The drm Font Package, v4.4

Donald P. Goodman III

August 18, 2015

Abstract

The drm package provides access to the DRM (Don's Revised Modern) family of fonts, which includes a variety of optical sizes in Roman, italic, and small caps, along with a set of symbols and ornaments. It is intended to be a full-body text font, but its larger sizes can also be used for simple display purposes, and its significant body of symbols can stand on its own. It comes complete with textual ("old-style") and lining figures, and even has *small-caps figures*, along with superior and inferior figures. It also comes with extensible decorative rules to be used with ornaments from itself or other fonts, along with an extremely flexible ellipsis package. Decorative initials are also provided by means of METAPOST macros and superimposed figures; these are flexible in color, size, and style.

Contents

1 Introduction

After some time of involvement with TeX and LaTeX your author finally bothered to go read *The TeXbook* and *The METAFONTbook*. This latter closes, of course, with Donald E. Knuth's famous exhortation to "Go forth now and create *masterpieces of digital typography!*" This call to arms stirred a longing to actually do so in my soul.

I had some experience with the METAFONT language through my work with METAPOST, so I thought I might try my hand at it. I started in fits and stops some years ago, and only a few months ago took up the cause again in earnest. I found that, as Knuth also warned,

> Warning: Type design can be hazardous to your other interests. Once you get hooked, you will develop intense feelings about letterforms; the medium will intrude on the messages that you read. And you will perpetually be thinking of improvements to the fonts that you see everywhere, especially those of your own design.

Truer words were never spoken.

This document is typeset in accordance with the docstrip utility for automatically extracting package code and documentation.

2 License(s)

The TeX and LaTeX code in this package is licensed under the LaTeX Project Public License v1.3c, the details of which can be found in Appendix B on page 53. It's a legal document, and bears all the concomitant complications of such. The basic import is that you can use and distribute these files as you will, provided only that you do not restrict their use by their

recipients; and that you can even modify them as you will, provided that if you distribute your modifications, you do so under a different name.

The fonts themselves are licensed under the SIL Open Font License, v1.1, the details of which can be found in Appendix C on page 58. It's a less complex legal document, but a legal document all the same. The basic import is that you can't sell the fonts all by themselves (why anyone would pay for them anyway is beyond me, but there it is); you can distribute original or modified versions of the fonts otherwise however you wish, as long as you keep the copyright notice and license with it; and if you distribute a modified version, that you do so under a different name; that you not use the name of the font designer to promote a modified version; and that any modified versions of the fonts must be kept under the same license.

Finally, the software I used to build things, which is very simple and mostly uninteresting stuff, is also made available, under the GNU General Public License v3. This is sufficiently well-known that it's not duplicated in this document; but the text is, of course, included in the distribution.

That said, I'm pretty easy-going about this sort of thing; so if for some reason the above terms don't suit you, feel free to contact me and see if we can work something else out. But honestly, the terms of these licenses are more than fair, and it's hard for me to see a reason to depart from them.

3 Usage

3.1 Basic Usage

Using the DRM fonts is beyond easy; just include the following in your preamble:

```
\usepackage{drm}
```

And you're done! This makes the DRM fonts the default for your document, and defines appropriate commands for using them. I've made every effort to make using DRM as unsurprising as possible, so the commands you'd use to change sizes, styles, shapes, and so forth should all work as expected. That said, there are some unusual shapes and options available, and these are explained below.

You do *not* need to load `textcomp`; all those symbols and more are available from `drm`.

3.2 Package Options

DRM doesn't offer too many options, because too many aren't really needed; but it does allow some control over what defaults it resets.

`typeone` **typeone** The `typeone` option will probably be used most of the time that `drm`

itself is used; it forces drm to use un-rasterized outlines rather than bitmapped pk files, letting the pdf viewer do the rendering. Given that most pdf viewers have an awfully hard time decently displaying prerendered bitmap fonts, the typeone option will often be useful.

nodefault **nodefault** The nodefault option means that drm will not change any of the defaults of the document; that is, loading drm with the nodefault option should have no effect whatsoever on the appearance of your document. The fonts are defined, though, so you can use them if you decide you want to. Symbols will *not* be redefined.

nodefaulttext **nodefaulttext** The nodefaulttext option means that text fonts are *not* redefined but math fonts *are*. Symbols will *not* be redefined.

nodefaultmath **nodefaultmath** The nodefaultmath option means that text fonts *are* redefined but math fonts are *not*. Symbols *will* be redefined.

symbolsonly **symbolsonly** The symbolsonly option defines all the commands for the symbols (note that this may overwrite certain default command names, like \textcopyright; if you need these undefined, load textcomp *after* drm).

The default is that none of these are selected; that is, the default is that both text and math fonts, along with symbols, are redefined to be DRM. Commands which are font-independent, like \tulipframe and \extrule, are always defined when the package is loaded. Also, the fonts themselves are always defined, so they can be accessed directly even if they are not the default.

\drmsym Note that even if symbols are not redefined, they are still available directly through the \drmsym command. This command takes one argument, typically a \char directive, which will be the decimal, octal (if preceded by '), or hexadecimal (if preceded by ") position of the desired symbol in the font. E.g.:

```
\drmsym{\char'117}   \drmsym{\char"4F}   \drmsym{\char79}
```

yields

○ ○ ○

3.3 Interaction with Other Packages

As far as your author has been able to tell, drm has no adverse reactions with any other packages. A few notes are probably appropriate, however.

textcomp You do *not* have to load textcomp when you're loading drm; drm defines all the symbols in textcomp, and then some, allowing access to them with the same commands. This is the drmsym font, which is encoded, like the textcomp font, as TS1. If you're not loading the symbols, though (e.g., you've loaded drm with options nodefault or nodefaulttext), you may still want to load textcomp.

lettrine The lettrine package is used to typeset large dropped capitals at the beginning of paragraphs; it's an extraordinarily flexible and well-designed package. drm works just fine with it; however, the proportions of the letters

make a small tweak advisable. If you'll be using lettrines larger than two lines high, the following will be helpful:

```
\setlength{\DefaultFindent}{2pt}
```

This will prevent your text from bumping into your lettrine.

greek-fontenc The `greek-fontenc` package must be installed for `drm` to work properly.

cbfonts-fd The `cbfonts-fd` package must also be installed for `drm` to work properly.

3.4 Further Work Needed

While I'm quite happy with DRM right now, there are a few notable places where it needs some additional work.

Kerning The kerning is sometimes suboptimal. There really isn't much else to say about this. For most of the fonts, the kerning is reasonably good (at least, in my opinion), but for upright italic, boldface, and occasionally small and titling caps, I do still find lacunæ in my handling of certain kerning pairs.

Internal Code While the code is parameterized enough that, for example, boldfacing was a relatively simple process, it could use some improvement in this. Also, some code was repeated that would surely be better off included in macros, especially the placing of accents.

Decorative Initials I love, love, *love* decorative initials, and want DRM to have them. But writing them is a *lot* of work. I decided I wanted the fonts as they stand done before I get to work on those; but it's still further work that needs to happen.

More Ornaments I'm pretty happy with what ornaments I've designed for DRM, but it needs more of them. A full, 8-bit ornamental font is in the works (the decorative initials will likely be A–Z in this font), but designing these is a similarly large amount of work, so it's still on the burners.

Greek Fonts DRM badly needs real Greek fonts. After I did the math fonts, Greek fonts seemed like a short step; but now I really need to add italic and boldface versions, and optically size the upright ones.

In addition to these specific needs, font metrics may still change, though only slightly, and shapes are subject to tweaking here and there. But even now, the DRM fonts are usable, reasonably complete, and (in my view, at least) attractive.

4 About the DRM Fonts

So I've been plumbing the depths of alphabet design, and having a great time doing it. The result is what you see before you, the DRM fonts.

Roman	*Italic*	Roman	*Italic*	Roman	*Italic*
fi	*fi*	ff	*ff*	fl	*fl*
fj	*fj*	ffi	*ffi*	ffl	*ffl*
ft	*ft*	Th	*Th*		

Table 1: Ligatures in the DRM fonts.

4.1 About the Fonts

They're not *modern*, per se, but they do have modern characteristics, most especially the distinction between thick and thin strokes and the vertical orientation. They have a number of old-style characteristics, as well, though, like the aforementioned "Q" tail, the relatively prominent serifs, and the slight but still present brackets.

Overall, they're fairly dark fonts on average, as well as fairly wide. To my eye, this makes them ideal for reading long passages.

They have some unusual features. For example, they have not only the standard run of f-ligatures (fi, fl, ff, ffi, ffl), but also some unusual f-ligatures (ft, fj), as well as a non-f-ligature (Th). Some larger-size examples of the ligatures can be found in Table 1. DRM also contains some unusual shapes, such as upright italic and TITLING CAPS.

4.2 Alternate Glyphs

`\drmshortq`

While this section is titled in the plural, there is at present only one such: `\drmshortq`, which gives us "Q" rather than "Q." This is mostly useful for situations in which the "Q" is followed by some character with a descender (say, "Qp" as opposed to "Qp"), or when it is being used as a dropped capital and the extended tail would overwrite the text. (An enlarged dropped capital might have a tail underscoring the entire paragraph, which might actually look attractive.)

If for some reason you'd like to use "Q" all the time, and consign the admittedly somewhat baroque "Q" to the dustbin, you can do so by issuing the following commands:

```
\catcode`\Q=\active\def Q{\drmshortq}
```

Note that this involves some deep TEX magic, and command names containing the character "Q" will be broken by this. Fortunately, such commands are few and far between; `drm` does not contain any.

4.3 Font Families

DRM contains a full set of the normal font families you'd expect: roman, bold, italic, small caps, and so forth. But it also contains some shapes that are rather unusual, as well as a wide variety of sizes, forms, and weights capable of filling most needs.

4.3.1 Optical Sizing

The advent of digital fonts made many typographers lazy. Previously, of course, a printer could only print fonts in sizes that he had; each size had to be separately cut and designed. Digital fonts seemed to relieve this problem; now we can simply scale up or down, and only design a single size! Experience has shown, however, that this produces suboptimal results, as Table 2 demonstrates.

Fourteen point font is different
from scaled seven point font.

Table 2: Scaling and Optical Sizing Compared

The human eye, as it turns out, does not perceive the world, least of all letterforms, as geometrically scaled versions of larger or smaller shapes. For example, at small point sizes the eye tends to run adjacent strokes together, so proportionally wider letters and increased letterspacing are appropriate in smaller sizes but not in larger. For another example, strokes often overlap the technical top or bottom lines because a curved line will appear to be lower than a straight line at the same height. This effect diminishes at larger sizes; so this overshoot might be zero at double pica (twenty-four point), still significant at pica (twelve-point), and quite large at six-point. If we merely scaled the six point to get our twenty-four point, this overshoot would make the curved strokes look comically larger than the straight ones; if we did the opposite, then our curved letters would seem noticeably shorter than our straight ones.

The only real solution to this is to use *optical sizes*; that is, have a reasonable set of sizes which are designed for use at that particular size. LaTeX (largely transparently to the user, thanks to the magic of NFSS) will then select the closest optical size and scale as necessary from that. This minimizes the effects of scaling on the appearance of the font, and gives vastly superior results.

DRM offers a reasonable selection of optical sizes, at 6-, 7-, 8-, 9-, 10-, 11-, 12-, 14-, 17-, and 24-point sizes in roman, italic, slanted, small caps, titling caps, and upright italic. This variety should be sufficient for the vast majority of needs.

However, traditional printing has developed a vast array of standard sizes, with the quaint, colorful names that always go with traditional crafts. Setting text in Brevier Roman or Long Primer Italic means something very specific. LaTeX only offers a few default font size commands (e.g., `\small`, `\normalsize`,

Point	Traditional Name		DRM
	American	British	
3	Excelsior	Minikin	\excelsior, \minikin
4	Brilliant		\brilliant
4.5	Diamond		\diamondsize
5	Pearl		\pearl
5.5	Agate	Ruby	\agate, \ruby
6	Nonpareille		\nonpareille
6.5	Minionette	Emerald	\minionette, \emerald
7	Minion		\minion
8	Brevier, Petit, small text		\brevier, \petit, \smalltext
9	Bourgeois; Galliard		\bourgeois, \galliard
10	Long Primer; Corpus; Garamond		\longprimer, \corpus, \garamond
11	Small Pica; Philosophy		\smallpica, \philosophy
12	Pica		\pica
14	English; Mittel; Augustin		\english, \mittel, \augustin
16	Columbian	Two-line Brevier	\columbian, \twolinebrevier
18	Great Primer		\greatprimer
20	Paragon		\paragon
21	Double Small Pica		\doublesmallpica
22	Double Small Pica	Double Pica	\doublesmallpicaus, \doublepicabrit
24	Double Pica	Two-line Pica	\doublepica, \twolinepica
28	Double English	Two-line English	\doubleenglish, \twolineenglish
30	Five-line Nonpareil		\fivelinenonpareil
32	Four-line Brevier		\fourlinebrevier
36	Double Great Primer	Two-line Great Primer	\doublegreatprimer, \twolinegreatprimer
44	Meridian	Two-line Double Pica; Trafalgar	\meridian, \twolinedoublepica, \trafalgar
48	Canon; Four-line		\canon, \fourline
60	Five-line Pica		\fivelinepica
72	Inch		\inch

Table 3: Traditional size names, both American and British, with their corresponding point sizes and DRM command names.

etc.), and DRM leaves those unchanged, as users expect specific things to happen when they issue those commands. However, DRM does offer those traditional size names as commands, as well, giving quite a bit more breadth in font size choice than the default before one must resort to explicit `\fontsize` commands. Table 3 on page 8 lists these commands by name; where there is a name unique to British typesetting that differs from the American name, both are offered as equivalents.

4.3.2 Small Caps

In the first place, it's important to have what typographers call "real" small caps, not "faked" small caps. There is a real and noticeable difference between the two. Real small caps are designed for a particular size; the stroke widths match, the spacing is appropriate, and so forth. Faked small caps are produced merely by scaling down normal capital letters for a given size, which produces inferior results. The two types are compared in Table 4.

THESE ARE REAL SMALL CAPS.

THESE ARE FAKED SMALL CAPS.

Table 4: Real and faked small caps compared.

Plainly, the results of real small caps are far superior, and faked ones should only be employed when the typographer has no small caps available, and possibly not even then.

DRM has, of course, a full set of real small cap fonts, which are appropriately scaled. But DRM goes even further than this, offering both *small caps* and *petite small caps*. DRM, though, considers petite small caps to be the normal type, and therefore refers to these as *small caps* and *titling caps*. The distinction between these two probably bears some explanation.

In Anglo-American typography small caps are typically a bit larger than the ex-height; in other countries, they are typically equal to the ex-height.[1]

I see advantages in both approaches. So-called "petite" small caps look great in running text but seem rather squashed in titles and headings; larger small caps look better in titles and headings (they maintain the gravity of all-caps without the impression of shouting, a rather common impression here in the Internet age) but are simply too large to blend well with normal lowercase text. So DRM offers both; NORMAL SMALL CAPS, accessed via the normal LaTeX `\textsc` and `\scshape` commands, are "petite" small caps; Anglo-American large small caps are available as TITLING SMALL CAPS, via the commands `\texttc` and `\tcshape`. (These stand, transparently enough, for "titling caps.") An example of the difference, which may give further ideas for the appropriate uses for each, is in Table 5.

`\texttc`
`\tcshape`

[1] These are sometimes called *petite small caps* among Anglo-American typographers.

THE ADVENTURES OF ROBINSON CRUSOE TITLING CAPS	THE ADVENTURES OF ROBINSON CRUSOE SMALL CAPS

Table 5: Small caps and titling caps compared.

