

Extension garamondx*

Michael Sharpe

5 février 2015

1 Introduction

Cette extension ajoute à `ugm` des fonctionnalités autrefois appelées *expert*, d'où le `x`. La révision de la version 1.095 peut affecter l'utilisation de `babel` (cf. section 5). Les polices `ugm`, (URW)+ + GaramondNo8, ne sont pas libres au sens du GNU, mais sont mis à disposition sous l'AFPL (*Aladdin Free Public License*), qui empêche leur diffusion au sein de `TEXLive`. Elles peuvent être téléchargées à l'aide du script `getnonfreefonts` qui fait partie de `TEXLive`. Les instructions d'installation sont disponibles sur :

<http://tug.org/fonts/getnonfreefonts/>.

Les polices de cette extension dérivent en définitive des polices `ugm`, et sont donc également soumises à la même licence AFPL dont tous les détails sont précisés sur :

<http://www.artifex.com/downloads/doc/Public.htm>.

En termes généraux, la licence permet l'utilisation illimitée des polices par n'importe qui, mais n'autorise aucune rémunération pour leur distribution. Elle contraint également ceux qui modifient les polices à les délivrer sous la même licence, et les oblige à fournir des informations sur la nature des changements et leur identité.

Dans les polices `ugm` du CTAN, il manque :

- un ensemble complet de ligatures en `f` (`f_f`, `f_f_i` et `f_f_l` sont manquantes);
- les petites capitales;
- les chiffres elzéviens.

Les glyphes eux-mêmes sont très proches de ceux de la police Stempel Garamond distribuée par Adobe, qui a de nombreux admirateurs, mais qui n'a pas, elle non plus, les ligatures du `f`. Le but ici est donc de créer une extension qui fournit ces traits manquants qui devraient, à mon avis, faire obligatoirement partie de toutes les extensions modernes de `LATEX`.

Dans le codage T1 de cette distribution, il manque seulement le glyphe `perthousandzero`, qui est rarement présent dans les polices PostScript, et n'est pratiquement jamais utilisé dans les extensions `LATEX`.

La dernière version contient une restitution complète de tous les styles dans le codage TS1. Par défaut, `garamondx.sty` réclame l'extension `textcomp`, mais quelques fonctionnalités sont absentes dans cette version. Pour bénéficier pleinement du codage TS1, chargez les extensions de cette façon :

```
\usepackage[full]{textcomp}
\usepackage{garamondx}
```

*Traduit par René Fritz le 21 février 2015.

2 Un peu d'histoire

Contrairement à la plupart des autres polices dont le nom renferme le terme Garamond, les glyphes de cette police sont en fait des rendus numériques des polices réellement conçues par Claude Garamond dans le milieu du xvi^e siècle — la plupart des autres polices Garamond sont plus proches des polices conçues par Jean Jannon quelques années plus tard. La société Stempel possédait le spécimen à partir duquel ils ont conçu les moulages métalliques des polices dans les années 20. Les premiers rendus numériques comprennent ceux de Bitstream sous le nom OriginalGaramond et Stempel Garamond d'Adobe, sous licence Linotype. (Il semble que bon nombre des lacunes des polices conçues par Linotype étaient des artefacts dus aux limites des machines pour lesquelles les polices ont été conçues et n'ont, dans la plupart des cas, pas été corrigées.)

La dernière version (TrueType, non PostScript) officielle de (URW)++ GaramondNo8 est disponible sur :

<ftp://mirror.cs.wisc.edu/pub/mirrors/ghost/AFPL/GhostPCL/urwfonts-8.71.tar.bz2>. et possède une collection de glyphes plus étendue que celles des versions PostScript. En particulier, les ligatures en f sont présentes, ainsi que les glyphes Eng et eng qui dans le codage T1 sont obtenus respectivement par les commandes \NG et \ng.

À ma connaissance, deux tentatives, assez récentes, ont visé à corriger ces polices.

