc.).

La documentation de `fontspec` [3] en donne une description détaillée, nous limiterons à quelques exemples basiques.

— `fontspec` charge un fichier de configuration `fontspec.cfg` qui active des options de base, notamment `Ligatures=TeX` pour les fontes chargées par `\setmainfont` et `\setsansfont`; ainsi, `--`, `---`, `!``, `?`` produiront respectivement `—`, `—`, `¡`, `¿`, l’apostrophe et les *quotes ASCII* `'`, ```, `"`, `'` et ``` seront bien converties en `'` (U+2019), `'` (U+2018), `“` (U+201D), `“` (U+201D) et `”` (U+201C).

— Si la fonte choisie le permet, l’option `Numbers=Lowercase` assure que les chiffres minuscules (ou elzéviens) `o123456789` seront utilisés par défaut. Son contraire est `Numbers=Lining` (option par défaut). Quelle que soit l’option choisie, on accède aux deux variantes par `\oldstylenums{2,97}` (2,97) ou `\liningnums{2,97}` (2,97).

— Lorsqu’on doit associer des fontes romaines, sans empattements ou à chasse fixe, d’origines différentes, une option très utile est `Scale=<decimal>` qui permet la mise à l’échelle, par exemple `Scale=1.023` ou `Scale=0.96`.

Une option très utilisée pour les fontes « secondaires » est `Scale=MatchLowercase` qui adapte la taille des fontes sans empattements ou à chasse fixe de façon à faire coïncider l’œil de ces fontes (hauteur du caractère « x ») avec celui de la fonte de base définie par `\setmainfont{}`. Il existe aussi l’option `Scale=MatchUppercase` qui aligne les hauteurs des capitales.

— Il est possible de pallier l’absence d’une police (italique ou penchée) en inclinant la police droite associée : la police à chasse fixe `Inconsolatazi4` (présente dans `TeXLive`) existe en normal et en gras mais n’a ni version italique ni version grasse italique, on les simule donc en inclinant les versions droites comme ceci :

```
\setmonofont{Inconsolatazi4}%
[ItalicFont = *-Regular, ItalicFeatures={FakeSlant=0.225},
 BoldItalicFont = *-Bold, BoldItalicFeatures={FakeSlant=0.225},
 Scale=MatchLowercase
]
```

Il est logique de choisir comme angle d’inclinaison celui des variantes italiques de la police de base, ici  $13^\circ$  ( $0.225 = \sin 13^\circ$ ) ce qui convient pour accompagner `Erewhon`.

### 3 Changement local d'attributs

Les options – aussi appelées attributs –, choisies au chargement d'une fonte sont valables pour tout le document. Il arrive qu'on souhaite les modifier localement, il existe pour cela une commande `\addfontfeatures{}` à utiliser dans un groupe pour en limiter la portée. En voici un exemple :

La fonte Libertinus (exemple 4 p. 5) propose les options `Ligatures=Historic` (pour les ligatures ct et st) et `Alternate=0` (variantes pour les caractères Q, h et &).

Quel accueil chez Christine & Octave!  
Quel accueil chez Christine & Octave!  
Quel accueil chez Christine & Octave!  
Quel accueil chez Christine & Octave!

```
Quel accueil chez Christine \& Octave!  
{\addfontfeatures{Alternate=0}}  
Quel accueil chez Christine \& Octave!  
{\addfontfeatures{Ligatures=Historic}}  
Quel accueil chez Christine \& Octave!  
{\addfontfeatures{Ligatures=Historic,  
                  Alternate=0}}  
Quel accueil chez Christine \& Octave!
```

Pour EBGaramond, les options à utiliser sont `RawFeature=+dlig` pour les ligatures ct et st et `Style=Swash` pour le Q-long.

Avec LuaLaTeX il est très facile de changer localement de police, par exemple pour un titre, une lettrine, etc. De nombreux exemples sont donnés dans [2].

### 4 Pour démarrer avec unicode-math

Il faut savoir que `unicode-math` charge `amsmath`, ce qui impose de charger `amsmath` (ou `mathtools`) *avant* `unicode-math` si on veut pouvoir leur passer des options.

Certaines extensions comme `fourier` ou `kpfonts` proposent une option permettant de présenter les équations « à la française » c'est-à-dire avec les majuscules latines ainsi que toutes les lettres grecques droites et les minuscules latines en italique.

Avec `unicode-math` le choix est plus large et disponible pour toutes les fontes mathématiques OpenType (elles ont toutes les lettres latines et grecques droites et italiques et aussi grasses et grasses italiques). Ce choix repose sur l'option `math-style` qui peut prendre quatre valeurs présentées dans le tableau suivant extrait du manuel [4] de `unicode-math`. L'option par défaut est `math-style=TeX`.

Cette option peut être passée soit directement à `unicode-math` (elle s'appliquera alors à *toutes* les fontes mathématiques) soit à la commande `\setmathfont{}` :

`\usepackage[math-style=french]{unicode-math}` ou bien

`\setmathfont{Fira Math}[math-style=french]` pour la fonte Fira Math.