\textitsc
\textittc
\itscshape
\ittcshape

Both small caps and titling caps come in *italic* (really simply *slanted*) varieties; these are accessed via `\textitsc` and `\textittc` (or `\itscshape` and `\ittcshape`). However, it should be noted that both these font shapes are designed for emphasizing text, as is slanting; so while it's perfectly possible to *SLANT YOUR SMALL CAPS* and *ITALICIZE YOUR TITLING CAPS*, this really should be done with great caution.

4.3.3 Slanted, Italic, and Upright Italic

DRM also offers the usual *italic* type, typically used for titles of certain types of works, emphasis, and similar functions; and the *slanted* type, which is usually simply a poor-man's italic in fonts which don't have a real italic, but occasionally does find its uses in full-featured fonts.

Non-typographers often think that italic type is simply normal type slanted to the right, but that's not really the case. A comparison is given in Table 6.

This is slanted text.
This is italic text.

Table 6: Slanted and italic text compared.

As Table 6 shows, true italic is much more than simply slanted roman. Many of the letterforms are quite different; the two-story "a" becomes a one-story "*a*," the straight "k" becomes a curved or (in DRM's case) looped "*k*," the double-looped "g" becomes a single-looped "*g*," and so forth.

So DRM offers true italic, often with some very ornate and, in the author's opinion, beautiful letterforms. The "*Q*" and "*J*" and particular favorites of his, but the somewhat unusual looped "*k*" and curled "*h*," along with some other shapes like "*2*" and "*g*," are also interestingly different from most other fonts, as well as visually striking in their own right.

DRM also offers upright italic, a face with the italic letterforms but not slanted at all. Some fonts have offered this as a difficult-to-access novelty, but as far the author knows DRM is the only one to offer it as a first-class citizen, accessed in the same way and just as easily as the more usual slanted italic shape. It is accessed via the commands `\textui` and `\uishape`. This shape is

\textui
\uishape

not commonly used in running text; it remains to be seen whether this is due to its being not useful, or simply to its being rarely easily available.

Your author can see certain uses for it; for example, when some font distinction is needed but no connotation of emphasis is desired, as in book titles. In any case, DRM makes it easy to use in the event that it is wanted.

4·3·4 Weights

Weight is the typographical term for what most folks call *boldface* type; however, the dimension can go much deeper than that. Fonts can be lighter than surrounding text as well as **heavier**, and heavier weights can often be **extended in width as well as heavier in weight**, or simply **heavier in weight without increasing its width**.

Some fonts take this to arguably absurd extremes, offering up to a dozen weights. I've never seen much sense in this, and consequently haven't gone to these lengths. Instead, DRM offers three weights: light, normal, and **bold**. As noted, bold fonts are often wider than medium weights, and are referred to as *bold extended*; DRM has a **bold extended**, as well. Lighter fonts are sometimes narrower, or *condensed*; DRM's light weight is not condensed, but rather normal width.

`\textl` `\lseries` `\textb` `\bseries` `\textbf` `\bfseries`

Light is achieved by `\textl` and `\lseries`; boldface by `\textb` and `\bseries`; bold extended by `\textbf` and `\bfseries`. Table 7 shows the differences between these weights in twelve-point size.

`\lseries`	This sentence shows one of DRM's weights.
`\mdseries`	This sentence shows one of DRM's weights.
`\bseries`	**This sentence shows one of DRM's weights.**
`\bfseries`	**This sentence shows one of DRM's weights.**

Table 7: DRM's font weights compared.

One will probably note that bold extended is actually bolder than normal bold (that is, `\bfseries` is bolder than `\bseries`). This is because the extra space means that it can be. This is probably not the way it should be; but I think that people expect at least that much boldness when selecting `\bfseries`, so I decided to make it that way.

Until v2.0, DRM did *not* offer a bold italic or a bold small caps; this is because both italics and small caps are already meant to serve for emphasis, and bolding your already emphasized text is really a bad idea. Bold italic I considered to be a particularly egregious typographical crime.

However, after some conversations and one example in which bold italic was actually used well (a display; *not* in running text), your author has tem-

pered his aggressive stance and provided a bold italic, bold upright italic, and bold versions of both small and titling caps. These are bold extended, and are available only in this one additional weight; this seemed appropriate given that their only appropriate use is displayed texts. They are accessed simply by requesting either bold or italic, and then requesting the other, like so:

```
\textbf{\textit{I hope you bold
italic types are happy now}}
```
I hope you bold italic types are happy now.

```
\textui{\textbf{I hope you bold
italic types are happy now}}
```
I hope you bold italic types are happy now.

```
\textsc{\textbf{Sometimes this
might help with displays.}}
```
SOMETIMES THIS MIGHT HELP WITH DISPLAYS.

```
\textbf{\texttc{Sometimes this
might help with displays.}}
```
SOMETIMES THIS MIGHT HELP WITH DISPLAYS.

There is still no bold italic small caps; I'll await an example of these being used appropriately before adding them.

4.4 Figures (Digits) (Numbers)

Typographers typically call them "figures"; mathematicians tend to prefer "digits"; most folks simply call them "numbers." Whatever we call them, there is a surprisingly large variety of ways to write them.

There are, at the very least, seven separate kinds of figures: textual, lining, tabular textual, tabular lining, small caps, superior, and inferior. Small caps figures are for some reason rather rare, and strictly speaking there could be tabular and non-tabular versions of them, as well, but I've never encountered them.

Textual figures, also called *old-style figures*, *lowercase figures*, or even *mediæval figures*, are the ones that look sort of . . . well, old-styled and lowercase. They are centered on the ex-height, like lowercase letters, and some have ascenders, some descenders, and some neither, like lowercase letters. They blend in with running text very well, whereas lining figures (which we'll get to in a moment) tend to stick out because they are all quite tall and often come in groups. Most commonly, "0," "1," and "2" have neither ascenders nor descenders; "6" and "8" have ascenders, and "3," "4," "5," "7," and "9" have descenders; DRM follows this typical scheme in its roman types. However, other systems have existed, particularly in France, where some famous fonts had an ascending rather than descending 3. DRM has a non-typical set in its italic fonts, with a descending 3 but an ascending 2: *0, 1, 2, 3, 4, 5, 6, 7, 8, 9*.

Textual figures are the default in DRM in the roman and italic fonts.

Lining figures, also called *titling figures* or *modern figures*, line up at the baseline and all have a common height, typically something close to the

height of capital letters or the ascenders of lowercase letters. They look like this: 0123456789. They're great when one wishes to draw extra attention to the figures, and practically mandatory when figures are being used with all caps; however, they throw off the color of the page and don't blend well with other running text. In DRM, we get lining figures by using the `\liningnums` command, which takes a single argument; namely, the number to be typeset in lining figures. Each individual number can be accessed by command, as well, of the form `\liningzero`, `\liningone`, and so forth.

`\liningnums`

Both of these types of figures can be *tabular* or not. This means, as a practical matter, *monospaced*; that is, with tabular figures each digit takes up an identical horizontal space. This is great for lining up numbers in columns, but produces rather bad spacing when used in running text.

In DRM, the default textual (old-style) figures *are not* tabular, while the lining figures *are* tabular. It is possible to have tabular textual figures and proportional lining figures, but I've never seen much sense in either, as it seems that they defeat the purposes of their own particular form.

DRM also has *small-cap figures*, a relative rarity in the typographical world. These are simply figures which match the style of the small caps fonts. Neither textual nor lining figures work well with small caps; lining figure are too tall, and textual figures' ascenders and descenders don't fit with the relatively straight lines of small caps text. So DRM has proportional (non-tabular), but short figures for small caps:

SMALL CAPS 0123 FIGURES.

Otherwise, it would like one of the following:

SMALL CAPS 0123 FIGURES.
SMALL CAPS 0123 FIGURES.

Neither of which looks very good. There are similar digits for titling caps. These are, of course, the defaults when using small or titling caps; if you need lining figures, you can still use `\liningnums`, and if you need textual figures, typeset them in normal roman text.

Finally, DRM provides *superior* and *inferior* figures. These are figures which are specially designed to appear in superscripted or subscripted text, respectively. These avoid text color and spacing problems from forming superior figures merely from scaling and raising normal figures. They look like the following:

^{30123456789}

3 0123456789

By default, in DRM (unless one of the `nodefault` options has been selected) footnote markers are made with superior figures. Otherwise, superior figures must be selected with either `\drmsupfigs`, or the option with a single argument, `\textdrmsupfigs`. (Inferior numerals are selected with `\drminffigs` and the single-argument option, `\textdrminffigs`.) Inferior figures are typically useful for chemical formulæ, but may conceivably find other uses, as well.

`\drmsupfigs`
`\textdrmsupfigs`
`\drminffigs`
`\textdrminffigs`

These special superior figures do end up looking significantly better than merely superscripted-and-scaled footnote labels:

…this fact.[6] As… | …this fact.^{6} As…

The superscripted and scaled version is too large, drawing more attention to itself than warranted (the purpose of a footnote is, after all, to provide citation without interrupting the flow of the text), not to mention that it protrudes above the height of the capitals and ascenders, making itself even more conspicuous; and the symbol itself is too thin, with lines almost spindly. The superior figure, on the right, tops off at the height of the ascenders, and is specially designed to have lines of the same width as the body font.[2] This ensures an overall better appearance when these figures are used.

Finally, as of v4.0, DRM offers *real dozenal characters*. In conjunction with the `dozenal` package (available, like `drm`, on CTAN), DRM will redefine the commands `\x` and `\e` to produce DRM dozenal characters, rather than the Computer Modernish characters of the `dozenal` package. By default, these are old-style; lining dozenal figures are also available.

0123456789ᘔ0 *0123456789ᘔ0*
0123456789ᘔ0 0123456789ᘔ0
0123456789ᘔ0 *0123456789ᘔ0*
0123456789ᘔ0 0123456789ᘔ0
0123456789ᘔ0 0123456789ᘔ0
0123456789ᘔ0 0123456789ᘔ0
0123456789ᘔ0 *0123456789ᘔ0*
$3^{0123456789ᘔ0}$ $3_{0123456789ᘔ0}$

[2] Or as near as is possible and attractive, anyway; the conscious design is better than the automatic solution.

In all the dozenal fonts, the "default" ten and eleven characters are held in the `X` and `E` slots; this means old-style in the roman and italic fonts, in all weights, but lining in the rest. In the roman and italic fonts, in all weights, lining figures are held in the `x` and `e` slots.

4.5 Symbols and Ornaments

One of DRM's strengths is its wide variety of symbols contained by default; rather than having to import separate fonts, or define macros to assemble common symbols out of their component parts, we can often simply use the symbols contained in DRM.

`\textnumero` `\textrefmark` `\textasterism` `\textradiation`

Starting with the staples of traditional typography like the numero (№) and the reference mark (※) to near-obsolete typesetting symbols like the asterism (⁂) to more unique symbols like the international sign for radiation hazards (☢), DRM has something for most needs.

Symbols of the DRM Font		
Religious Symbols		
✠	`\textcrusadecross`	"Crusader" cross
✡	`\textcrusadecrossoutline`	"Crusader" cross in outline
†	`\textlatincross`	Latin cross
☨	`\textlatincrossoutline`	Latin cross in outline
✚	`\textgreekcross`	Greek cross
✙	`\textgreekcrossoutline`	Greek cross in outline
✖	`\textsaltirecross`	Saltire cross; cross of St. Andrew
✗	`\textsaltirecrossoutline`	Saltire cross, cross of St. Andrew in outline
⚱	`\texteucharist`	Traditional representation of the Eucharist; chalice with Host and rays
✡	`\textstardavid`	Traditional Star of David
✸	`\textstardavidsolid`	Traditional Star of David, solid
✡	`\textstardavidoutline`	Traditional Star of David in outline
Genealogical Symbols		
★	`\textborn`	Symbol for born

†	\textdied	Symbol for died
⚯	\textdivorced	Symbol for divorced
⚭	\textmarried	Symbol for married
🍃	\textleaf	Leaf symbol
♂	\textmale	Symbol for male
♀	\textfemale	Symbol for female
	Intellectual Property Symbols	
®	\textregistered	Registered mark
™	\texttrademark	Trademark sign
℠	\textservicemark	Service mark sign
℗	\textsoundrecording	Sound recording sign
©	\textcopyright	Copyright mark
ↄ	\textcopyleft	Copyleft mark
	Astronomical Symbols	
☉	\textsun	Sun, Sol
🌣	\textsunvar	Variant Sun or Sol; with ray
☽	\textwaxcrescent	Waxing crescent moon
○	\textfullmoon	Full moon
☾	\textwanecrescent	Waning crescent moon
●	\textnewmoon	New moon
☿	\textmercury	Mercury; Hermes
⊕	\textearth, \textterra	Earth, Terra
♁	\textearthvar,\textterravar	Variant Earth, Terra
♂	\textmars	Mars, Ares
♀	\textvenus	Venus, Aphrodite
♃	\textjupiter	Jupiter, Jove, Zeus
♄	\textsaturn	Saturn
♅	\texturanus	Uranus
♅	\texturanusvar	Variant Uranus
♆	\textneptune	Neptune, Poseidon
♇	\textpluto	Pluto
♇	\textplutovar	Variant Pluto
⚳	\textceres	Ceres

⚴	`\textpallas`	Pallas
⚵	`\textjuno`	Juno, Hera
⚶	`\textjunovar`	Variant Juno, Hera
⚳	`\textvesta`	Vesta
⚶	`\textvestavar`	Variant Vesta
⚵	`\textastraea`	Astræa
⚶	`\textastraeavar`	Variant Astræa
⚲	`\texthebe`	Hebe
⚢	`\textiris`	Iris
♈	`\textaries`, `\textari`	Aries
♉	`\texttaurus`, `\texttau`	Taurus
♊	`\textgemini`, `\textgem`	Gemini
♋	`\textcancer`, `\textcnc`	Cancer
♌	`\textleo`	Leo
♍	`\textvirgo`, `\textvir`	Virgo
♎	`\textlibra`, `\textlib`	Libra
♏	`\textscorpius`, `\textsco`	Scorpius
♐	`\textsagittarius`, `\textsgr`	Sagittarius
♑	`\textcapricorn`, `\textcap`	Capricorn
♒	`\textaquarius`, `\textaqr`	Aquarius
♓	`\textpisces`, `\textpsc`	Pisces
★	`\textstar`	Star
☄	`\textcomet`	Comet
□	`\textquadrature`	Quadrature
☍	`\textopposition`	Opposition
☌	`\textconjunction`	Conjunction
☊	`\textascendingnode`	Ascending node
☋	`\textdescendingnode`	Descending node

Currency Symbols

$	`\textdollarsign`	Dollar sign
$	`\textolddollarsign`	Old-style dollar sign; double-slashed dollar sign
¢	`\textcentsign`	Cent sign
¢	`\textoldcentsign`	Old-style cent sign; diagonally slashed cent sign

£	`\textpoundsterling`	British pound sterling sign
₤	`\textoldpoundsterling`, `\textlira`	Old-style British pound sterling sign; double-slashed British pound sterling sign; Italian lira sign
€	`\texteuro`	Euro sign
¥	`\textyen`	Japanese yen sign
฿	`\textbaht`	Thai baht sign
₡	`\textcolon`	Costa Rican, Salvadoran colon sign
₫	`\textdong`	Vietnamese dong sign
ƒ	`\textflorin`	Florin sign
₲	`\textguarani`	Uruguayan guarani sign
₦	`\textnaira`	Nigerian naira sign
₱	`\textpeso`, `\textruble`	Mexican peso sign; Russian ruble sign
₩	`\textwon`	Won sign
¤	`\textcurrency`	Generic currency
	Roman Numerals	
I	`\romone`	Roman numeral one
V	`\romfive`	Roman numeral five
X	`\romten`	Roman numeral ten
L	`\romfifty`	Roman numeral fifty
C	`\romhundred`	Roman numeral hundred
D	`\romfivehundred`	Roman numeral hundred
M	`\romthousand`	Roman numeral thousand
MDCLI	`\romanize{1668}`	Convert Indo-arabic numeral to Roman numerals
	Lining Numerals	
0	`\liningzero`	Lining numeral 0