La première, sur laquelle ce travail est basé, est celle de Gael Varoquaux, disponible sur : <http://gael-varoquaux.info/computers/garamond/index.html>. Son extension ggm, qui n'est pas distribuée sur le CTAN, ne semble pas avoir été largement diffusée.

La seconde est une distribution OpenType de Rogério Brito et Khaled Hosni disponible sur : <https://github.com/rbrito/urw-garamond>. Brito semble avoir fait un effort pour obtenir (URW)++ afin de délivrer les polices sous une licence moins restrictive, qui ne semble pas avoir porté ses fruits. Leur projet s'adressait principalement aux utilisateurs de Lua_T_E_X et X_Y_L_A_T_E_X, et reste incomplet.

De l'extension ggm, j'ai gardé (a) une base de départ pour améliorer les métriques ; (b) le glyphe Q ornementé, mais en option.

3 Nouveautés

Les principales nouveautés sont : (i) les petites capitales conçues en Regular, Italic, Bold and Bold Italic ; (ii) les chiffres elzéviens conçus pour chaque style et graisse ; (iii) un ensemble complet de ligatures du f ; (iv) les macros pour personnaliser les chiffres ou le Q utilisés par défaut ; (v) une police Text Companion complète dans chaque style et graisse. Pour plus de détails sur les points (i) et (ii), voir les sections dédiées.

4 Options

Cette extension utilise le codage T1 qu'elle intègre en interne : il n'est donc pas nécessaire de l'appeler séparément. De même, l'extension textcomp est chargée automatiquement, vous donnant accès à de nombreux symboles absents dans le codage T1 (cependant, il est préférable de charger l'extension textcomp avec l'option full avant d'appeler garamondx).

- L’option `scaled` peut être utilisée pour mettre toutes les polices à l’échelle indiquée par le nombre spécifié. Par exemple, `scaled=.9` réduit toutes les polices à 90 % de leur taille. Si vous entrez seulement l’option `scaled` sans valeur, la taille par défaut sera de 95 %, ce qui est la bonne mesure pour amener la hauteur des capitales de GaramondNo8 à 0,665 em ; une taille à peu près normale pour une police de texte, mais avec une hauteur d’x inférieure à la normale qui est typique des polices Garamond.
- Par défaut, l’extension utilise les chiffres alignés 0123456789 plutôt que les chiffres elzéviens 0123456789. L’option `osf` force le passage des chiffres au style ancien que je préfère, 0123456789, où le 1 ressemble au chiffre 1 aligné avec une hampe raccourcie, tandis que l’option `osfI` utilise les chiffres elzéviens traditionnels 0123456789, où le 1 ressemble à la lettre I avec une hampe raccourcie. Quelle que soit l’option utilisée,
 - `\textlf{1}` produira le chiffre 1 aligné ;
 - `\textosf{1}` produira mon style ancien préféré 1 ;
 - `\textosfI{1}` produira la style ancien traditionnel 1.
- Par défaut, la lettre Q est dans la version traditionnelle du GaramondNo8. Elle peut être remplacée partout par la version ornementée à l’aide de l’option, `swashQ` qui vous donnera, par exemple, Quo*u*! Que vous ayez précisé l’option `swashQ` ou non, vous pouvez imprimer un Q ornementé dans la graisse et la forme courante en écrivant `\swashQ`. Ainsi, `\swashQ uash` donnera Qu*u*ash.