Dans le cas où on charge `fourier-otf` comme dans l'exemple 1 p. 4 on peut passer l'option à `fourier-otf` comme ceci :

`\usepackage[math-style=french]{fourier-otf}`

TABLE 1 : Effets de l'option `math-style`.

Option	Latin	Grec
<code>math-style=ISO</code>	$(a, z, B, X)$	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=TeX</code>	$(a, z, B, X)$	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=french</code>	$(a, z, B, X)$	$(\alpha, \beta, \Gamma, \Xi)$
<code>math-style=upright</code>	$(a, z, B, X)$	$(\alpha, \beta, \Gamma, \Xi)$

Deux autres options se passent de la même manière : `nabla=italic` (`upright` par défaut) et `partial=upright` (`italic` par défaut), elle permettent de modifier la présentation des opérateurs  $\nabla$  et  $\partial$  (défaut) ou  $\nabla$  et  $\partial$  avec ces options.

D'autre part, `unicode-math` introduit de nouvelles commandes `\symup{}` et `\symit{}` qui facilitent la saisie des variantes droites ou italiques : on code `\symit{\beta}` (ou `\symit{\beta}`) pour saisir un  $\beta$  italique lorsque l'option `math-style=french` est activée ou inversement `\symup{\beta}` (ou `\symup{\beta}`) pour un  $\beta$  droit avec l'option par défaut.

Plus généralement, les utilisateurs sont invités à remplacer systématiquement les anciennes commandes `\mathxy{}` par `\symxy{}`, voir le tableau 2.

TABLE 2 : Nouvelles commandes

Commande	remplace	Commande	remplace
<code>\symup</code>	<code>\mathrm</code>	<code>\symbfup</code>	
<code>\symit</code>		<code>\symbfit</code>	
<code>\symcal</code>	<code>\mathcal</code>	<code>\symbfcal</code>	<code>\mathbfcal</code>
<code>\symscr</code>	<code>\mathscr</code>	<code>\symbfscr</code>	<code>\mathbfscr</code>
<code>\symfrak</code>	<code>\mathfrak</code>	<code>\symbffrak</code>	<code>\mathbffrak</code>
<code>\symsfup</code>	<code>\mathsf</code>	<code>\symbfsfup</code>	
<code>\symsfit</code>		<code>\symbfsfit</code>	
<code>\symbb</code>	<code>\mathbb</code>		
<code>\symbbit</code>			

Les noms sont parlants `bf` (boldface) pour le gras, `sf` pour sans-sérif.

Les anciennes commandes `\mathcal`, `\mathrm`, `\mathbf`, etc., fonctionnent comme avant... c'est-à-dire avec des résultats parfois inattendus : prenons le produit des deux variables droites `f`, et `i`, le codage traditionnel `\mathrm{f}\mathrm{i}` donne « `fi` » (avec ligature!) tandis que `\symup{f}\symup{i}` produit « `fi` » sans ligature parasite.

Idem en gras, `\mathbf{f}\mathbf{i}` donne « `fi` » contre « `fi` » pour le nouveau codage `\symbf{f}\symbf{i}`.

Il est donc recommandé de réserver les commandes en `\math.` aux noms d'opérateurs et d'utiliser systématiquement les commandes en `\sym.` pour les symboles, noms de variables, d'ensembles, etc.

Ceci n'est qu'une brève introduction à la documentation de `unicode-math` dont je recommande vivement la lecture [4].

## **Remerciements**

Un grand merci à Jacques André pour ses multiples contributions à ce document et pour sa relecture attentive!

## Références

- [1] MANUEL PÉGOURIÉ GONNARD, *Un guide pour LuaLaTeX*, Cahiers Gutenberg n° 54-55, [http://cahiers.gutenberg.eu.org/cg-bin/article/CG\\_2010\\_\\_\\_54-55\\_13\\_0.pdf](http://cahiers.gutenberg.eu.org/cg-bin/article/CG_2010___54-55_13_0.pdf)
- [2] JACQUES ANDRÉ, *Utilisation de \fontspec pour des caractères spéciaux*, <http://jacques-andre.fr/fontex/casseau+fontspec.pdf>
- [3] WILL ROBERTSON, *The fontspec package*, fichier `fontspec.pdf` de votre distribution TeX (en anglais) accessible sous TeXLive par la commande `texdoc fontspec`.
- [4] WILL ROBERTSON, *Experimental Unicode mathematical typesetting : The unicode-math package*, fichier `unicode-math.pdf` de votre distribution TeX (en anglais) accessible sous TeXLive par la commande `texdoc unicode-math`.
- [5] ULRİK VIETH, *An updated survey of OpenType math fonts*, TUGBoat, vol. 44, no 2 (à paraître).

# Annexes

## A Conversion de codage

Dorénavant, il n'y a aucune raison d'utiliser un codage différent de UTF-8 pour *tous* les documents textuels : il convient de demander à votre éditeur de texte de travailler en UTF-8 par défaut.