1	`\liningone`	Lining numeral 1
2	`\liningtwo`	Lining numeral 2
3	`\liningthree`	Lining numeral 3
4	`\liningfour`	Lining numeral 4
5	`\liningfive`	Lining numeral 5
6	`\liningsix`	Lining numeral 6
7	`\liningseven`	Lining numeral 7
8	`\liningeight`	Lining numeral 8
9	`\liningnine`	Lining numeral 9
3091	`\liningnums{3091}`	Convert figures into lining figures

Traditional and Innovative Typography

№	`\textnumero`	Numero
※	`\textrefmark`	Reference mark
⁂	`\textasterism`	Asterism
ª	`\textfemininеordinal`	Feminine Ordinal
º	`\textmasculineordinal`	Masculine Ordinal
¹	`\textsupone`	Superscript 1; superior digit 1
²	`\textsuptwo`	Superscript 2; superior digit 2
³	`\textsupthree`	Superscript 3; superior digit 3
¶	`\textpilcrowsolid`	Solid-lined pilcrow
¶	`\textpilcrowoutline`	Outlined pilcrow
§	`\textsection`	Section mark
†	`\textdagger`, `\textdag`, `\dag`	Dagger
‡	`\textdbldagger`, `\textdbldag`, `\dbldag`	Double dagger
‖	`\textpipe`	Pipe
¦	`\textbrokenpipe`	Broken pipe
℞	`\textrecipe`	Recipe mark
‽	`\textintbang`	Interrobang
⸘	`\textopenintbang`	Opening interrobang

Text-mode Math Symbols

′	`\textprime`	Single prime mark
″	`\textdoubleprime`	Double prime mark
‴	`\texttripleprime`	Triple prime mark
√	`\textsqrt`	Square root sign; radical
¼	`\textquarter`	One-quarter fraction, slanted
½	`\texthalf`	One-half fraction, slanted
¾	`\textthreequarters`	Three-quarters fraction, slanted
⅓	`\textthird`	One-third fraction, slanted
⅔	`\texttwothirds`	Two-thirds fraction, slanted
‰	`\textperbiqua`, `\textpermille`	Perbiqua, permille, per thousand
‱	`\textpertriqua`, `\textpertenmille`	Pertriqua, per ten thousand
=	`\textequals`	Equals sign
/	`\textslash`	Forward slash
×	`\texttimes`	Multiplication; times
÷	`\textdiv`	Division sign
↑	`\textuparrow`	Upward-pointing arrow
↓	`\textdownarrow`	Downward-pointing arrow
←	`\textleftarrow`	Left-pointing arrow
→	`\textrightarrow`	Right-pointing arrow

Warning Signs

☢	`\textradiation`	Radiation warning sign
☢	`\textradiationnocircle`	Radiation warning sign, no enclosing circle
☣	`\textbiohazard`	Biohazard warning sign

☣	`\textbiohazardnocircle`	Biohazard warning sign, no enclosing circle
⚠	`\texthighvoltage`	High voltage warning sign
ϟ	`\texthighvoltagenotriangle`	High voltage warning sign, no enclosing triangle
⚠	`\textgeneralwarning`	General warning sign
	Bullets and Other Marks	
•	`\textbullet`	Solid circular bullet
○	`\textopenbullet`	Open circular bullet
♥	`\textheart`	Solid heart
♡	`\textopenheart`	Open heart
♪	`\texteighthnote`	Eighth note
◆	`\textdiamond`	Solid diamond; solid lozenge
◇	`\textopendiamond`, `\textlozenge`	Open diamond; open lozenge
°	`\textdegree`	Degree symbol
℃	`\textdegreec`	Degrees Celsius
~	`\texttilde`, `\tilde`	Tilde
	Ornaments and Fleurons	
❧	`\textrightupfleuron`	Rightward-pointing, upward fleuron
❧	`\textrightdownfleuron`	Rightward-pointing, downward fleuron
❧	`\textleftupfleuron`	Leftward-pointing, upward fleuron
❧	`\textleftdownfleuron`	Leftward-pointing, downward fleuron
❧	`\textupleftfleuron`	Upward-pointing, leftward fleuron
❧	`\textuprightfleuron`	Upward-pointing, rightward fleuron
❧	`\textdownrightfleuron`	Downward-pointing, rightward fleuron

\textdownleftfleuron

Downward-pointing, leftward fleuron

\textsquaretulip

Square of four tulips, facing up and down

\textsquaretulipside

Square of four tulips, facing left and right

\textupdoubletulip

Double tulips, facing upward

\textdowndoubletulip

Double tulips, facing downward

\textrightdoubletulip

Double tulips, facing righward

\textleftdoubletulip

Double tulips, facing leftward

\textupleftcornertulip

Single corner-facing tulip, for upper left corners

\textuprightcornertulip

Single corner-facing tulips, for upper right corners

\textlowleftcornertulip

Single corner-facing tulips, for lower left corners

\textlowrightcornertulip

Single corner-facing tulip, for lower right corners

\textupsingletuliplong

Single tulip, upward-facing

\textdownsingletuliplong

Single tulip, downward-facing

\textleftsingletuliplong

Single tulip, leftward-facing

	`\textrightsingletuliplong`	Single tulip, rightward-facing
	`\textupsingletulip`	Single tulip, upright
	`\textdownsingletulip`	Single tulip, downward
	`\textleftsingletulip`	Single tulip, leftward
	`\textrightsingletulip`	Single tulip, rightward
	`\spearright`	Rightward-pointing spear head
	`\spearleft`	Leftward-pointing spear head
=	`\horizspearext`	Extension piece for horizontal shafts
	`\spearup`	Upward-pointing spear head
	`\speardown`	Downward-pointing spear head
‖	`\vertspearext`	Extension piece for vertical shafts
	`\fleurdelis`, `\fleurdelys`	Fleur-de-lis
	`\fleurdelisdown`, `\fleurdelysdown`	Fleur-de-lis, downward
	`\fleurdelisleft`, `\fleurdelysleft`	Fleur-de-lis, leftward
	`\fleurdelisright`, `\fleurdelysright`	Fleur-de-lis, rightward
	`\woundcordleftext`	Wound cord, leftward facing, extender
	`\woundcordrightext`	Wound cord, rightward facing, extender
	`\woundcordleftend`	Wound cord, left end

	`\woundcordrightend`	Wound cord, right end
	`\woundcordleftendinv`	Wound cord, left end, inverted
	`\woundcordrightendinv`	Wound cord, right end, inverted

These ornaments are often quite useful for decorative purposes, though textual ornaments are too often neglected these days. (The sturdy `adforn` and intricate `psvectorian` packages for LaTeX are notable and admirable exceptions.) The possibilities with even just a few decorative shapes are endless.

4.6 Special Symbol and Ornamental Commands

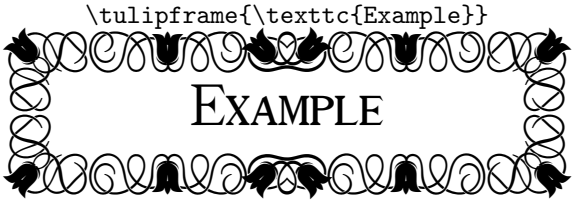
Because typography is an ancient art full of arcane knowledge, there are some things that simply won't fit into the general rules. As a result, DRM offers a few interesting tidbits that your author hasn't found, or hasn't found useful, elsewhere. We start with a few commands for using the textual ornaments DRM provides, followed by some more mundane but still useful typographical tools.

4.6.1 Ornamental Commands

Having just mentioned the great decorative utility of old-fashioned textual ornaments,[3] it would be remiss not to offer some tools for actually using such ornaments short of entering them in and designing interlocking boxes by hand. Ornaments, being inherently decorative rather than systematic, are not always subject to automation; but some limited applications can be, and DRM tries to offer some help with them.

`\tulipframe` DRM offers `\tulipframe`, which frames a title in decorative tulip fleurons:

```
\tulipframe{\texttc{Example}}
```



The nature of the tulip fleurons in DRM's symbol font is such that these frames can be extended or shrunk as one wishes. `\tulipframe`, alas, is not that intelligent; it doesn't grow or shrink with the text, but simply sits as it is. Doing better than this will require box-fiddling by hand. However, since DRM offers vertical *and* horizontal tulip ornaments, it's possible to have ornamental frames of any height or width.

[3] *See supra* at 20.

DRM also offers an extremely flexible *rule system*, allowing the creation of vertical and horizontal rules of any length, out of any characters, in the beginning, the middle figures, and the end. Meet `\extrule`, or *extensible rule*, which can produce rules with whatever characters you'd like.

`\extrule`

`\extrule` requires five arguments, as shown below:

`\extrule` {⟨*orient*⟩} {⟨*len*⟩} {⟨*start*⟩} {⟨*end*⟩} {⟨*ext*⟩}

orient The rule's *orientation*. This can take the value {⟨*h*⟩}, for *horizontal*, or {⟨*v*⟩}, for *vertical*.

len The rule's *length*. This will be the total length of the rule, including the start and end characters. It can be passed in any form understood by eTEX's `\numexpr` and `\dimexpr`, meaning that you can give it formulæ, such as `{0.2\linewidth}`.

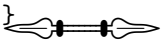
start The first character in the rule; this means either the left character in a horizontal rule, or the bottom character in a vertical rule.

end The last character in the rule, either the right in a horizontal or the top in a vertical.

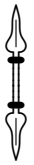
ext The extension character; this is the character which will be repeated until the rule is the appropriate length.

DRM offers several useful characters for producing such rules, which are designed to line up properly and thus produce attractive decorative rules. Among these are the *spear characters*, and we will demonstrate their use with a couple of sample rules:

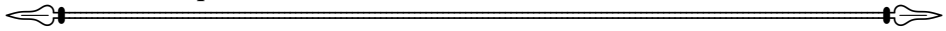
```
\extrule{h}{\linewidth/2}{\spearleft}
{\spearright}{\horizspearext}
```



```
\extrule{v}{\linewidth/6}{\speardown}
{\spearup}{\vertspearext}
```



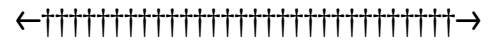
Of course, these are typically more useful when longer, as in the rule below, which is equal to the `\linewidth`:



As is evident, `\extrule` also suppresses indentation, which is almost certainly the right choice. If you want an indent with it, it's easy enough to put one in explicitly.

While characters like these, designed to line up correctly, are naturally the most likely candidates for such rules, you can use any characters you'd like, which can sometimes lead to some interesting choices:

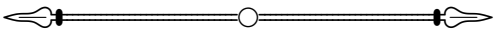
```
\extrule{h}{\linewidth/2}{\textleftarrow}{\textrightarrow}{\dag}
```



This is a pretty absurd example, of course, but it's likely that better ones could and will be devised.

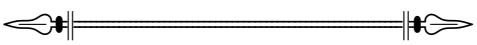
And what about when we desire a special character in the *middle* of the rule? Use *two* `\extrule`s and put the symbol you want in the middle between them; make sure you comment out the end of your first and second lines, so as not to introduce any extraneous spaces:

```
\extrule{h}{\linewidth/2}{\spearleft}{}{\horizspearext}%
\textbigcircle%
\extrule{h}{\linewidth/2}{}{\spearright}{\horizspearext}%
```

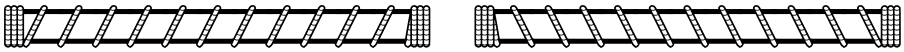


There's no reason we can't put more than one character into these slots, as well, if we want to mix them with some different characters:

```
\extrule{h}{\linewidth/2}{\spearleft\raisebox{0.8pt}{\textpipe}}%
{\raisebox{0.8pt}{\textpipe}\spearright}{\horizspearext}%
```



Some other useful characters for decorative rules are the "wound cord" characters:



Finally, these rules can often form very dignified page borders. The border on this page, for example, was formed very simply by the following (using `\usepackage[absolute]{textpos}`):

```
\setlength{\TPHorizModule}{\linewidth}
\begin{textblock}{1}(0.22,2)
\extrule{v}{\textheight/6*7}{\textbigcircle}{\spearup}{\vertspearext}%
\hskip-1.3em%
\extrule{h}{8*\textwidth/6}{}{\spearright}{\horizspearext}%
\end{textblock}
```

This takes a little hand-tuning (e.g., the `\hskip` prior to the horizontal `\extrule`, and the offset in the parentheses), but once done, it can look quite nice.

4.6.2 Ellipses

DRM also has some unreasonably configurable ellipses. Your author included these because he's often been displeased by the default ellipsis options. (Of course, there is the excellent `ellipsis` package; but why not fix the problem here, when I've got the chance?) DRM offers two ellipsis commands, `\drmelip`, which gives a three-dot ellipsis, and `\drmfelip`, which gives a four-dot ellipsis.

`\drmelip`
`\drmfelip`

I was always taught then when an ellipsis occurs after a period, four dots should be used, the first dot being the period itself and the next three being the ellipsis. However, using `\ldots` and similar commands after a period always seems to result in spacing that was subtlely (or not-so-subtlely) off. So DRM tries to fix that problem with these commands.

The default behavior of the two:

`\drmelip`	`Trying out\drmelip the ellipsis.` Trying out…the ellipsis.
`\drmfelip`	`\drmelip and so on\drmfelip` …and so on….

Table 9: A demonstration of DRM's two types of ellipses.

It goes without saying, of course, that these ellipses won't break across lines.

There are four parameters that govern how these ellipses actually appear: the space before the ellipsis starts, the space in between the ellipsis characters, the space after the ellipsis ends, and the character used for the ellipsis. Each of these parameters are configurable.

`\drmelipgap` is a LaTeX length which determines how much space is between each ellipsis character; reset it, if you like, with the standard `\setlength` command. By default, it is just under three points (2.9, to be precise.)

`\drmelipgap`

`\drmelipbef` and `\drmelipaft` are, as the names imply, the lengths which govern the amount of space before and after the ellipsis. Reset them with the LaTeX `\setlength` command. By default, they are 2.4 points and 1.4 points, respectively.

`\drmelipbef`
`\drmelipaft`

Finally, the `\drmelipchar` macro tells LaTeX what character is used for the ellipsis. By default, this is ., but it can be `\def`ed or `\renewcommand`ed to be anything you like. Always wanted an ellipse made out ampersands for some reason? Or perhaps one made out of daggers?

`\drmelipchar`

```
\def\drmelipchar{\dag}\drmelip
```

†††

It's probably wise not to abuse this, but it's good for a little fun sometimes, and it's easier to use (though obviously much less flexible) than TEX's `\dotfill` incantations.

It is occasionally useful, however; e.g., some legal writing makes ellipses out of asterisks:

```
\def\drmelipchar{$^*$}The decision is hereby\drmelip reversed.
```

The decision is hereby * * * reversed.

So once in a while, we might actually be able to use this feature for something other than its novelty value.

4.6.3 Decorative Initials

DRM, as of v3.0, provides for decorative initials. These are not traditional decorative initials, however, with intricate patterns provided individually for each letter. They are, rather, formed with a single background pattern, with the necessary letter superimposed. The goal is to make the background pattern interchangeable. The color of that background pattern and the color of the foreground letter can be controlled separately. Despite the single background pattern, therefore, this provides for a remarkable degree of flexibility.

`\drmdecinit` `\drmdecinit` is the name of the game here, a command which takes five arguments, all of which are mandatory.

