

The garamondx package

Michael Sharpe

March 16, 2015

1 Overview

This package provides an extension of the `ugm` package, adding features that were once referred to as *expert*, whence the `x`. There has been a revision as of version 1.095 which may affect those using `babel`. See section 5 for details. The `ugm` fonts, (URW)++ GaramondNo8, are not free in the sense of GNU but are made available under the AFPL (Aladdin Free Public License), which is restrictive enough to prevent their distribution as part of T_EXLive. They may be downloaded using the `getnonfreefonts` script that used to be part of T_EXLive. Instructions for installation are laid out at

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

The fonts in this package are derived ultimately from the `ugm` fonts, and so are also subject to the same AFPL license, the precise details of which are spelled out at

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

In broad terms, the license allows unlimited use of the fonts by anyone, but does not not permit any fee for their distribution. It also restricts those who modify the fonts to release them under the same license, and requires them to provide information about the changes and the identity of the modifier.

The `ugm` fonts on CTAN lack:

- a full set of f-ligatures (`f_f`, `f_f_i` and `f_f_l` are missing);
- small caps;
- old style figures.

The glyphs themselves are very close to those in Adobe's Stempel Garamond font package, which has many admirers, though they also lack the same f-ligatures. So, the goal here is to make a package which provides these missing features which should, in my opinion, be an essential part of any modern L^AT_EX font package.

The only glyph missing from the T1 encoding in this distribution is `perthousandzero`, which is only rarely present in PostScript fonts, and is almost never required as part of L^AT_EX packages.

The latest version contains a full rendition of the TS1 encoding in all styles. By default, the `textcomp` package is required by `garamondx.sty`, but the default version lacks a couple of features. To be able to make use of all TS1 features, load the packages thus:

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

2 Some History

Unlike most other fonts having Garamond as part of the name, the glyphs in this font are in fact digital renderings of fonts actually designed by Claude Garamond in the mid sixteenth century — most other Garamond fonts are closer to fonts designed by Jean Jannon some years later. The Stempel company owned the specimen from which they designed metal castings of the fonts in the 1920's. Early digital renderings include those by Bitstream under the name OriginalGaramond, and Stempel Garamond from Adobe, licensed from LinoType. (It appears that many of the deficiencies of fonts designed by LinoType were artifacts of the limitations of the machines for which the fonts were designed, and have in most cases not been corrected.)

The latest version (TrueType, not PostScript) of the official (URW)++ GaramondNo8 is available from

<ftp://mirror.cs.wisc.edu/pub/mirrors/ghost/AFPL/GhostPCL/urwfonts-8.71.tar.bz2>

which has a more extensive collection of glyphs than the PostScript versions. In particular, the f-ligatures are there, as well as the glyphs Eng, eng that are part of the T1 encoding under the names Ng, ng.

To my knowledge, there have been two fairly recent attempts to rework these fonts. The first, upon which this work is based, was by Gael Varoquaux, available at

<http://gael-varoquaux.info/computers/garamond/index.html>

His ggm package seems never to have been widely distributed, not having appeared on CTAN.

The second was an OpenType package by Rogério Brito and Khaled Hosny at

<https://github.com/rbrito/urw-garamond>

Brito seems to have made an effort to get (URW)++ to release the fonts under a less restrictive license, which does not appear to have been successful. Their project was aimed mainly towards users of LuaTeX and XeTeX, and remains incomplete.

What I kept from the ggm package was (a) a starting point for improved metrics; (b) the swash Q glyph, though not as the default Q.

3 New in this package

The most important items are (i) newly designed Small Cap fonts for Regular, Italic, Bold and Bold Italic; (ii) newly designed old style figures for each weight/style; (iii) a full set of f-ligatures; (iv) macros to allow customizations of the default figures and the default Q; (v) a full text companion font in each weight/style. For details of (i) and (ii), see the last section.

4 Package Options

The package uses T1 encoding—this is built into the package and need not be specified separately. Likewise, the textcomp package is loaded automatically, giving you access to many symbols not included in the T1 encoding. (It is better though to load textcomp with option full before loading garamondx.)

- The option `scaled` may be used to scale all fonts by the specified number. Eg, `scaled=.9` scales all fonts to 90% of natural size. If you provide just the option `scaled` without a value, the default is 0.95, which is about the correct scaling to bring the Cap-height of GaramondNo8 down to .665em, about normal for a text font, but with a smaller than normal x-height that is typical of Garamond fonts.
- By default, the package uses lining figures 0123456789 rather than oldstyle figures 0123456789. The option `osf` forces the figure style to a modified oldstyle that I prefer, 0123456789, where the 1 looks like a lining figure 1 with a shortened stem, while the option `osfI` uses the more traditional oldstyle figures 0123456789, where the 1 looks like the letter I with a shortened stem. No matter which option you use:
 - `\textlf{1}` produces the lining figure 1;
 - `\textosf{1}` produces my preferred oldstyle 1;
 - `\textosfI{1}` produces the traditional oldstyle 1.
- The default version of the letter Q is the traditional one from GaramondNo8. It may be replaced everywhere by the swash version via the option `swashQ`, which gives you, eg, Quoi?