5 Nouveau dans la version 1.095

La dernière version de `garamondx` utilise `\AtEndPreamble` de l’extension `etoolbox` pour appliquer les options de style de chiffre `osf`, `osfI` après tous les autres choix du préambule. Il semble que cette macro puisse être incompatible avec `babel` dans certains cas où le choix de la langue implique un codage différent de T1. Si vous trouvez lorsque vous utilisez `babel` que vous obtenez des erreurs concernant la corruption des tables NFSS, vous devriez utiliser l’option `babel` de `garamondx` et éventuellement réorganiser l’ordre de quelques macros. Par exemple, si vous avez utilisé,

```
\usepackage[osf]{garamondx}
\usepackage[libertine,bigdelims]{newtxmath}
```

et trouvez des erreurs lors de l’utilisation de `babel`, vous devriez changer pour

```
\usepackage{garamondx}
\usepackage[libertine,bigdelims]{newtxmath}
\useosf
```

Les problèmes avec `babel` ne se produisent pas dans l’une des conditions suivantes :

- en composant avec `latex+dvips` ;
- en utilisant seulement l’environnement `otherlanguage*` ;
- en supprimant l’utilisation de `\AtEndPreamble`.

5.1 Exemples

Voici les effets de quelques options qui ne peuvent être utilisées quand l'absence de l'option `babel` :

```
\usepackage[scaled=.9,osf]{garamondx}% réduit à 90 % mon style ancien
\usepackage[scaled,osf]{garamondx}% réduit à 95 % mon style ancien
\usepackage[osfI]{garamondx}% style ancien traditionnel
\usepackage[osfI,swashQ]{garamondx}% idem avec tous les Q ornementés
```

6 Effets de fontaxes

Cette extension charge `fontaxes` pour accéder aux petites capitales italiques. Prenez garde au fait que `fontaxes` modifie le comportement de certaines macros essentielles de \LaTeX telles que `\textsc` et `\textup`. D'ordinaire, \LaTeX autorise la combinaison de certains styles de texte, ainsi `\textbf{\textit{a}}` produit un a gras italique, alors que d'autres styles ne peuvent pas être combinés, ainsi `\textsc{\textup{a}}` donne le même résultat que `\textup{a}` : c'est-à-dire la lettre a verticale, et non en petite capitale. Avec `fontaxes`, `\textsc{\textup{a}}` produit un a en petite capitale et non un a vertical. Il dispose d'une macro `\textulc` qui annule les petites capitales, de sorte que, par exemple, `\textsc{\textulc{a}}` produit un a qui n'est pas en petite capitale, quels que soient les autres choix de styles en vigueur, tels le gras ou l'italique.

7 Chiffres supérieurs

Les versions TrueType de `GaramondNo8` ont un ensemble complet des chiffres supérieurs, contrairement à leurs homologues PostScript. Seuls les glyphes des chiffres supérieurs de graisse ordinaire ont été copiés dans `NewG8-sups.pfb` et `NewG8-sups.afm` et introduits par un `tfm` nommé `NewG8-sups.tfm` qui peut être utilisé par l'extension `superiors` pour rendre les appels de notes de bas de page ajustables. Consultez le `superiors-doc.pdf` (vous pouvez le trouver dans \TeX Live en tapant `texdoc superiors` dans la fenêtre du Terminal). Pour l'appeler, tapez simplement :

```
\usepackage[supstfm=NewG8-sups]{superiors}
```

8 Glyphes en codage TS1

La table suivante présente la police `Text Companion` dans le codage `TS1` qui est entièrement restitué dans cette extension. Vous trouverez ensuite les macros pour appeler ces glyphes. Bien que présentés en maigre et dans une forme verticale, ces glyphes sont disponibles dans toutes les graisses et formes.