Reste le cas des anciens documents encore codés en LATIN-1, LATIN-9, APPLEMAC ou autres. Votre éditeur est probablement capable de faire la conversion<sup>10</sup>, sinon il suffit de faire appel à un utilitaire comme `iconv` (sous Linux ou MacOSX), sous Windows il existe un équivalent `Liblconv`. Si le texte initial est en LATIN-1, la commande `iconv -c -f LATIN1 -t UTF-8 source.tex > source-utf8.tex` créera un nouveau fichier `source-utf8.tex` codé en UTF-8 tout en conservant l'original. Pour APPLEMAC remplacer `LATIN1` par `MAC`.

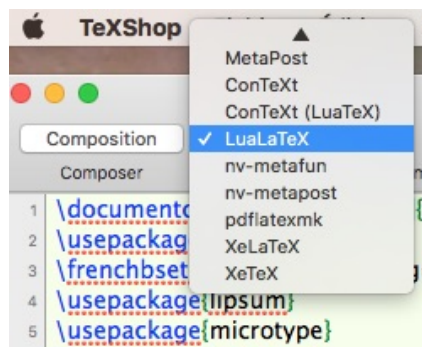
## B Choix du moteur de compilation

### B.1 TeXshop

Pour utiliser le format LuaLaTeX depuis TeXShop, deux possibilités :

1. Par barre de menu

Une fois un fichier-source ouvert et prêt à être compilé, il suffit de choisir le moteur en faisant dérouler la fenêtre Composition dans la barre de menu et d'y choisir LuaLaTeX (comme on choisit les autres compilateurs, tels que `bibtex`, etc.).



2. Par ligne de commande

On peut insérer en tête de son programme la ligne suivante :

```
% !TEX TS-program = LuaLaTeX
```

et le compiler avec LaTeX ou avec LuaLaTeX comme dans le point 1.

---

10. Sous Emacs, menu « Options », onglet « Multilingual Environment », choisir « Set Coding Systems » puis « For saving This Buffer », le nom du codage (utf-8) est à taper dans la ligne de contrôle en bas de l'écran. Il y a aussi un raccourci clavier : `Ctrl-x <Return> f`.

## B.2 TeXmaker

Dans le menu « Compilation rapide » choisir « LuaLaTeX ».

## B.3 TeXworks

Le choix du format (pdfLaTeX, LuaLaTeX, etc.) se fait dans le menu en haut à gauche de la fenêtre, il affiche « pdfaTeX » par défaut.

## B.4 Emacs + AucTeX

Lorsqu'un fichier contenant un source TeX est ouvert, cliquer sur le menu « Command », choisir l'onglet « TeXing Options » et activer « Use LuaTeX Engine ». Les prochaines compilations seront faites avec ce moteur et non avec le moteur par défaut (pdfTeX), ce jusqu'à la fermeture du fichier. Le choix du moteur ne survivra pas à la réouverture du fichier...

Une autre façon de faire est d'ajouter la ligne `%% TeX-engine: luatex` aux variables locales en toute fin de fichier, par exemple

```
%%% Local Variables:
%%% mode: latex
%%% TeX-engine: luatex
%%% coding: utf-8
%%% End:
```

Notez que ces variables sont lues une seule fois à l'ouverture du fichier ; il faudra donc recharger celui-ci pour qu'elles soient prises en compte mais le choix du moteur est permanent : le-dit fichier sera systématiquement compilé avec le moteur LuaTeX, sauf nouvelle modification des variables locales.

Emacs permet aussi de passer facilement de la saisie d'un texte latin à du grec ou du cyrillique (voire du chinois). Voir le menu « Options », onglet « Multilingual Environment » puis « Select Input Method » où on choisit la langue secondaire, ici le grec pour les maths. Ensuite le basculement d'une langue à l'autre se fait par le raccourci clavier `Ctrl-\`.



LATIN MODERN

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

EREWHON

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

NEW COMPUTER MODERN

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

TEX GYRE TERMES

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

TEX GYRE PAGELLA

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

PAGELLA & ASANA MATH

Si les « bons » guillemets, l’euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**  
Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## TeX Gyre Bonum

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## TeX Gyre Schola

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## Libertinus

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## EBGaramond

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## XCharter

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## Old Standard

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel `\og`, `\fg`, `\texteuro`, `\zeta`, `\Gamma`...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## CONCRETE & EULER

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## CONCRETE-MATH

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## KPFonts ROMAN

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## DejaVu Serif

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## STIX TWO

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## CAMBRIA

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789 ?

## NOTO SANS

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## Lato & LeteSansMath

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## FIRA SANS

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## KPFonts SANS

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## ARSENAL

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## GFS NEOHELLENIC

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou elzéviens 0123456789?

## MintSpirit & Euler

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou  
elzéviens 0 1 2 3 4 5 6 7 8 9 ?

## LUCIOLE

Si les « bons » guillemets, l'euro €, les lettres grecques ne peuvent pas être saisis *directement* au clavier, il reste la possibilité du codage traditionnel \og, \fg, \texteuro, \zeta, \Gamma...

$$\zeta(z) = \sum_{n=1}^{\infty} \frac{1}{n^z} = \frac{1}{\Gamma(z)} \int_0^{\infty} \frac{t^{z-1}}{e^t - 1} dt$$

**Du gras, du gras italique.**

Chiffres standard 0123456789 ou  
elzéviens 0123456789 ?