Whether or not you have specified the option `swashQ`, you may print a swash Q in the current weight and shape by writing `\swashQ` — eg,

`\swashQ` uash.

produces Quash.

5 Using `babel` with `garamondx`

Recent version of `garamondx` have made use of `\AtEndPreamble` from the `etoolbox` package to apply the figure style options `osf`, `osfI` after all other preamble choices. This is useful to ensure that oldstyle figures are not specified until after all math packages have been loaded. It now appears that this macro can be incompatible with `babel` in some cases where the language choice involves an encoding other than T1. If you find when using `babel` that you get errors about corruption of your NFSS tables or find that some `babel` features do not work as expected, you may need to use the `babel` option to `garamondx` and possibly rearrange the order of some macros. First of all, you should load `babel` before `garamondx`. When `garamondx.sty` tuns, it sets a flag if it detects that `babel` has already been loaded, and modifies some commands. This flag may also be set by the option `babel`. If for some reason you must load `babel` after `garamondx`, specify the `babel` option when loading `garamondx`, but be aware that `babel` may not work as expected in this case.

Examples:

1:

```
% No math package
\usepackage[greek,polutonico,english]{babel}
\usepackage[osf]{garamondx} % or osfI
```

2:

```
% No math package (same effect as previous)
\usepackage[greek,polutonico,english]{babel}
```

```
\usepackage{garamondx}
\useosf % or \useosfI
```

3:

% with math package

```
\usepackage{greek.polutonico,english}{babel}
\usepackage{garamondx} % don't use options osf or osfI
\usepackage[garamondx,bigdelims]{newtxmath}
\useosf % or \useosfI, must follow math package
```

5.1 Examples

The following show the effects of some options that may apply only when the `babel` option is not specified:

```
\usepackage[scaled=.9,osf]{garamondx}% scaled to 90%, my oldstyle
\usepackage[scaled,osf]{garamondx}% scaled to 95%, my oldstyle
\usepackage[osfI]{garamondx}% traditional oldstyle
\usepackage[osfI,swashQ]{garamondx}% traditional oldstyle, all Q rendered as swash Q
```

6 Text effects under fontaxes

This package loads the `fontaxes` package in order to access italic small caps. You should pay attention to the fact that `fontaxes` modifies the behavior of some basic \LaTeX text macros such as `\textsc` and `\textup`. Under normal \LaTeX , some text effects are combined, so that, for example, `\textbf{\textit{a}}` produces bold italic *a*, while other effects are not, eg, `\textsc{\textup{a}}` has the same effect as `\textup{a}`, producing the letter *a* in upright, not small cap, style. With `fontaxes`, `\textsc{\textup{a}}` produces instead upright small cap *a*. It offers a macro `\textulc` that undoes small caps, so that, eg, `\textsc{\textulc{a}}` produces *a* in non-small cap mode, with whatever other style choices were in force, such as bold or italics.

7 Superior figures

The TrueType versions of GaramondNo8 have a full set of superior figures, unlike their PostScript counterparts. The superior figure glyphs in regular weight only have been copied to `NewG8-sups.pfb` and `NewG8-sups.afm` and provided with a `tfm` named `NewG8-sups.tfm` that can be used by the `superiors` package to provide adjustable footnote markers. See `superiors-doc.pdf` (you can find it in \TeX Live by typing `texdoc superiors` in a Terminal window.) The simplest invocation is

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

8 Glyphs in TS1 encoding

The layout of the TS1 encoded Text Companion font, which is fully rendered in this package, is as follows. See below for the macros that invoke these glyphs. Though shown in regular weight, upright

shape only, the glyphs are available in all weights and shapes.

	'0	'1	'2	'3	'4	'5	'6	'7	
'00x	\ 0	' 1	^ 2	~ 3	¨ 4	" 5	° 6	˘ 7	"0x
'01x	˘ 8	˘ 9	˙ 10	˘ 11	˘ 12	˘ 13	14	15	
'02x	16	17	„ 18	19	20	■ 21	■ 22	23	"1x
'03x	← 24	→ 25	ˆ 26	ˆ 27	ˆ 28	ˆ 29	30	31	
'04x	ƒ 32	33	34	35	\$ 36	37	38	' 39	"2x
'05x	40	41	* 42	43	˘ 44	= 45	• 46	/ 47	
'06x	0 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	˘ 129	¨ 130	¨ 131	† 132	‡ 133	∥ 134	‰ 135	"8x
'21x	• 136	°C 137	\$ 138	¢ 139	f 140	© 141	¥ 142	₤ 143	
'22x	⊕ 144	ℙ 145	£ 146	℞ 147	? 148	ı 149	đ 150	™ 151	"9x
'23x	‰ 152	¶ 153	฿ 154	№ 155	‰ 156	€ 157	◊ 158	SM 159	
'24x	{ 160	} 161	¢ 162	£ 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	