`\drmdecinit` {⟨*width*⟩} {⟨*height*⟩} {⟨*bgcolor*⟩} {⟨*fgcolor*⟩} {⟨*fgchar*⟩}

These are largely self-explanatory, so a few examples will likely do. Note that `drm` uses the excellent `gmp` package to get the METAPOST code to be part of the LaTeX code, allowing LaTeX to control significant parts of the formatting. This means that one will have to run a shell script along with compiling the document, similarly to `bibtex`, `makeindex`, or a host of others.

```
\lettrine[lines=4,nindent=0pt,
 findent=-1em]{\drmdecinit{40pt}
 {40pt}{blue}{(.625,0,0)}{L}}{orem
 ipsum}
```

OREM IPSUM dolor sit amet, consectetur adipiscing elit. Ut porttitor libero lacus, a rhoncus dolor finibus vel. Morbi volputate condimentum ornare. In scelerisque aliquam. . . .

```
\lettrine[lines=4,nindent=0pt,
 findent=-1em]{\drmdecinit{40pt}
 {40pt}{blue}{red}{L}}{orem ipsum}
```

OREM IPSUM dolor sit amet, consectetur adipiscing elit. Ut porttitor libero lacus, a rhoncus dolor finibus vel. Morbi volputate condimentum ornare. In scelerisque aliquam. . . .

The colors are METAPOST colors; unfortunately, this means that we can use only "black," "white," "red," "green," and "blue" by name. However, any valid

METAPOST color specification will work. In the example above, for example, to get a darker red, one could use `.4red`, or one could specify colors in RGB notation, as shown above on the left. Note that, when doing this latter, the parentheses are necessary.

These decorative initials lend themselves to some other, sometimes unexpected, uses. For example, decorative enumerates. It is best to use lining figures rather than textual figures for this.

`\drmdecinitfont` DRM offers the command `\drmdecinitfont`, which is the font which DRM uses for the decorative initials. Because of the internals of the `gmp` package, the simple name of the font can't be inserted here; it must be defined in a particular way. The default is, of course, to use DRM, and is defined thus:

```
\def\drmdecinitfont{\unexpanded{\font\drminitfontcom=drm10}}%
```

In other words, one must define the fonts in the old-fashioned TEX way. The above is the default; so whenever you've changed it for some reason, you `\drmdecinitfontdefault` can get it back to the above by entering `\drmdecinitfontdefault`; this simply restores the default definition as given above.

For the ornate enumerations, we can simply redefine `\drmdecinitfont` to use the lining figures from `drmsym`, which conveniently are located at precisely the code points that one would expect them. Simply issue:

```
\def\drmdecinitfont{\unexpanded{\font\drminitfontcom=drmsym10} %}%
\renewcommand{\labelenumi}{%
 \drmdecinit{14pt}{14pt}{blue}{red}{\theenumi}}
```

This redefines `\drmdecinitfont` to use `drmsym` rather than simply `drm`, then redefines the enumerate labels to be DRM decorative initials, resulting in the following:


The first item.


The second item.

Any character whatever can be used this way, provided that the font is correctly selected. Doubtlessly many creative uses for this ability will be found.

One trick, almost necessary when using these on any significant scale, is a macro to make them less typing. For example, to use them as four-line lettrines using Daniel Flipo's excellent `lettrine` package:

```
\def\declettrine#1#2{%
 \lettrine[lines=4,nindent=0pt,findent=-1em]%
{\drmdecinit{40pt}{40pt}{blue}{red}{#1}}{#2}%
}%
```

Rather than having to type the whole of the above each time now, one can do it in a more natural manner:

```
\declettrine{L}{orem ipsum}
dolor sit amet, consectetur
adipiscing elit. Ut porttitor
libero lacus, a rhoncus dolor
finibus vel. Morbi volputate
condimentum ornare. In
scelerisque aliquam\drmfelip
```

OREM IPSUM dolor sit amet, consectetur adipiscing elit. Ut porttitor libero lacus, a rhoncus dolor finibus vel. Morbi volputate condimentum ornare. In scelerisque aliquam. . . .

4.7 Math

Your author is far from a mathematician, so he's not really able to judge the quality of the following; but DRM does offer matching math fonts. These are limited to the default TEX math fonts, however; AMS extensions and the like are not available. Perhaps one day (after finishing the ornaments and decorative initials) they will be, but for now one will have to pull in other fonts for anything that goes beyond plain TEX. Using them in bold goes a long way to making them match the rest of DRM.

First, we have a full set of mathematical Greek letters. As seems to be the custom, the capitals are upright and the lowercase slanted. These can all be accessed via the customary TEX math character names.

Greek Letters

A	`A`	α	`$\alpha$`	B	`B`	β	`$\beta$`
Γ	`$\Gamma$`	γ	`$\gamma$`	Δ	`$\Delta$`	δ	`$\delta$`
E	`E`	ϵ	`$\epsilon$`	Z	`Z`	ζ	`$\zeta$`
H	`H`	η	`$\eta$`	Θ	`$\Theta$`	θ	`$\theta$`
I	`I`	ι	`$\iota$`	K	`K`	κ	`$\kappa$`
Λ	`$\Lambda$`	λ	`$\lambda$`	M	`M`	μ	`$\mu$`
N	`N`	ν	`$\nu$`	Ξ	`$\Xi$`	ξ	`$\xi$`
O	`O`	o	`o`	Π	`$\Pi$`	π	`$\pi$`
P	`P`	ρ	`$\rho$`	Σ	`$\Sigma$`	σ	`$\sigma$`
T	`T`	τ	`$\tau$`	Y	`Y`	υ	`$\upsilon$`
Φ	`$\Phi$`	ϕ	`$\phi$`	X	`X`	χ	`$\chi$`
Ψ	`$\Psi$`	ψ	`$\psi$`	Ω	`$\Omega$`	ω	`$\omega$`
ϑ	`$\vartheta$`	ϖ	`$\varpi$`	ς	`$\varsigma$`	φ	`$\varphi$`

This alphabet led directly to DRM's Greek font, which we discuss elsewhere.[4]

[4] *See supra*, Section 4.8, at 28.

Math Calligraphic							
$\mathcal{A}$	$\mathcal{B}$	$\mathcal{C}$	$\mathcal{D}$	$\mathcal{E}$	$\mathcal{F}$	$\mathcal{G}$	$\mathcal{H}$
$\mathcal{I}$	$\mathcal{J}$	$\mathcal{K}$	$\mathcal{L}$	$\mathcal{M}$	$\mathcal{N}$	$\mathcal{O}$	$\mathcal{P}$
$\mathcal{Q}$	$\mathcal{R}$	$\mathcal{S}$	$\mathcal{T}$	$\mathcal{U}$	$\mathcal{V}$	$\mathcal{W}$	$\mathcal{X}$
$\mathcal{Y}$	$\mathcal{Z}$						

DRM also has its own extensible characters and variable-sized math characters; a few examples in various sizes are below.

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\prod_{i=1}^{n} i^2 = \left(\frac{n(n+1)(2n+1)}{6} \right)$$

$$\sum_{P_i \in Paths(I)} Probes(P_i)$$

$$\overbrace{abcdefghijklmnop}$$

By default, using `\big` and friends doesn't work, a problem I haven't been able to resolve. However, by requiring `amsmath`, `drm` provides a more directly flexible mechanism for this: `\bigd`, which allows arbitrarily sized delimiters. It takes a single argument, which is an integer describing the desired size:

`\bigd`

$$\bigd{2}\{ \quad \bigd{4}\{ \quad \bigd{8}\{ \quad \bigd{16}\{$$

`\left` and `\right` work as expected with DRM's delimiters.

This symbols, of course, also work inline (as opposed to displayed, which is what we have above); you can take $\sqrt{\frac{2}{3}}$ and have $(3 \times \left(\frac{4}{3} \right))$ just as easily in a paragraph as in a display, though you may want to take care that you're not using too much space for your lines. (I didn't take care in this paragraph, and you can see how bad it looks.)

4.8 Greek

Because DRM offers Greek characters in math, it was a short step to offer actual Greek text, and so I've done so, according to the standard LGR encoding. I can just barely read the Greek alphabet and remember very little of the grammar, and what little I once knew was all ancient and koine, but here it is. DRM's Greek support is limited; while it offers all the normal *polutoníko* accents, subscripts, and breathings, along with some archaic characters like the digamma, there is no italic, small caps, or various weights. DRM isn't, therefore, really suitable for typesetting whole Greek works; it will, however, offer attractive typesetting of Greek phrases and quotations within a text set otherwise in the Latin alphabet.

`\grktext` The `\grktext` command changes the current font encoding to LGR, which for DRM's purposes means it's typesetting with Greek characters from then on. The macro `\textgrk` is similar, but takes a single argument, which is typeset in Greek characters. A few examples follow.

`\textgrk`

```
We know that                    {\grktext >En {>a}rq\char'254\
\textgrk{Aqilleuc} was one of   {>~h}n {<o} l{\'o}goc, ka{\'i}
the Greeks' greatest warriors.  {<o} l{\'o}goc {\~>h}n
                                pr{\'o}c t{\'o}n je{\'o}n,
                                ka{\'i} je{\'o}c {>\~h}n {<o}
                                l{\'o}goc.}
```

We know that Αχιλλευς was one of the Greeks' greatest warriors.

'Εν 'αρχῇ ' ην 'ο λόγος, καί 'ο λόγος ἦν πρός τόν θεόν, καί θεός 'ῆν 'ο λόγος.

I understand that `babel` has facilities for making the typesetting of all the πολυτονικο accents much cleaner, but I don't write enough in Greek to have learned to use it, resulting in the mess you see above. Note that `drm` (the package) does *not* pull in `babel` or the `polutoniko` option, or any other Greek typesetting package; it simply provides the fonts. If you're typesetting long enough passages that you need Greek hyphenation and the like, you'll have to invoke the appropriate package yourself.

5 Implementation

Load the required packages. DRM contains TS1, LGR, OML, OMS, and T1 encoded fonts, so we load `fontenc` with all these encodings as options. We also load `modroman` for the `\romanize` macro, defined below. Finally, we load `gmp` for the decorative initials (this allows including METAPOST code in LaTeX source).

```
1 \RequirePackage[LGR,OML,OMS,TS1,T1]{fontenc}
2 \RequirePackage{modroman}
```

```
\RequirePackage{amsmath}
\RequirePackage{gmp}
```

Now we declare our options.

```
\newif\ifnodefault\nodefaultfalse
\newif\ifnodefaultmath\nodefaultmathfalse
\newif\ifnodefaulttext\nodefaulttextfalse
\newif\ifsymbolsonly\symbolsonlyfalse
\newif\iftypeone\typeonefalse
\DeclareOption{nodefault}{\nodefaulttrue\nodefaultmathtrue%
\nodefaulttexttrue}
\DeclareOption{nodefaultmath}{\nodefaultmathtrue}
\DeclareOption{nodefaulttext}{\nodefaulttexttrue}
\DeclareOption{symbolsonly}{\symbolsonlytrue\nodefaulttrue%
\nodefaulttexttrue\nodefaultmathtrue}
\DeclareOption{typeone}{\typeonetrue}
\ProcessOptions
```