	'0	'1	'2	'3	'4	'5	'6	'7	
'00x	\` 0	' 1	^ 2	~ 3	¨ 4	" 5	° 6	ˇ 7	"0x
'01x	˘ 8	ˉ 9	˙ 10	˚ 11	˛ 12	˜ 13	14	15	
'02x	16	17	„ 18	19	20	■ 21	■ 22	23	"1x
'03x	← 24	→ 25	ˆ 26	˜ 27	˘ 28	˙ 29	30	31	
'04x	ƒ 32	33	34	35	\$ 36	37	38	' 39	"2x
'05x	40	41	* 42	43	, 44	= 45	• 46	/ 47	
'06x	o 48	1 49	2 50	3 51	4 52	5 53	6 54	7 55	"3x
'07x	8 56	9 57	58	59	< 60	– 61	> 62	63	
'10x	64	65	66	67	68	69	70	71	"4x
'11x	72	73	74	75	76	Ů 77	78	○ 79	
'12x	80	81	82	83	84	85	86	Ω 87	"5x
'13x	88	89	90	∥ 91	92	∥ 93	↑ 94	↓ 95	
'14x	\` 96	97	★ 98	o o 99	† 100	101	102	103	"6x
'15x	104	105	106	107	☞ 108	⊙ 109	♪ 110	111	
'16x	112	113	114	115	116	117	118	119	"7x
'17x	120	121	122	123	124	125	~ 126	= 127	
'20x	˘ 128	ˇ 129	¨ 130	˘ 131	† 132	‡ 133	∥ 134	‰ 135	"8x
'21x	• 136	°C 137	\$ 138	¢ 139	f 140	¢ 141	₩ 142	₯ 143	
'22x	₣ 144	₧ 145	£ 146	₪ 147	? 148	ı 149	đ 150	™ 151	"9x
'23x	‰ 152	¶ 153	₹ 154	№ 155	% 156	€ 157	o 158	SM 159	
'24x	{ 160	} 161	¢ 162	ₓ 163	⊠ 164	¥ 165	166	₪ 167	"Ax
'25x	¨ 168	© 169	ª 170	© 171	¬ 172	® 173	® 174	- 175	
'26x	° 176	± 177	² 178	³ 179	' 180	μ 181	¶ 182	• 183	"Bx
'27x	※ 184	¹ 185	º 186	√ 187	¼ 188	½ 189	¾ 190	€ 191	
'32x	208	209	210	211	212	213	× 214	215	"Dx
'33x	216	217	218	219	220	221	222	223	
'36x	240	241	242	243	244	245	÷ 246	247	"Fx
'37x	248	249	250	251	252	253	254	255	
	"8	"9	"A	"B	"C	"D	"E	"F	

LISTE DES MACROS POUR ACCÉDER AUX SYMBOLES TS1 EN MODE TEXTE :

Notez que les positions 0-12 et 26-29 renferment des accents, utilisés comme \t{a} pour faire un tirant (ici un : *tie-after accent*) sur la lettre a (ā). Les positions 23 et 31 ne contiennent pas de glyphes visibles, mais chacune occupe une hauteur précisée par leur nom.

- 0 \capitalgrave
- 1 \capitalacute
- 2 \capitalcircumflex
- 3 \capitaltilde
- 4 \capitaldieresis
- 5 \capitalhungarumlaut
- 6 \capitalring

7 \capitalcaron
8 \capitalbreve
9 \capitalmacron
10 \capitaldotaccent
11 \capitalcedilla
12 \capitalogonek
13 \textquotestraightbase
18 \textquotestraightdblbase
21 \texttwelveudash
22 \textthreequartersemdash
23 \textcapitalcompwordmark
24 \textleftarrow
25 \textrightarrow
26 \t % tie accent, skewed right
27 \capitaltie % skewed right
28 \newtie % tie accent centered
29 \capitalnewtie % ditto
31 \textascendercompwordmark
32 \textblank
36 \textdollar
39 \textquotesingle
42 \textasteriskcentered
45 \textdblhyphen
47 \textfractionsolidus
48 \textzerooldstyle
49 \textoneoldstyle
50 \texttwooldstyle
49 \textthreeoldstyle
50 \textfouroldstyle
51 \textfiveoldstyle
52 \textsixoldstyle
53 \textsevenoldstyle
54 \texteightoldstyle
55 \textnineoldstyle
60 \textlangle
61 \textminus
62 \textrangle
77 \textmho
79 \textbigcircle
87 \textohm
91 \textlbrackdbl
93 \textrbrackdbl
94 \textuparrow
95 \textdownarrow
96 \textasciigrave
98 \textborn