LIST OF MACROS TO ACCESS THE TS1 SYMBOLS IN TEXT MODE:

(Note that slots 0–12 and 26–29 are accents, used like `\t{a}` for a tie accent over the letter a. Slots 23 and 31 do not contain visible glyphs, but have heights indicated by their names.)

- 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

```

There is a macro `\textcircled` that may be used to construct a circled version of a single letter using `\textbigcircle`. The letter is always constructed from the small cap version, so, in effect, you can only construct circled uppercase letters: `\textcircled{M}` and `\textcircled{m}` have the same effect, namely \textcircled{M} .

9 Implementation details

9.1 Small Cap fonts

The small cap fonts were created from the capitals A–Z using FontForge to scale the sizes down uniformly to 67%, then boosting the horizontal and vertical stems up by 130%. The results provided a rough basis for the individual adjustments that had to be made to each glyph. Using FontForge, the stems were resized appropriately, often requiring a reworking of the shape. The end results are the only possible description of those transformations. Following the creation of those glyphs, appropriate metrics were created using FontForge, the end results of which are provided. The regular weight, upright shape, has been reworked much more than other weights, and looks considerably better, in my opinion. Making a small cap font from scratch takes some real work to get the glyphs, the metrics and the kerning right. In both the regular and bold upright shapes, standard accented glyphs are provided, as well as some special characters and `a_e` and `o_e` ligatures and the glyphs `lslash` and `oslash`.

The small cap macro `\textsc` cooperates with `\textbf` and `\textit`, so you may use, for example:

```
\textsc{Caps and Small Caps}
```


to produce *CAPS AND SMALL CAPS*,

```
\textit{\textsc{Caps and Small Caps}}
```

to produce *CAPS AND SMALL CAPS*,

```
\textbf{\textsc{Caps and Small Caps}}
```

to produce **CAPS AND SMALL CAPS**, and

```
\textbf{\textit{\textsc{Caps and Small Caps}}}
```

to produce *CAPS AND SMALL CAPS*.

9.2 Old style figures

The old style figures were created based on the existing lining figures, reducing the stem lengths of 0 and 1 to lower-case size using FontForge, and lowering the vertical positions of others. The shapes were then modified in FontForge to have more of a traditional oldstyle appearance — the end results show the transformations involved.

9.3 Text Companion glyphs

To provide full versions of the TS1 glyphs, a number of glyphs (tie accents, born, died, married, divorced, referencemark, numero, discount, estimated, copleft, centoldstyle) were adapted from Computer Modern, though with weights appropriate to garamondx. The other glyphs were copied virtually from a small modification of glyphs from txfonts, which has an essentially full rendition of the Text Companion glyphs in all weights/styles.

10 Matching math packages

Paul Pichaureau’s mathdesign package has an option `garamond` that sets text to `ugm` and math to his package that matches `ugm`. To use his math package with `garamondx` you write

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

Another possibility is to use the `garamondx` option to `newtxmath`, which uses `garamondx` upper and lower cases italic letters, properly metrized for math, in place of the default Times italics. This requires version 1.06 or higher of the `newtxmath` package.

```
\usepackage[full]{textcomp}  
\usepackage{garamondx} % defaults to lining figures, good for math  
\usepackage[varqu,varl]{zi4} % typewriter font inconsolata  
\usepackage[sf]{libertine} % biolinum as sans-serif  
\usepackage[garamondx,cmbraces]{newtxmath}  
\useosf % changes figure style in garamondx to osf for text, not math
```

Note that the last command, as well as its companion `\useofI`, may only be used in the preamble, and must not precede `\usepackage{garamondx}`.

11 License

The fonts in this package are derived from the (URW)+ + GaramondNo8 fonts which were released under the AFPL, and so the same holds for these fonts. The other support files are subject to the LaTeX Project Public License. See

<http://www.ctan.org/tex-archive/help/Catalogue/licenses.lpl.html>
for the details of that license.

The package and font modifications described above are Copyright Michael Sharpe, msharpe@ucsd.edu, October 1, 2013.

11.1 Font files covered by the 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-Osf-reg.pfb
NewG8-Reg-SC.afm
NewG8-Reg-SC.pfb
NewG8-Reg.afm
NewG8-Reg.pfb
NewG8-sups.afm
NewG8-sups.pfb