Begin defining the font families. First, define the fonts with the file `drm.map` if the option `typeone` was requested; otherwise, load the METAFONT files directly.

```
\iftypeone
\RequirePackage{ifpdf}
\ifpdf
\pdfmapfile{=drm.map}
\fi
\DeclareFontFamily{T1}{drm}{}
\DeclareFontFamily{TS1}{drm}{}
\DeclareFontFamily{LGR}{drm}{}
\DeclareFontFamily{U}{drmsups}{}
\DeclareFontFamily{U}{drminfs}{}
\DeclareFontFamily{T1}{drmdoz}{}
\DeclareFontShape{T1}{drmdoz}{m}{n}{ <-7> drmdoz6
<7> drmdoz7 <8> drmdoz8 <9> drmdoz9 <10-12> drmdoz10
<12-13> drmdoz12 <14-17> drmdoz14 <17-24> drmdoz17
<24-> drmdoz24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{sl}{ <-7> drmdozsl6
<7> drmdozsl7 <8> drmdozsl8 <9> drmdozsl9 <10-12> drmdozsl10
<12-13> drmdozsl12 <14-17> drmdozsl14 <17-24> drmdozsl17
<24-> drmdozsl24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{it}{ <-7> drmdozit6
<7> drmdozit7 <8> drmdozit8 <9> drmdozit9 <10-12> drmdozit10
<12-13> drmdozit12 <14-17> drmdozit14 <17-24> drmdozit17
<24-> drmdozit24 }{}
\DeclareFontShape{T1}{drmdoz}{l}{n}{ <-7> drmdozl6
<7> drmdozl7 <8> drmdozl8 <9> drmdozl9 <10-12> drmdozl10
<12-13> drmdozl12 <14-17> drmdozl14 <17-24> drmdozl17
<24-> drmdozl24 }{}
\DeclareFontShape{T1}{drmdoz}{b}{n}{ <-7> drmdozb6
<7> drmdozb7 <8> drmdozb8 <9> drmdozb9 <10-12> drmdozb10
```

```
<12-13> drmdozb12 <14-17> drmdozb14 <17-24> drmdozb17
<24-> drmdozb24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{n}{ <-7> drmdozbx6
<7> drmdozbx7 <8> drmdozbx8 <9> drmdozbx9 <10-12> drmdozbx10
<12-13> drmdozbx12 <14-17> drmdozbx14 <17-24> drmdozbx17
<24-> drmdozbx24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{it}{ <-7> drmdozitbx6
<7> drmdozitbx7 <8> drmdozitbx8 <9> drmdozitbx9 <10-12> drmdozitbx10
<12-13> drmdozitbx12 <14-17> drmdozitbx14 <17-24> drmdozitbx17
<24-> drmdozitbx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{ui}{ <-7> drmdozui6
<7> drmdozui7 <8> drmdozui8 <9> drmdozui9 <10-12> drmdozui10
<12-13> drmdozui12 <14-17> drmdozui14 <17-24> drmdozui17
<24-> drmdozui24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{ui}{ <-7> drmdozuibx6
<7> drmdozuibx7 <8> drmdozuibx8 <9> drmdozuibx9 <10-12> drmdozuibx10
<12-13> drmdozuibx12 <14-17> drmdozuibx14 <17-24> drmdozuibx17
<24-> drmdozuibx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{sc}{ <-7> drmdozsc6
<7> drmdozsc7 <8> drmdozsc8 <9> drmdozsc9 <10-12> drmdozsc10
<12-13> drmdozsc12 <14-17> drmdozsc14 <17-24> drmdozsc17
<24-> drmdozsc24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{tc}{ <-7> drmdoztc6
<7> drmdoztc7 <8> drmdoztc8 <9> drmdoztc9 <10-12> drmdoztc10
<12-13> drmdoztc12 <14-17> drmdoztc14 <17-24> drmdoztc17
<24-> drmdoztc24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{sc}{ <-7> drmdozscbx6
<7> drmdozscbx7 <8> drmdozscbx8 <9> drmdozscbx9 <10-12> drmdozscbx10
<12-13> drmdozscbx12 <14-17> drmdozscbx14 <17-24> drmdozscbx17
<24-> drmdozscbx24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{tc}{ <-7> drmdoztcbx6
<7> drmdoztcbx7 <8> drmdoztcbx8 <9> drmdoztcbx9 <10-12> drmdoztcbx10
<12-13> drmdoztcbx12 <14-17> drmdoztcbx14 <17-24> drmdoztcbx17
<24-> drmdoztcbx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{itsc}{<-7> drmdozitsc6 <7>
drmdozitsc7 <8> drmdozitsc8 <9> drmdozitsc9 <10-12>
drmdozitsc10 <12-14> drmdozitsc12 <14-17> drmdozitsc14
<17-24> drmdozitsc17 <24-> drmdozitsc24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{ittc}{<-7> drmdozittc6 <7>
drmdozittc7 <8> drmdozittc8 <9> drmdozittc9 <10-12>
drmdozittc10 <12-14> drmdozittc12 <14-17> drmdozittc14
<17-24> drmdozittc17 <24-> drmdozittc24 }{}
\DeclareFontShape{U}{drminfs}{m}{n}{ <-7> drminf6
<7> drminf7 <8> drminf8 <9> drminf9 <10-12> drminf10
<12-13> drminf12 <14-17> drminf14 <17-24> drminf17
<24-> drminf24 }{}
\DeclareFontShape{U}{drmsups}{m}{n}{ <-7> drmfigs6
<7> drmfigs7 <8> drmfigs8 <9> drmfigs9 <10-12> drmfigs10
<12-13> drmfigs12 <14-17> drmfigs14 <17-24> drmfigs17
<24-> drmfigs24 }{}
```

```
\DeclareFontShape{T1}{drm}{m}{n}{ <-7> drm6 <7> drm7 <8> drm8
<9> drm9 <10-12> drm10 <12-13> drm12 <14-17> drm14
<17-24> drm17 <24-> drm24 }{}
\DeclareFontShape{T1}{drm}{m}{sc}{<-7> drmsc6 <7> drmsc7
<8> drmsc8 <9> drmsc9 <10-12> drmsc10 <12-14> drmsc12
<14-17> drmsc14 <17-24> drmsc17 <24-> drmsc24 }{}
\DeclareFontShape{T1}{drm}{m}{tc}{<-7> drmtc6 <7> drmtc7
<8> drmtc8 <9> drmtc9 <10-12> drmtc10 <12-14> drmtc12
<14-17> drmtc14 <17-24> drmtc17 <24-> drmtc24 }{}
\DeclareFontShape{T1}{drm}{bx}{sc}{<-7> drmscbx6 <7> drmscbx7
<8> drmscbx8 <9> drmscbx9 <10-12> drmscbx10 <12-14> drmscbx12
<14-17> drmscbx14 <17-24> drmscbx17 <24-> drmscbx24 }{}
\DeclareFontShape{T1}{drm}{bx}{tc}{<-7> drmtcbx6 <7> drmtcbx7
<8> drmtcbx8 <9> drmtcbx9 <10-12> drmtcbx10 <12-14> drmtcbx12
<14-17> drmtcbx14 <17-24> drmtcbx17 <24-> drmtcbx24 }{}
\DeclareFontShape{T1}{drm}{m}{itsc}{<-7> drmitsc6 <7> drmitsc7
<8> drmitsc8 <9> drmitsc9 <10-12> drmitsc10 <12-14> drmitsc12
<14-17> drmitsc14 <17-24> drmitsc17 <24-> drmitsc24 }{}
\DeclareFontShape{T1}{drm}{m}{ittc}{<-7> drmittc6 <7> drmittc7
<8> drmittc8 <9> drmittc9 <10-12> drmittc10 <12-14> drmittc12
<14-17> drmittc14 <17-24> drmittc17 <24-> drmittc24 }{}
\DeclareFontShape{T1}{drm}{m}{sl}{<-7> drmsl6 <7> drmsl7
<8> drmsl8 <9> drmsl9 <10-12> drmsl10 <12-14> drmsl12
<14-17> drmsl14 <17-24> drmsl17 <24-> drmsl24 }{}
\DeclareFontShape{T1}{drm}{m}{it}{ <-7> drmit6 <7> drmit7
<8> drmit8 <9> drmit9 <10-12> drmit10 <12-14> drmit12
<14-17> drmit14 <17-24> drmit17 <24-> drmit24 }{}
\DeclareFontShape{T1}{drm}{bx}{it}{ <-7> drmitbx6 <7> drmitbx7
<8> drmitbx8 <9> drmitbx9 <10-12> drmitbx10 <12-14> drmitbx12
<14-17> drmitbx14 <17-24> drmitbx17 <24-> drmitbx24 }{}
\DeclareFontShape{T1}{drm}{m}{ui}{<-7> drmui6 <7> drmui7
<8> drmui8 <9> drmui9 <10-12> drmui10 <12-14> drmui12
<14-17> drmui14 <17-24> drmui17 <24-> drmui24 }{}
\DeclareFontShape{T1}{drm}{bx}{ui}{<-7> drmuibx6 <7> drmuibx7
<8> drmuibx8 <9> drmuibx9 <10-12> drmuibx10 <12-14> drmuibx12
<14-17> drmuibx14 <17-24> drmuibx17 <24-> drmuibx24 }{}
\DeclareFontShape{T1}{drm}{l}{n}{<-7> drml6 <7> drml7
<8> drml8 <9> drml9 <10-12> drml10 <12-14> drml12
<14-17> drml14 <17-24> drml17 <24-> drml24 }{}
\DeclareFontShape{T1}{drm}{b}{n}{<-7> drmb6 <7> drmb7
<8> drmb8 <9> drmb9 <10-12> drmb10 <12-14> drmb12
<14-17> drmb14 <17-24> drmb17 <24-> drmb24 }{}
\DeclareFontShape{T1}{drm}{bx}{n}{<-7> drmbx6 <7> drmbx7
<8> drmbx8 <9> drmbx9 <10-12> drmbx10 <12-14> drmbx12
<14-17> drmbx14 <17-24> drmbx17 <24-> drmbx24 }{}
\DeclareFontShape{TS1}{drm}{m}{n}{<-7> drmsym7
<8> drmsym8 <9> drmsym9 <10-12> drmsym10 <12-14> drmsym12
<14-17> drmsym14 <17-24> drmsym17 <24-> drmsym24 }{}
\DeclareFontShape{LGR}{drm}{m}{n}{<-> drmgrk10 }{}
\DeclareSymbolFont{dozens}{T1}{drmdoz}{m}{n}
```

```
\else
\DeclareFontFamily{T1}{drm}{}
\DeclareFontFamily{TS1}{drm}{}
\DeclareFontFamily{LGR}{drm}{}
\DeclareFontFamily{U}{drmsups}{}
\DeclareFontFamily{U}{drminfs}{}
\DeclareFontFamily{T1}{drmdoz}{}
\DeclareFontShape{T1}{drmdoz}{m}{n}{ <-7> drmdoz6
<7> drmdoz7 <8> drmdoz8 <9> drmdoz9 <10-12> drmdoz10
<12-13> drmdoz12 <14-17> drmdoz14 <17-24> drmdoz17
<24-> drmdoz24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{sl}{ <-7> drmdozsl6
<7> drmdozsl7 <8> drmdozsl8 <9> drmdozsl9 <10-12> drmdozsl10
<12-13> drmdozsl12 <14-17> drmdozsl14 <17-24> drmdozsl17
<24-> drmdozsl24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{it}{ <-7> drmdozit6
<7> drmdozit7 <8> drmdozit8 <9> drmdozit9 <10-12> drmdozit10
<12-13> drmdozit12 <14-17> drmdozit14 <17-24> drmdozit17
<24-> drmdozit24 }{}
\DeclareFontShape{T1}{drmdoz}{l}{n}{ <-7> drmdozl6
<7> drmdozl7 <8> drmdozl8 <9> drmdozl9 <10-12> drmdozl10
<12-13> drmdozl12 <14-17> drmdozl14 <17-24> drmdozl17
<24-> drmdozl24 }{}
\DeclareFontShape{T1}{drmdoz}{b}{n}{ <-7> drmdozb6
<7> drmdozb7 <8> drmdozb8 <9> drmdozb9 <10-12> drmdozb10
<12-13> drmdozb12 <14-17> drmdozb14 <17-24> drmdozb17
<24-> drmdozb24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{n}{ <-7> drmdozbx6
<7> drmdozbx7 <8> drmdozbx8 <9> drmdozbx9 <10-12> drmdozbx10
<12-13> drmdozbx12 <14-17> drmdozbx14 <17-24> drmdozbx17
<24-> drmdozbx24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{it}{ <-7> drmdozitbx6
<7> drmdozitbx7 <8> drmdozitbx8 <9> drmdozitbx9 <10-12> drmdozitbx10
<12-13> drmdozitbx12 <14-17> drmdozitbx14 <17-24> drmdozitbx17
<24-> drmdozitbx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{ui}{ <-7> drmdozui6
<7> drmdozui7 <8> drmdozui8 <9> drmdozui9 <10-12> drmdozui10
<12-13> drmdozui12 <14-17> drmdozui14 <17-24> drmdozui17
<24-> drmdozui24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{ui}{ <-7> drmdozuibx6
<7> drmdozuibx7 <8> drmdozuibx8 <9> drmdozuibx9 <10-12> drmdozuibx10
<12-13> drmdozuibx12 <14-17> drmdozuibx14 <17-24> drmdozuibx17
<24-> drmdozuibx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{sc}{ <-7> drmdozsc6
<7> drmdozsc7 <8> drmdozsc8 <9> drmdozsc9 <10-12> drmdozsc10
<12-13> drmdozsc12 <14-17> drmdozsc14 <17-24> drmdozsc17
<24-> drmdozsc24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{tc}{ <-7> drmdoztc6
<7> drmdoztc7 <8> drmdoztc8 <9> drmdoztc9 <10-12> drmdoztc10
<12-13> drmdoztc12 <14-17> drmdoztc14 <17-24> drmdoztc17
```

```
<24-> drmdoztc24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{sc}{ <-7> drmdozscbx6
<7> drmdozscbx7 <8> drmdozscbx8 <9> drmdozscbx9 <10-12> drmdozscbx10
<12-13> drmdozscbx12 <14-17> drmdozscbx14 <17-24> drmdozscbx17
<24-> drmdozscbx24 }{}
\DeclareFontShape{T1}{drmdoz}{bx}{tc}{ <-7> drmdoztcbx6
<7> drmdoztcbx7 <8> drmdoztcbx8 <9> drmdoztcbx9 <10-12> drmdoztcbx10
<12-13> drmdoztcbx12 <14-17> drmdoztcbx14 <17-24> drmdoztcbx17
<24-> drmdoztcbx24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{itsc}{<-7> drmdozitsc6 <7>
drmdozitsc7 <8> drmdozitsc8 <9> drmdozitsc9 <10-12>
drmdozitsc10 <12-14> drmdozitsc12 <14-17> drmdozitsc14
<17-24> drmdozitsc17 <24-> drmdozitsc24 }{}
\DeclareFontShape{T1}{drmdoz}{m}{ittc}{<-7> drmdozittc6 <7>
drmdozittc7 <8> drmdozittc8 <9> drmdozittc9 <10-12>
drmdozittc10 <12-14> drmdozittc12 <14-17> drmdozittc14
<17-24> drmdozittc17 <24-> drmdozittc24 }{}
\DeclareFontShape{U}{drminfs}{m}{n}{ <-7> drminf6
<7> drminf7 <8> drminf8 <9> drminf9 <10-12> drminf10
<12-13> drminf12 <14-17> drminf14 <17-24> drminf17
<24-> drminf24 }{}
\DeclareFontShape{U}{drmsups}{m}{n}{ <-7> drmfigs6
<7> drmfigs7 <8> drmfigs8 <9> drmfigs9 <10-12> drmfigs10
<12-13> drmfigs12 <14-17> drmfigs14 <17-24> drmfigs17
<24-> drmfigs24 }{}
\DeclareFontShape{T1}{drm}{m}{n}{ <-7> drm6 <7> drm7 <8> drm8
<9> drm9 <10-12> drm10 <12-13> drm12 <14-17> drm14
<17-24> drm17 <24-> drm24 }{}
\DeclareFontShape{T1}{drm}{m}{sc}{<-7> drmsc6 <7> drmsc7
<8> drmsc8 <9> drmsc9 <10-12> drmsc10 <12-14> drmsc12
<14-17> drmsc14 <17-24> drmsc17 <24-> drmsc24 }{}
\DeclareFontShape{T1}{drm}{m}{tc}{<-7> drmtc6 <7> drmtc7
<8> drmtc8 <9> drmtc9 <10-12> drmtc10 <12-14> drmtc12
<14-17> drmtc14 <17-24> drmtc17 <24-> drmtc24 }{}
\DeclareFontShape{T1}{drm}{bx}{sc}{<-7> drmscbx6 <7> drmscbx7
<8> drmscbx8 <9> drmscbx9 <10-12> drmscbx10 <12-14> drmscbx12
<14-17> drmscbx14 <17-24> drmscbx17 <24-> drmscbx24 }{}
\DeclareFontShape{T1}{drm}{bx}{tc}{<-7> drmtcbx6 <7> drmtcbx7
<8> drmtcbx8 <9> drmtcbx9 <10-12> drmtcbx10 <12-14> drmtcbx12
<14-17> drmtcbx14 <17-24> drmtcbx17 <24-> drmtcbx24 }{}
\DeclareFontShape{T1}{drm}{m}{itsc}{<-7> drmitsc6 <7> drmitsc7
<8> drmitsc8 <9> drmitsc9 <10-12> drmitsc10 <12-14> drmitsc12
<14-17> drmitsc14 <17-24> drmitsc17 <24-> drmitsc24 }{}
\DeclareFontShape{T1}{drm}{m}{ittc}{<-7> drmittc6 <7> drmittc7
<8> drmittc8 <9> drmittc9 <10-12> drmittc10 <12-14> drmittc12
<14-17> drmittc14 <17-24> drmittc17 <24-> drmittc24 }{}
\DeclareFontShape{T1}{drm}{m}{sl}{<-7> drmsl6 <7> drmsl7
<8> drmsl8 <9> drmsl9 <10-12> drmsl10 <12-14> drmsl12
<14-17> drmsl14 <17-24> drmsl17 <24-> drmsl24 }{}
\DeclareFontShape{T1}{drm}{m}{it}{ <-7> drmit6 <7> drmit7
```

```
<8> drmit8 <9> drmit9 <10-12> drmit10 <12-14> drmit12
<14-17> drmit14 <17-24> drmit17 <24-> drmit24 }{}
\DeclareFontShape{T1}{drm}{bx}{it}{ <-7> drmitbx6 <7> drmitbx7
<8> drmitbx8 <9> drmitbx9 <10-12> drmitbx10 <12-14> drmitbx12
<14-17> drmitbx14 <17-24> drmitbx17 <24-> drmitbx24 }{}
\DeclareFontShape{T1}{drm}{m}{ui}{<-7> drmui6 <7> drmui7
<8> drmui8 <9> drmui9 <10-12> drmui10 <12-14> drmui12
<14-17> drmui14 <17-24> drmui17 <24-> drmui24 }{}
\DeclareFontShape{T1}{drm}{bx}{ui}{<-7> drmuibx6 <7> drmuibx7
<8> drmuibx8 <9> drmuibx9 <10-12> drmuibx10 <12-14> drmuibx12
<14-17> drmuibx14 <17-24> drmuibx17 <24-> drmuibx24 }{}
\DeclareFontShape{T1}{drm}{l}{n}{<-7> drml6 <7> drml7
<8> drml8 <9> drml9 <10-12> drml10 <12-14> drml12
<14-17> drml14 <17-24> drml17 <24-> drml24 }{}
\DeclareFontShape{T1}{drm}{b}{n}{<-7> drmb6 <7> drmb7
<8> drmb8 <9> drmb9 <10-12> drmb10 <12-14> drmb12
<14-17> drmb14 <17-24> drmb17 <24-> drmb24 }{}
\DeclareFontShape{T1}{drm}{bx}{n}{<-7> drmbx6 <7> drmbx7
<8> drmbx8 <9> drmbx9 <10-12> drmbx10 <12-14> drmbx12
<14-17> drmbx14 <17-24> drmbx17 <24-> drmbx24 }{}
\DeclareFontShape{TS1}{drm}{m}{n}{<-7> drmsym7
<8> drmsym8 <9> drmsym9 <10-12> drmsym10 <12-14> drmsym12
<14-17> drmsym14 <17-24> drmsym17 <24-> drmsym24 }{}
\DeclareFontShape{LGR}{drm}{m}{n}{<-> drmgrk10 }{}
\DeclareSymbolFont{dozens}{T1}{drmdoz}{m}{n}
\fi
```

Next, we define the appropriate dozenal characters using the DRM dozenal fonts if and only if the `dozenal` package is loaded. If you want to override this behavior and use the default, Computer Modern-ish `dozenal` fonts, load `dozenal` *after* `drm`.

```
\@ifpackageloaded{dozenal}{%
\def\doz#1{{\fontfamily{drmdoz}\fontencoding{T1}\selectfont #1}}%
\def\drmsupfont{drmsups}\def\drminffont{drminfs}%
\renewcommand\x{%
\ifx\f@family\drmsupfont X%
\else\ifx\f@family\drminffont X%
\else\TextOrMath{\protect\doz{{X}}}{\doz@X}\fi\fi}%
\renewcommand\e{%
\ifx\f@family\drmsupfont E%
\else\ifx\f@family\drminffont E%
\else\TextOrMath{\protect\doz{{E}}}{\doz@E}\fi\fi}%
\DeclareMathSymbol{\doz@X}{\mathord}{dozens}{88}%
\DeclareMathSymbol{\doz@E}{\mathord}{dozens}{69}%
}{}
```

Now, set the default text font as DRM unless `nodefault` or `nodefaulttext` has been specified. Also redefine the default footnote counters to use superior figures rather than automatically scaled figures.

```
\ifnodefault\else\ifnodefaulttext\else
```

```
\renewcommand\encodingdefault{T1}
\renewcommand\familydefault{drm}
\def\f@@n@te{footnote}
\def\@makefnmark{%
\hbox{\drmsupfigs\@thefnmark}
}%
\fi\fi
```

Give ourselves a shortcut to access the short-tailed letter "q," just in case we need it.

```
\def\drmshortq{{\usefont{T1}{drm}{m}{n}\char'137}}
```

That gives us the satisfying "Q" rather than the "Q" we would otherwise get. Useful for circumstance when the "Q" is followed by characters which hang below the baseline, or in a dropped initial.