99 \textdivorced
100 \textdied
108 \textleaf
109 \textmarried
110 \textmusicalnote
126 \texttildelow
127 \textdblhyphenchar
128 \textasciibreve
129 \textasciicaron
130 \textacutedbl
131 \textgravedbl
132 \textdagger
133 \textdaggerdbl
134 \textbardbl
135 \textperthousand
136 \textbullet
137 \textcelsius
138 \textdollaroldstyle
139 \textcentoldstyle
140 \textflorin
141 \textcolonmonetary
142 \textwon
143 \textnaira
144 \textguarani
145 \textpeso
146 \textlira
147 \textrecipe
148 \textinterrobang
149 \textinterrobangdown
150 \textdong
151 \texttrademark
152 \textpertenthousand
153 \textpilcrow
154 \textbaht
155 \textnumero
156 \textdiscount
157 \textestimated
158 \textopenbullet
159 \textservicemark
160 \textlquill
161 \textrquill
162 \textcent
163 \textsterling
164 \textcurrency
165 \textyen
166 \textbrokenbar

167 `\textsection`
168 `\textasciidieresis`
169 `\textcopyright`
170 `\textordfeminine`
171 `\textcopyleft`
172 `\textlnot`
173 `\textcircledP`
174 `\textregistered`
175 `\textasciimacron`
176 `\textdegree`
177 `\textpm`
178 `\texttwosuperior`
179 `\textthreesuperior`
180 `\textasciiacute`
181 `\textmu`
182 `\textparagraph`
183 `\textperiodcentered`
184 `\textreferencemark`
185 `\textonesuperior`
186 `\textordmasculine`
187 `\textsurd`
188 `\textonequarter`
189 `\textonehalf`
190 `\textthreequarters`
191 `\texteuro`
214 `\texttimes`
246 `\textdiv`

La macro `\textcircled` permet d'encercler une lettre en utilisant `\textbigcircle`. La lettre est toujours composée en petite capitale, si bien que les lettres encerclées sont toujours en majuscules : `\textcircled{M}` et `\textcircled{m}` donnent un résultat identique, à savoir .

9 Détails de mise en œuvre

9.1 Petites capitales

Les petites capitales ont été créées à partir des capitales réduites à 67 % de manière uniforme avec FontForge, puis épaissies à 130 % au niveau des pleins horizontaux et verticaux. Les résultats ont servi d'ébauche pour façonner chaque glyphe. Avec FontForge, les pleins ont été redimensionnés correctement, imposant souvent une refonte de la forme. Les résultats eux-mêmes sont la seule description possible de ces transformations. Après la création des glyphes, les informations métriques appropriées ont été établies grâce à FontForge. La forme verticale en graisse normale, a été bien plus retravaillée que pour les autres graisses, et semble meilleure, à mon avis. Créer des petites capitales à partir de zéro exige un véritable effort pour obtenir les glyphes, les métriques et le bon crénage. Dans les deux formes verticales

normales et grasses, les glyphes standards accentués, certains caractères spéciaux, les ligatures a_e et o_e et les glyphes lslash et oslash sont fournis.

La macro pour les petites capitales `\textsc` coopère avec `\textbf` et `\textit`, de sorte que vous pouvez utiliser, par exemple :

```
\textsc{Capitales et Petites Capitales}
```

pour produire **CAPITALES ET PETITES CAPITALES**,

```
\textit{\textsc{Capitales et Petites Capitales}}
```

pour produire *CAPITALES ET PETITES CAPITALES*,

```
\textbf{\textsc{Capitales et Petites Capitales}}
```

pour produire **CAPITALES ET PETITES CAPITALES**, et

```
\textbf{\textit{\textsc{Capitales et Petites Capitales}}}
```

pour produire *CAPITALES ET PETITES CAPITALES*.