Now we move on to define commands for the more unusual shapes, since LaTeX 2_ε doesn't have them built in. We start with TITLING SMALL CAPS, then move on to upright italics. We also define `\textgrk` and `\grktext`, for typesetting in Greek characters. Finally, we also define the commands to produce the superior and inferior figures.

```
\def\tcshape{\fontshape{tc}\selectfont}
\def\texttc#1{{\tcshape#1}}
\def\ittcshape{\fontshape{ittc}\selectfont}
\def\textittc#1{{\ittcshape#1}}
\def\itscshape{\fontshape{itsc}\selectfont}
\def\textitsc#1{{\itscshape#1}}
\def\uishape{\fontshape{ui}\selectfont}
\def\textui#1{{\uishape#1}}
\def\grktext{\fontencoding{LGR}\selectfont}
\def\textgrk#1{{\grktext#1}}
\def\drmsupfigs{\usefont{U}{drmsups}{m}{n}}
\def\textdrmsupfigs#1{{\drmsupfigs#1}}
\def\drminffigs{\usefont{U}{drminfs}{m}{n}}
\def\textdrminffigs#1{{\drminffigs#1}}
```

Next, we define the weights. We know that `\textbf` will give us normal boldface, and that `\textmd` will return us to medium weight; but since DRM also has a light weight and a bold non-extended, we need to define commands for those, as well.

```
\def\lseries{\fontseries{l}\selectfont}
\def\textl#1{{\lseries#1}}
\def\bseries{\fontseries{b}\selectfont}
\def\textb#1{{\bseries#1}}
```

Moving on, we define size commands based on traditional English-language printers' names. Why? *Because we can*, that's why.

```
\def\loosen{\addtolength{\baselineskip}{1pt}}
\def\excelsior{\fontsize{3pt}{3.5pt}\selectfont}
\def\minikin{\excelsior}
\def\brilliant{\fontsize{4pt}{4.5pt}\selectfont}
```

```
\def\diamondsize{\fontsize{4.5pt}{5pt}\selectfont}
\def\pearl{\fontsize{5pt}{6pt}\selectfont}
\def\agate{\fontsize{5.5pt}{6.5pt}\selectfont}
\def\ruby{\agate}
\def\nonpareille{\fontsize{6pt}{7pt}\selectfont}
\def\minionette{\fontsize{6.5pt}{7.5pt}\selectfont}
\def\emerald{\minionette}
\def\minion{\fontsize{7pt}{8pt}\selectfont}
\def\brevier{\fontsize{8pt}{9pt}\selectfont}
\def\petit{\brevier}
\def\smalltext{\brevier}
\def\bourgeois{\fontsize{9pt}{10pt}\selectfont}
\def\galliard{\bourgeois}
\def\longprimer{\fontsize{10pt}{12pt}\selectfont}
\def\corpus{\longprimer}
\def\garamond{\longprimer}
\def\smallpica{\fontsize{11pt}{13pt}\selectfont}
\def\philosophy{\smallpica}
\def\pica{\fontsize{12pt}{15pt}\selectfont}
\def\english{\fontsize{14pt}{17pt}\selectfont}
\def\mittel{\english}
\def\augustin{\english}
\def\columbian{\fontsize{16pt}{19pt}\selectfont}
\def\twolinebrevier{\columbian}
\def\greatprimer{\fontsize{18pt}{22pt}\selectfont}
\def\paragon{\fontsize{20pt}{24pt}\selectfont}
\def\doublesmallpica{\fontsize{21pt}{25pt}\selectfont}
\def\doublesmallpicaus{\fontsize{22pt}{26pt}\selectfont}
\def\doublepicabrit{\doublesmallpicaus}
\def\doublepica{\fontsize{24pt}{28pt}\selectfont}
\def\twolinepica{\doublepica}
\def\doubleenglish{\fontsize{28pt}{33pt}\selectfont}
\def\twolineenglish{\doubleenglish}
\def\fivelinenonpareil{\fontsize{30pt}{35pt}\selectfont}
\def\fourlinebrevier{\fontsize{32pt}{38pt}\selectfont}
\def\doublegreatprimer{\fontsize{36pt}{42pt}\selectfont}
\def\twolinegreatprimer{\doublegreatprimer}
\def\meridian{\fontsize{44pt}{50pt}\selectfont}
\def\twolinedoublepica{\meridian}
\def\trafalgar{\meridian}
\def\canon{\fontsize{48pt}{54pt}\selectfont}
\def\fourline{\canon}
\def\fivelinepica{\fontsize{60pt}{66pt}\selectfont}
\def\inch{\fontsize{72pt}{78pt}\selectfont}
```

Now we move on to define the math fonts. This turned out to be a surprisingly convoluted process, and I only marginally understand what's going on here; but it works, and I'll try to go through it as best as I can.

First, we make LaTeX aware of our math fonts:

```
\DeclareFontFamily{OML}{drm}{}
```

```
\DeclareFontShape{OML}{drm}{m}{n}{ <-> drmmi10 }{}
\DeclareFontFamily{OMS}{drm}{}
\DeclareFontShape{OMS}{drm}{m}{n}{ <-> drmsy10 }{}
\DeclareFontFamily{OMX}{drm}{}
\DeclareFontShape{OMX}{drm}{m}{n}{ <-> drmomx10 }{}
```

Next, we declare something called a math *version*; this way we can define new math shapes without clobbering the default settings. I'm not sure why this is really necessary, but it appears to be; so we define a math version `drmmath`:

```
\DeclareMathVersion{drmmath}
```

Now we define our symbol fonts. This lets LaTeX know where to yank its symbols from when typesetting a math formula.

```
\ifnodefault\else\ifnodefaultmath\else
\SetSymbolFont{operators}{drmmath}{T1}{drm}{m}{n}
\SetSymbolFont{letters}{drmmath}{OML}{drm}{m}{n}
\DeclareSymbolFont{symbs}{TS1}{drm}{m}{n}
\SetSymbolFont{symbs}{drmmath}{TS1}{drm}{m}{n}
\DeclareSymbolFont{drmmathsy}{OMS}{drm}{m}{n}
\SetSymbolFont{drmmathsy}{drmmath}{OMS}{drm}{m}{n}
\DeclareSymbolFont{drmmathomx}{OMX}{drm}{m}{n}
\SetSymbolFont{drmmathomx}{drmmath}{OMX}{drm}{m}{n}
\fi\fi
```

Now we write in our *math alphabets*, so that when we request `\mathcal` or something similar we'll get DRM and not Computer Modern. We define `\mathcal`, of course, and also `\drmmathlets`, just in case we want to request DRM directly.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathAlphabet{\drmmathlets}{OML}{drm}{m}{n}
\SetMathAlphabet{\drmmathlets}{drmmath}{OML}{drm}{m}{n}
\DeclareMathAlphabet{\mathcal}{OMS}{drm}{m}{n}
\SetMathAlphabet{\mathcal}{drmmath}{OMS}{drm}{m}{n}
\fi\fi
```

Now ensure that we get lining figures in math mode.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{0}{0}{symbs}{48}
\DeclareMathSymbol{1}{0}{symbs}{49}
\DeclareMathSymbol{2}{0}{symbs}{50}
\DeclareMathSymbol{3}{0}{symbs}{51}
\DeclareMathSymbol{4}{0}{symbs}{52}
\DeclareMathSymbol{5}{0}{symbs}{53}
\DeclareMathSymbol{6}{0}{symbs}{54}
\DeclareMathSymbol{7}{0}{symbs}{55}
\DeclareMathSymbol{8}{0}{symbs}{56}
\DeclareMathSymbol{9}{0}{symbs}{57}
\fi\fi
```

Now, it appears to be necessary to redefine all the math symbols, so we do that. Beginning with the Greek letters:

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{\Gamma}{0}{letters}{0}
\DeclareMathSymbol{\Delta}{0}{letters}{1}
\DeclareMathSymbol{\Theta}{0}{letters}{2}
\DeclareMathSymbol{\Lambda}{0}{letters}{3}
\DeclareMathSymbol{\Xi}{0}{letters}{4}
\DeclareMathSymbol{\Pi}{0}{letters}{5}
\DeclareMathSymbol{\Sigma}{0}{letters}{6}
\DeclareMathSymbol{\Upsilon}{0}{letters}{7}
\DeclareMathSymbol{\Phi}{0}{letters}{8}
\DeclareMathSymbol{\Psi}{0}{letters}{9}
\DeclareMathSymbol{\Omega}{0}{letters}{10}
\DeclareMathSymbol{\alpha}{0}{letters}{11}
\DeclareMathSymbol{\beta}{0}{letters}{12}
\DeclareMathSymbol{\gamma}{0}{letters}{13}
\DeclareMathSymbol{\delta}{0}{letters}{14}
\DeclareMathSymbol{\epsilon}{0}{letters}{15}
\DeclareMathSymbol{\zeta}{0}{letters}{16}
\DeclareMathSymbol{\eta}{0}{letters}{17}
\DeclareMathSymbol{\theta}{0}{letters}{18}
\DeclareMathSymbol{\iota}{0}{letters}{19}
\DeclareMathSymbol{\kappa}{0}{letters}{20}
\DeclareMathSymbol{\lambda}{0}{letters}{21}
\DeclareMathSymbol{\mu}{0}{letters}{22}
\DeclareMathSymbol{\nu}{0}{letters}{23}
\DeclareMathSymbol{\xi}{0}{letters}{24}
\DeclareMathSymbol{\pi}{0}{letters}{25}
\DeclareMathSymbol{\rho}{0}{letters}{26}
\DeclareMathSymbol{\sigma}{0}{letters}{27}
\DeclareMathSymbol{\tau}{0}{letters}{28}
\DeclareMathSymbol{\upsilon}{0}{letters}{29}
\DeclareMathSymbol{\phi}{0}{letters}{30}
\DeclareMathSymbol{\chi}{0}{letters}{31}
\DeclareMathSymbol{\psi}{0}{letters}{32}
\DeclareMathSymbol{\omega}{0}{letters}{33}
\DeclareMathSymbol{\varepsilon}{0}{letters}{34}
\DeclareMathSymbol{\vartheta}{0}{letters}{35}
\DeclareMathSymbol{\varpi}{0}{letters}{36}
\DeclareMathSymbol{\varrho}{0}{letters}{37}
\DeclareMathSymbol{\varsigma}{0}{letters}{38}
\DeclareMathSymbol{\varphi}{0}{letters}{39}
\fi\fi
```

Now let's define some of the other symbols in the OML encoding.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{\leftharpoonup}{0}{letters}{40}
\DeclareMathSymbol{\leftharpoondown}{0}{letters}{41}
\DeclareMathSymbol{\rightharpoonup}{0}{letters}{42}
\DeclareMathSymbol{\rightharpoondown}{0}{letters}{43}
\DeclareMathSymbol{\triangleright}{0}{letters}{46}
```

```
\DeclareMathSymbol{\triangleleft}{0}{letters}{47}
\DeclareMathSymbol{\flat}{0}{letters}{91}
\DeclareMathSymbol{\natural}{0}{letters}{92}
\DeclareMathSymbol{\sharp}{0}{letters}{93}
\DeclareMathSymbol{\smile}{0}{letters}{94}
\DeclareMathSymbol{\frown}{0}{letters}{95}
\DeclareMathSymbol{\ell}{0}{letters}{96}
\DeclareMathSymbol{\imath}{0}{letters}{123}
\DeclareMathSymbol{\jmath}{0}{letters}{124}
\DeclareMathSymbol{\wp}{0}{letters}{125}
\fi\fi
```

Now we go on to define the symbols from the OMS-encoded fonts.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{-}{2}{drmmathsy}{'000}
\DeclareMathSymbol{\cdot}{2}{drmmathsy}{'001}
\DeclareMathSymbol{\times}{2}{drmmathsy}{'002}
\DeclareMathSymbol{\ast}{2}{drmmathsy}{'003}
\DeclareMathSymbol{\div}{2}{drmmathsy}{'004}
\DeclareMathSymbol{\diamond}{2}{drmmathsy}{'005}
\DeclareMathSymbol{\pm}{2}{drmmathsy}{'006}
\DeclareMathSymbol{\mp}{2}{drmmathsy}{'007}
\DeclareMathSymbol{\oplus}{2}{drmmathsy}{'010}
\DeclareMathSymbol{\ominus}{2}{drmmathsy}{'011}
\DeclareMathSymbol{\otimes}{2}{drmmathsy}{'012}
\DeclareMathSymbol{\oslash}{2}{drmmathsy}{'013}
\DeclareMathSymbol{\odot}{2}{drmmathsy}{'014}
\DeclareMathSymbol{\bigcirc}{2}{drmmathsy}{'015}
\DeclareMathSymbol{\circ}{2}{drmmathsy}{'016}
\DeclareMathSymbol{\bullet}{2}{drmmathsy}{'017}
\DeclareMathSymbol{\asymp}{3}{drmmathsy}{'020}
\DeclareMathSymbol{\equiv}{3}{drmmathsy}{'021}
\DeclareMathSymbol{\subseteq}{3}{drmmathsy}{'022}
\DeclareMathSymbol{\supseteq}{3}{drmmathsy}{'023}
\DeclareMathSymbol{\leq}{3}{drmmathsy}{'024}
\DeclareMathSymbol{\geq}{3}{drmmathsy}{'025}
\DeclareMathSymbol{\preceq}{3}{drmmathsy}{'026}
\DeclareMathSymbol{\succeq}{3}{drmmathsy}{'027}
\DeclareMathSymbol{\sim}{3}{drmmathsy}{'030}
\DeclareMathSymbol{\approx}{3}{drmmathsy}{'031}
\DeclareMathSymbol{\subset}{3}{drmmathsy}{'032}
\DeclareMathSymbol{\supset}{3}{drmmathsy}{'033}
\DeclareMathSymbol{\ll}{3}{drmmathsy}{'034}
\DeclareMathSymbol{\gg}{3}{drmmathsy}{'035}
\DeclareMathSymbol{\prec}{3}{drmmathsy}{'036}
\DeclareMathSymbol{\succ}{3}{drmmathsy}{'037}
\DeclareMathSymbol{\simeq}{3}{drmmathsy}{'047}
\DeclareMathSymbol{\propto}{3}{drmmathsy}{'057}
\DeclareMathSymbol{\prime}{0}{drmmathsy}{'060}
\DeclareMathSymbol{'}{0}{drmmathsy}{'060}
```

```
\DeclareMathSymbol{\infty}{0}{drmmathsy}{'061}
\DeclareMathSymbol{\in}{0}{drmmathsy}{'062}
\DeclareMathSymbol{\ni}{0}{drmmathsy}{'063}
\DeclareMathSymbol{\bigtriangleup}{2}{drmmathsy}{'064}
\DeclareMathSymbol{\bigtriangledown}{2}{drmmathsy}{'065}
\DeclareMathSymbol{/}{2}{drmmathsy}{'066}
%\DeclareMathSymbol{'}{2}{drmmathsy}{'067}
\DeclareMathSymbol{\forall}{2}{drmmathsy}{'070}
\DeclareMathSymbol{\exists}{2}{drmmathsy}{'071}
\DeclareMathSymbol{\neg}{2}{drmmathsy}{'072}
\DeclareMathSymbol{\emptyset}{2}{drmmathsy}{'073}
\DeclareMathSymbol{\Im}{0}{drmmathsy}{'074}
\DeclareMathSymbol{\Re}{0}{drmmathsy}{'075}
\DeclareMathSymbol{\top}{0}{drmmathsy}{'076}
\DeclareMathSymbol{\bot}{0}{drmmathsy}{'077}
%\DeclareMathSymbol{\aleph}{0}{drmmathsy}{'080}
\DeclareMathSymbol{\cup}{2}{drmmathsy}{'133}
\DeclareMathSymbol{\cap}{2}{drmmathsy}{'134}
\DeclareMathSymbol{\uplus}{2}{drmmathsy}{'135}
\DeclareMathSymbol{\wedge}{2}{drmmathsy}{'136}
\DeclareMathSymbol{\vee}{2}{drmmathsy}{'137}
\DeclareMathSymbol{\vdash}{3}{drmmathsy}{'140}
\DeclareMathSymbol{\dashv}{3}{drmmathsy}{'141}
\fi\fi
```

Now we define some arrow symbols; there is a surprisingly large variety of these.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{\leftarrow}{0}{drmmathsy}{'040}
\DeclareMathSymbol{\rightarrow}{0}{drmmathsy}{'041}
\DeclareMathSymbol{\leftrightarrow}{0}{drmmathsy}{'044}
\DeclareMathSymbol{\nearrow}{0}{drmmathsy}{'045}
\DeclareMathSymbol{\searrow}{0}{drmmathsy}{'046}
\DeclareMathSymbol{\Leftarrow}{0}{drmmathsy}{'050}
\DeclareMathSymbol{\Rightarrow}{0}{drmmathsy}{'051}
\DeclareMathSymbol{\Leftrightarrow}{0}{drmmathsy}{'054}
\DeclareMathSymbol{\nwarrow}{0}{drmmathsy}{'055}
\DeclareMathSymbol{\swarrow}{0}{drmmathsy}{'056}
\DeclareMathSymbol{\wr}{2}{drmmathsy}{'157}
\DeclareMathSymbol{\surd}{0}{drmmathsy}{'160}
\DeclareMathSymbol{\amalg}{2}{drmmathsy}{'161}
\DeclareMathSymbol{\nabla}{0}{drmmathsy}{'162}
\DeclareMathSymbol{\sqcup}{2}{drmmathsy}{'164}
\DeclareMathSymbol{\sqcap}{2}{drmmathsy}{'165}
\DeclareMathSymbol{\sqsubseteq}{2}{drmmathsy}{'166}
\DeclareMathSymbol{\sqsupseteq}{2}{drmmathsy}{'167}
\DeclareMathSymbol{\dagger}{2}{drmmathsy}{'171}
\DeclareMathSymbol{\ddagger}{2}{drmmathsy}{'172}
\DeclareMathSymbol{:}{2}{operators}{'072}
\DeclareMathSymbol{;}{0}{operators}{'073}
```

```
\DeclareMathSymbol{.}{0}{letters}{'072}
\DeclareMathSymbol{,}{0}{operators}{'054}
\DeclareMathSymbol{\ldotp}{0}{letters}{'072}
\DeclareMathSymbol{\clubsuit}{0}{letters}{'174}
\DeclareMathSymbol{\diamondsuit}{0}{letters}{'175}
\DeclareMathSymbol{\heartsuit}{0}{letters}{'176}
\DeclareMathSymbol{\spadesuit}{0}{letters}{'177}
\DeclareMathSymbol{\partial}{0}{drmmathsy}{'100}
\fi\fi
```

Now we define some of the large/small symbols, like `\sum` and `\prod`. It proved necessary to cancel out the previous definitions of these, or LaTeX complained about them being already defined; it seems that it ought to be possible to redefine them only for a given math version, but I haven't figured it out yet.

```
\ifnodefault\else\ifnodefaultmath\else
\let\coprod\relax
\DeclareMathSymbol{\coprod}{\mathop}{drmmathomx}{"60}
\let\bigvee\relax
\DeclareMathSymbol{\bigvee}{\mathop}{drmmathomx}{"57}
\let\bigwedge\relax
\DeclareMathSymbol{\bigwedge}{\mathop}{drmmathomx}{"56}
\let\biguplus\relax
\DeclareMathSymbol{\biguplus}{\mathop}{drmmathomx}{"55}
\let\bigcap\relax
\DeclareMathSymbol{\bigcap}{\mathop}{drmmathomx}{"54}
\let\bigcup\relax
\DeclareMathSymbol{\bigcup}{\mathop}{drmmathomx}{"53}
\let\intop\relax
\DeclareMathSymbol{\intop}{\mathop}{drmmathomx}{"52}
    \def\int{\intop\nolimits}
\let\prod\relax
\DeclareMathSymbol{\prod}{\mathop}{drmmathomx}{"51}
\let\sum\relax
\DeclareMathSymbol{\sum}{\mathop}{drmmathomx}{"50}
\let\bigotimes\relax
\DeclareMathSymbol{\bigotimes}{\mathop}{drmmathomx}{"4E}
\let\bigoplus\relax
\DeclareMathSymbol{\bigoplus}{\mathop}{drmmathomx}{"4C}
\let\bigodot\relax
\DeclareMathSymbol{\bigodot}{\mathop}{drmmathomx}{"4A}
\let\ointcup\relax
\DeclareMathSymbol{\ointop}{\mathop}{drmmathomx}{"48}
    \def\oint{\ointop\nolimits}
\let\bigsqcup\relax
\DeclareMathSymbol{\bigsqcup}{\mathop}{drmmathomx}{"46}
\fi\fi
```

Moving on to delimiters.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathSymbol{|}{0}{drmmathsy}{'152}
```

```
\let\backslash\relax\DeclareMathSymbol{\backslash}{0}{drmmathsy}{'156}
\fi\fi
```

Declare the math accents.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathAccent{\vec}{\mathord}{letters}{126}
\DeclareMathAccent{\acute}{\mathalpha}{operators}{1}
\DeclareMathAccent{\hat}{\mathalpha}{operators}{2}
\DeclareMathAccent{\grave}{\mathalpha}{operators}{0}
\DeclareMathAccent{\check}{\mathalpha}{operators}{7}
\DeclareMathAccent{\bar}{\mathalpha}{operators}{9}
\DeclareMathAccent{\dot}{\mathalpha}{operators}{10}
\DeclareMathAccent{\ddot}{\mathalpha}{operators}{4}
\DeclareMathAccent{\breve}{\mathalpha}{operators}{8}
\DeclareMathAccent{\tilde}{\mathalpha}{operators}{3}
\fi\fi
```

Declare our math delimiters, so that TEX's delimiter-expanding magic can work with our new characters.

```
\def\bigd#1{\bBigg@{#1}}
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathDelimiter{(}{\mathopen}{operators}{"28}
{drmmathomx}{"00}
\DeclareMathDelimiter{)}{\mathopen}{operators}{"29}
{drmmathomx}{"01}
\let\{\relax\let\}\relax
\DeclareMathDelimiter{\{}{\mathopen}{operators}{"7B}
{drmmathomx}{"08}
\DeclareMathDelimiter{\}}{\mathopen}{operators}{"7D}
{drmmathomx}{"09}
\DeclareMathDelimiter{[}{\mathopen}{operators}{"5B}
{drmmathomx}{"02}
\DeclareMathDelimiter{]}{\mathopen}{operators}{"5D}
{drmmathomx}{"03}
\DeclareMathDelimiter{\lfloor}{\mathopen}{drmmathsy}{"62}
{drmmathomx}{"04}
\DeclareMathDelimiter{\rfloor}{\mathclose}{drmmathsy}{"63}
{drmmathomx}{"05}
\DeclareMathDelimiter{\lceil}{\mathopen}{drmmathsy}{"64}
{drmmathomx}{"06}
\DeclareMathDelimiter{\rceil}{\mathclose}{drmmathsy}{"65}
{drmmathomx}{"07}
\DeclareMathDelimiter{\langle}{\mathopen}{drmmathsy}{"68}
{drmmathomx}{"0A}
\DeclareMathDelimiter{\rangle}{\mathclose}{drmmathsy}{"69}
{drmmathomx}{"0B}
\DeclareMathDelimiter{|}{\mathclose}{drmmathomx}{"0C}
{drmmathomx}{"0C}
\DeclareMathDelimiter{\vert}{\mathclose}{drmmathomx}{"0C}
{drmmathomx}{"0C}
\DeclareMathDelimiter{\|}{\mathclose}{drmmathomx}{"0D}
```

```
{drmmathomx}{"0D}
\DeclareMathDelimiter{\Vert}{\mathclose}{drmmathomx}{"0D}
{drmmathomx}{"0D}
\DeclareMathDelimiter{\uparrow}{\mathrel}{drmmathomx}{"78}
{drmmathomx}{"78}
\DeclareMathDelimiter{\downarrow}{\mathrel}{drmmathomx}{"79}
{drmmathomx}{"79}
\DeclareMathDelimiter{\updownarrow}{\mathrel}{drmmathsy}{"6C}
{drmmathomx}{"3F}
\DeclareMathDelimiter{\Uparrow}{\mathrel}{drmmathsy}{"2A}
{drmmathomx}{"7E}
\DeclareMathDelimiter{\Downarrow}{\mathrel}{drmmathsy}{"2B}
{drmmathomx}{"7F}
\DeclareMathDelimiter{\Updownarrow}{\mathrel}{drmmathsy}{"6D}
{drmmathomx}{"77}
\fi\fi
```

Next, we define a *math radical*, which essentially means a square root sign. Curiously, the thickness of the rule enclosing the square root sign is governed by the *height* of the square root character; this means that almost the entire character is *depth*. I had to jimmy a bit with the default LaTeX `\sqrt` definition to make the root numbers (say, the 3 for the cube root) line up properly, as well, which is what all the `\r@@t` business here is.

```
\ifnodefault\else\ifnodefaultmath\else
\DeclareMathRadical{\sqrtsign}{drmmathsy}{"70}{drmmathomx}{"70}
\DeclareRobustCommand\sqrt{\@ifnextchar[\@sqrt\sqrtsign}
\def\r@@t#1#2{
\setbox\z@\hbox{$\m@th#1\sqrtsign{#2}$}
\dimen@\ht\z@ \advance\dimen@-\dp\z@
\mkern5mu\raise.8\dimen@\copy\rootbox
\mkern-7mu\box\z@}
\fi\fi
```

Now, finally, we declare `drmmath` to be the default math version, so that all this will become the norm in a document declaring the `drm` package. Unless, of course, either `nodefault` or `nodefaultmath` has been specified as an option.

```
\ifnodefault\else\ifnodefaultmath\else
\mathversion{drmmath}
\fi\fi
```