9.2 Chiffres elzéviens

Les chiffres elzéviens ont été créés à partir des chiffres alignés existants, en réduisant les pleins du 0 et du 1 à la taille de minuscules avec FontForge, et en abaissant les positions verticales des autres chiffres. Les formes ont été ensuite modifiées avec FontForge pour obtenir un aspect plus traditionnel du style ancien — les résultats eux mêmes montrent les transformations concernées.

9.3 Glyphes de Text Companion

Pour fournir un jeu complet de ces glyphes en codage TS1, certains d’entre eux (*tie accents, born, died, married, divorced, referencemark, numero, discount, estimated, copyleft, centoldstyle*) ont été adaptés à partir de Computer Modern, mais avec des graisses appropriées à garamondx. Les autres glyphes ont été pratiquement copiés à partir des glyphes légèrement modifiés de txfonts, qui restitue complètement les glyphes de Text Companion dans toutes les graisses et tous les styles.

10 Utilisation avec des extensions math

L’extension `mathdesign` de Paul Pichaureau possède l’option `garamond` qui rend compatible son extension `math` avec le texte en `ugm`. Pour utiliser `mathdesign` avec `garamondx` tapez :

```
\usepackage[full]{textcomp}  
\usepackage[garamond]{mathdesign}  
\usepackage{garamondx}
```

Il est aussi possible d'utiliser l'option `garamondx` de `newtxmath` pour employer les italiques capitales et minuscules de `garamondx`, parfaitement adaptées aux maths, au lieu des italiques Times utilisées par défaut. Cela n'est possible qu'à partir de la version 1.06 de l'extension `newtxmath`.

```
\usepackage[full]{textcomp}
\usepackage{garamondx} % avec les chiffres alignés bons pour les maths
\usepackage[varqu,varl]{zi4} % police machine à écrire inconsolata
\usepackage[sf]{libertine} % police biolinum sans-serif
\usepackage[garamondx,cmbraces]{newtxmath}
\useosf % style des chiffres en osf pour le texte et non pour les maths
```

REMARQUE. – La dernière commande, ainsi que son homologue `\useosfI`, ne peuvent être utilisées que dans le préambule, et ne doivent jamais précéder `\usepackage{garamondx}`.

11 Licence

Les polices de cette extension sont dérivées des polices (URW)++ GaramondNo8 qui ont été délivrées sous l'AFPL ; il en est donc de même pour ces polices. Les autres fichiers associés sont soumis à la *L^AT_EX Project Public License* ; voir sur le site : <http://www.ctan.org/tex-archive/help/Catalogue/licenses.lppl.html> pour les détails de cette licence.

Les modifications de cette extension et des polices décrites ci-dessus sont protégées par Copyright Michael Sharpe, msharpe@ucsd.edu, October 1, 2013.

11.1 Fichiers couverts par l'AFPL

```
NewG8-Bol.afm
NewG8-Bol.pfb
NewG8-Bol-SC.afm
NewG8-Bol-SC.pfb
NewG8-BolIta.afm
NewG8-BolIta.pfb
NewG8-BolIta-SC.afm
NewG8-BolIta-SC.pfb
NewG8-Ita-SC.afm
NewG8-Ita-SC.pfb
NewG8-Ita.afm
NewG8-Ita.pfb
newG8-Osf-bol.afm
newG8-Osf-bol.pfb
newG8-Osf-bolita.afm
newG8-Osf-bolita.pfb
newG8-Osf-ita.afm
newG8-Osf-ita.pfb
newG8-Osf-reg.afm
```

newG8-0sf-reg.pfb
NewG8-Reg-SC.afm
NewG8-Reg-SC.pfb
NewG8-Reg.afm
NewG8-Reg.pfb
NewG8-sups.afm
NewG8-sups.pfb