Now we define the special symbols. First, we define `\drmsym`, which takes a single argument to be typeset from the `drmsym` font. Then we define a (rather huge) macro for redefining all the symbols. This macro will be called only if `nodefault` or `nodefaulttext` have not been selected, or if `symbolsonly` has been selected.

```
\def\drmsym#1{{\fontencoding{TS1}\selectfont\fontfamily{drm}\selectfont#1}}
\def\drmsymbolredef{
\def\textbigcircle{\drmsym{\char'117}}
\def\textregistered{\drmsym{\char'256}}
\def\texttrademark{\drmsym{\char'227}}
```

```
\def\textservicemark{\drmsym{\char'237}}
\def\textsoundrecording{\drmsym{\char'255}}
\def\textcopyright{\drmsym{\char'251}}
\def\textcopyleft{\drmsym{\char'253}}
\def\textborn{\drmsym{\char'142}}
\def\textdied{\drmsym{\char'144}}
\def\textdivorced{\drmsym{\char'143}}
\def\textmarried{\drmsym{\char'155}}
\def\textleaf{\drmsym{\char'154}}
\def\textmale{\drmsym{\char'153}}
\def\textfemale{\drmsym{\char'145}}
\def\textcrusadecross{\drmsym{\char'130}}
\def\textcrusadecrossoutline{\drmsym{\char'131}}
\def\textlatincross{\drmsym{\char'144}}
\def\textlatincrossoutline{\drmsym{\char'134}}
\def\textgreekcross{\drmsym{\char'170}}
\def\textgreekcrossoutline{\drmsym{\char'171}}
\def\textsaltirecross{\drmsym{\char'172}}
\def\textsaltirecrossoutline{\drmsym{\char'173}}
\def\texteucharist{\drmsym{\char'120}}
\def\textstardavid{\drmsym{\char'140}}
\def\textstardavidsolid{\drmsym{\char'141}}
\def\textstardavidoutline{\drmsym{\char'151}}
\def\textsun{\drmsym{\char'330}}
\def\textsunvar{\drmsym{\char'331}}
\def\textwaxcrescent{\drmsym{\char'332}}
\def\textfullmoon{\drmsym{\char'333}}
\def\textwanecrescent{\drmsym{\char'334}}
\def\textnewmoon{\drmsym{\char'335}}
\def\textmercury{\drmsym{\char'336}}
\def\textearth{\drmsym{\char'337}}
\def\textterra{\drmsym{\char'337}}
\def\textearthvar{\drmsym{\char'340}}
\def\textterravar{\drmsym{\char'340}}
\def\textmars{\drmsym{\char'153}}
\def\textvenus{\drmsym{\char'145}}
\def\textjupiter{\drmsym{\char'341}}
\def\textsaturn{\drmsym{\char'342}}
\def\texturanus{\drmsym{\char'343}}
\def\texturanusvar{\drmsym{\char'344}}
\def\textneptune{\drmsym{\char'345}}
\def\textceres{\drmsym{\char'346}}
\def\textpallas{\drmsym{\char'347}}
\def\textjuno{\drmsym{\char'350}}
\def\textjunovar{\drmsym{\char'351}}
\def\textvesta{\drmsym{\char'352}}
\def\textvestavar{\drmsym{\char'353}}
\def\textastraea{\drmsym{\char'354}}
\def\textastraeavar{\drmsym{\char'355}}
\def\texthebe{\drmsym{\char'356}}
```

```
\def\textiris{\drmsym{\char'357}}
\def\textaries{\drmsym{\char'360}}
\def\textari{\drmsym{\char'360}}
\def\texttaurus{\drmsym{\char'361}}
\def\texttau{\drmsym{\char'361}}
\def\textgemini{\drmsym{\char'362}}
\def\textgem{\drmsym{\char'362}}
\def\textcancer{\drmsym{\char'363}}
\def\textcnc{\drmsym{\char'363}}
\def\textleo{\drmsym{\char'364}}
\def\textvirgo{\drmsym{\char'365}}
\def\textvir{\drmsym{\char'365}}
\def\textlibra{\drmsym{\char'367}}
\def\textlib{\drmsym{\char'367}}
\def\textscorpius{\drmsym{\char'370}}
\def\textsco{\drmsym{\char'370}}
\def\textsagittarius{\drmsym{\char'371}}
\def\textsgr{\drmsym{\char'371}}
\def\textcapricorn{\drmsym{\char'372}}
\def\textcap{\drmsym{\char'372}}
\def\textaquarius{\drmsym{\char'373}}
\def\textaqr{\drmsym{\char'373}}
\def\textpisces{\drmsym{\char'374}}
\def\textpsc{\drmsym{\char'374}}
\def\textpluto{\drmsym{\char'375}}
\def\textplutovar{\drmsym{\char'376}}
\def\textstar{\drmsym{\char'142}}
\def\textcomet{\drmsym{\char'377}}
\def\textquadrature{\drmsym{\char'310}}
\def\textopposition{\drmsym{\char'311}}
\def\textconjunction{\drmsym{\char'312}}
\def\textascendingnode{\drmsym{\char'315}}
\def\textdescendingnode{\drmsym{\char'314}}
\def\textdollarsign{\drmsym{\char'044}}
\def\textolddollarsign{\drmsym{\char'212}}
\def\textcentsign{\drmsym{\char'242}}
\def\textoldcentsign{\drmsym{\char'213}}
\def\textpoundsterling{\drmsym{\char'243}}
\def\textoldpoundsterling{\drmsym{\char'222}}
\def\textlira{\drmsym{\char'222}}
\def\texteuro{\drmsym{\char'277}}
\def\textyen{\drmsym{\char'245}}
\def\textbaht{\drmsym{\char'232}}
\def\textcolon{\drmsym{\char'215}}
\def\textdong{\drmsym{\char'226}}
\def\textflorin{\drmsym{\char'214}}
\def\textguarani{\drmsym{\char'220}}
\def\textnaira{\drmsym{\char'217}}
\def\textpeso{\drmsym{\char'221}}
\def\textruble{\drmsym{\char'221}}
```

```
\def\textwon{\drmsym{\char'216}}
\def\textcurrency{\drmsym{\char'244}}
\def\romone{\drmsym{\char'100}}
\def\romfive{\drmsym{\char'101}}
\def\romten{\drmsym{\char'102}}
\def\romfifty{\drmsym{\char'103}}
\def\romhundred{\drmsym{\char'104}}
\def\romfivehundred{\drmsym{\char'105}}
\def\romthousand{\drmsym{\char'106}}
\def\liningzero{\drmsym{\char'060}}
\def\liningone{\drmsym{\char'061}}
\def\liningtwo{\drmsym{\char'062}}
\def\liningthree{\drmsym{\char'063}}
\def\liningfour{\drmsym{\char'064}}
\def\liningfive{\drmsym{\char'065}}
\def\liningsix{\drmsym{\char'066}}
\def\liningseven{\drmsym{\char'067}}
\def\liningeight{\drmsym{\char'070}}
\def\liningnine{\drmsym{\char'071}}
\def\textnumero{\drmsym{\char'233}}
\def\textrefmark{\drmsym{\char'270}}
\def\textasterism{\drmsym{\char'302}}
\def\textfemininеordinal{\drmsym{\char'252}}
\def\textmasculineordinal{\drmsym{\char'272}}
\def\textsupone{\drmsym{\char'271}}
\def\textsuptwo{\drmsym{\char'262}}
\def\textsupthree{\drmsym{\char'263}}
\def\textpilcrowsolid{\drmsym{\char'231}}
\def\textpilcrowoutline{\drmsym{\char'266}}
\def\textsection{\drmsym{\char'247}}
\def\textdagger{\drmsym{\char'204}}
\def\textdag{\drmsym{\char'204}}
\def\dag{\drmsym{\char'204}}
\def\textdbldagger{\drmsym{\char'205}}
\def\textdbldag{\drmsym{\char'205}}
\def\dbldag{\drmsym{\char'205}}
\def\textpipe{\drmsym{\char'206}}
\def\textbrokenpipe{\drmsym{\char'246}}
\def\textprime{\drmsym{\char'264}}
\def\textdoubleprime{\drmsym{\char'303}}
\def\texttripleprime{\drmsym{\char'313}}
\def\textsqrt{\drmsym{\char'273}}
\def\textquarter{\drmsym{\char'274}}
\def\texthalf{\drmsym{\char'275}}
\def\textthreequarters{\drmsym{\char'276}}
\def\textthird{\drmsym{\char'304}}
\def\texttwothirds{\drmsym{\char'305}}
\def\textpermille{\drmsym{\char'207}}
\def\textperbiqua{\drmsym{\char'207}}
\def\textpertenmille{\drmsym{\char'230}}
```

```
\def\textpertriqua{\drmsym{\char'230}}
\def\textequals{\drmsym{\char'055}}
\def\textslash{\drmsym{\char'057}}
\def\texttimes{\drmsym{\char'326}}
\def\textdiv{\drmsym{\char'366}}
\def\textradiation{\drmsym{\char'161}}
\def\textradiationnocircle{\drmsym{\char'160}}
\def\textbiohazard{\drmsym{\char'163}}
\def\textbiohazardnocircle{\drmsym{\char'162}}
\def\texthighvoltage{\drmsym{\char'166}}
\def\texthighvoltagenotriangle{\drmsym{\char'165}}
\def\textgeneralwarning{\drmsym{\char'164}}
\def\textuparrow{\drmsym{\char'136}}
\def\textdownarrow{\drmsym{\char'137}}
\def\textleftarrow{\drmsym{\char'030}}
\def\textrightarrow{\drmsym{\char'031}}
\def\textrecipe{\drmsym{\char'223}}
\def\textintbang{\drmsym{\char'224}}
\def\textopenintbang{\drmsym{\char'225}}
\def\textbullet{\drmsym{\char'210}}
\def\textopenbullet{\drmsym{\char'236}}
\def\textheart{\drmsym{\char'174}}
\def\textopenheart{\drmsym{\char'175}}
\def\texteighthnote{\drmsym{\char'156}}
\def\textdiamond{\drmsym{\char'306}}
\def\textopendiamond{\drmsym{\char'307}}
\def\textlozenge{\drmsym{\char'307}}
\def\textdegree{\drmsym{\char'260}}
\def\texttilde{\drmsym{\char'176}}
\def\textasciitilde{\texttilde}
\def\tilde{\texttilde}
\def\textasciicircum{\drmsym{\char'002}}
\def\textdegreec{\drmsym{\char'211}}
\def\textrightupfleuron{\drmsym{\char'016}}
\def\textrightdownfleuron{\drmsym{\char'017}}
\def\textleftupfleuron{\drmsym{\char'020}}
\def\textleftdownfleuron{\drmsym{\char'021}}
\def\textupleftfleuron{\drmsym{\char'050}}
\def\textuprightfleuron{\drmsym{\char'051}}
\def\textdownrightfleuron{\drmsym{\char'077}}
\def\textdownleftfleuron{\drmsym{\char'107}}
\def\textsquaretulip{\drmsym{\char'023}}
\def\textsquaretulipside{\drmsym{\char'046}}
\def\textupdoubletulip{\drmsym{\char'024}}
\def\textdowndoubletulip{\drmsym{\char'027}}
\def\textrightdoubletulip{\drmsym{\char'036}}
\def\textleftdoubletulip{\drmsym{\char'037}}
\def\textupleftcornertulip{\drmsym{\char'053}}
\def\textuprightcornertulip{\drmsym{\char'072}}
\def\textlowleftcornertulip{\drmsym{\char'073}}
```

```
\def\textlowrightcornertulip{\drmsym{\char'110}}
\def\textupsingletuliplong{\drmsym{\char'111}}
\def\textdownsingletuliplong{\drmsym{\char'112}}
\def\textleftsingletuliplong{\drmsym{\char'113}}
\def\textrightsingletuliplong{\drmsym{\char'114}}
\def\textupsingletulip{\drmsym{\char'116}}
\def\textdownsingletulip{\drmsym{\char'121}}
\def\textleftsingletulip{\drmsym{\char'122}}
\def\textrightsingletulip{\drmsym{\char'123}}
\def\spearright{\drmsym{\char'124}}
\def\spearleft{\drmsym{\char'125}}
\def\horizspearext{\drmsym{\char'126}}
\def\spearup{\drmsym{\char'132}}
\def\speardown{\drmsym{\char'146}}
\def\vertspearext{\drmsym{\char'147}}
\def\fleurdelis{\drmsym{\char'157}}
\def\fleurdelys{\drmsym{\char'157}}
\def\fleurdelisdown{\drmsym{\char'167}}
\def\fleurdelysdown{\drmsym{\char'167}}
\def\fleurdelisleft{\drmsym{\char'316}}
\def\fleurdelysleft{\drmsym{\char'316}}
\def\fleurdelisright{\drmsym{\char'177}}
\def\fleurdelysright{\drmsym{\char'177}}
\def\woundcordleftext{\drmsym{\char'317}}
\def\woundcordrightext{\drmsym{\char'324}}
\def\woundcordleftend{\drmsym{\char'320}}
\def\woundcordrightend{\drmsym{\char'321}}
\def\woundcordleftendinv{\drmsym{\char'323}}
\def\woundcordrightendinv{\drmsym{\char'322}}
}
\ifnodefault\else\ifnodefaulttext\else
\drmsymbolredef
\fi\fi
\ifsymbolsonly\drmsymbolredef\fi
```

Now, rather than require people to enter the Roman numeral macros by hand, we provide a command, `\romanize`, which takes as its only argument an Indo-Arabic numeral and converts it into a Roman numeral. This macro is a thin wrapper around one from the `modroman` package, and in fact requires `modroman` to work.

```
\def\romanize#1{%
\RedefineMRmdclxvij{\romthousand}{\romfivehundred}
{\romhundred}{\romfifty}{\romten}{\romfive}{\romone}{\romone}%
\nbshortroman{#1}%
}%
```

We also define a command for producing lining numerals rather than old-style figures, so that these long-winded command names don't need to be typed if lining numerals will be used frequently. It takes the number to be output as lining as its only argument.

```
\def\liningnums#1{%
\drmsym{#1}%
}%
```

Now we define the decorative tulip frame macro, as an example of the beautiful constructions which are possible with fleurons and other textual ornaments.

```
\def\tulipframe#1{%
\vbox{%
\hbox to\linewidth{\hfil%
{\drmsym{\char'053}}%
{\drmsym{\char'111}}%
{\drmsym{\char'024}}%
{\drmsym{\char'111}}%
{\drmsym{\char'072}}\hfil}%
\vskip-0.5\baselineskip%
\hbox to\linewidth{\hfil%
#1%
\hfil%
}%
\vskip-0.5\baselineskip%
\hbox to\linewidth{\hfil%
{\drmsym{\char'073}}%
{\drmsym{\char'112}}%
{\drmsym{\char'027}}%
{\drmsym{\char'112}}%
{\drmsym{\char'110}}\hfil}%
}%
}%
```

Next, we define the macros for the extensible rules. Lots of down-and-dirty TEX stuff here.

```
\newcount\counterA
\newcount\counterB
\newcount\iter
\newlength{\extcharwid}
\newlength{\leftcharwid}
\newlength{\rightcharwid}
\newlength{\greaterwid}
\def\extrule#1#2#3#4#5{%
\if#1h%
\settowidth{\extcharwid}{#5}%
\settowidth{\leftcharwid}{#3}%
\settowidth{\rightcharwid}{#4}%
\counterB=\numexpr\dimexpr#2\relax\relax%
\advance\counterB by-\leftcharwid%
\advance\counterB by-\rightcharwid%
\counterA=\dimexpr\extcharwid\relax%
\divide\counterB by\counterA%
\iter=0%
\noindent#3%
```

```
\loop%
\ifnum\iter<\counterB%
\advance\iter by 1%
#5%
\repeat%
#4%
\fi%
\if#1v%
\setbox0=\hbox{#3}%
\leftcharwid=\ht0\advance\leftcharwid by\dp0%
\setbox0=\hbox{#4}%
\rightcharwid=\ht0\advance\rightcharwid by\dp0%
\setbox0=\hbox{#5}%
\extcharwid=\ht0\advance\extcharwid by\dp0%
\ifdim\leftcharwid>\rightcharwid%
\greaterwid=\leftcharwid%
\else%
\greaterwid=\rightcharwid%
\fi\if\extcharwid>\greaterwid%
\greaterwid=\extcharwid%
\fi%
\counterB=\numexpr\dimexpr#2\relax\relax%
\advance\counterB by-\leftcharwid%
\advance\counterB by-\rightcharwid%
\counterA=\dimexpr\extcharwid\relax%
\divide\counterB by\counterA%
\iter=0%
\noindent\vbox to\dimexpr#2{\baselineskip=0pt%
\hbox to\greaterwid{\hfil#4\hfil}%
\loop%
\ifnum\iter<\counterB%
\advance\iter by 1%
\vss\hbox to\greaterwid{\hfil#5\hfil}%
\repeat%
\vss\hbox to\greaterwid{\hfil#3\hfil}}%
\fi%
}%
```

Next, we move on to define the unreasonably complex and configurable ellipsis commands. First we define the `\drmelip`, then the four-dotted `\drmfelip`.

```
\newlength{\drmelipgap}\setlength{\drmelipgap}{2.9pt}
\newlength{\drmelipbef}\setlength{\drmelipbef}{2.4pt}
\newlength{\drmelipaft}\setlength{\drmelipaft}{1.4pt}
\def\drmelipchar{.}
\def\drmelip{%
\hbox{%
\hbox to\the\drmelipbef{\hfil}%
\drmelipchar%
\hbox to\drmelipgap{\hfil}%
\drmelipchar%
```

```
\hbox to\drmelipgap{\hfil}%
\drmelipchar%
\hbox to\drmelipaft{\hfil}%
}%
}%
\newlength{\drmfelipbef}\setlength{\drmfelipbef}{0pt}
\newlength{\drmfelipaft}\setlength{\drmfelipaft}{\the\drmelipaft}
\newlength{\drmfelipwid}
\def\drmfelip{%
\hbox{%
\hbox to\the\drmfelipbef{\hfil}%
\drmelipchar%
\hbox to\drmelipgap{\hfil}%
\drmelipchar%
\hbox to\drmelipgap{\hfil}%
\drmelipchar%
\hbox to\drmelipgap{\hfil}%
\drmelipchar%
\hbox to\drmfelipaft{\hfil}%
}%
}%
```

Now, we begin the decorative initials. These are designed using a common background written in METAPOST with a DRM figure superimposed, so a great deal of the code in this section is, in fact, METAPOST rather than TEX or METAFONT.

We begin by defining `\drmdecinit`, which takes five arguments: the width, the height, the color of the background, the color of the letter, and the letter itself. It includes a METAPOST macro, `along`, derived from http://tex.stackexchange.com/questions/176665/define-a-pair-point-along-a-path-length-metapost.

```
\def\drmdecinitfontdefault{%
\def\drmdecinitfont{%
\unexpanded{\font\drminitfontcom=drm10}
}%
}%
\def\drmdecinitfont{%
\unexpanded{\font\drminitfontcom=drm10}  %
}%
\def\drmdecinit#1#2#3#4#5{%
\begin{mpost}
primarydef pct along pat =
(arctime (pct * (arclength pat)) of pat) of pat
enddef;
w=#1; h=#2;
pen thinpen; thinpen = pencircle scaled (w/288);
pen medpen; medpen = pencircle scaled (w/144);
pen thickpen; thickpen = pencircle scaled (w/144 + w/288);
pen ththickpen; ththickpen = pencircle scaled (w/72);
pen thththickpen; thththickpen = pencircle scaled (w/36);
leaflen = w/6.5;
```

```
leafletlen = w/20;
leafletwid = w/40;
leafletgap = w/40;
pen leafpen; leafpen = pencircle xscaled leafletlen
yscaled leafletwid;
pen sideleafpen; sideleafpen = pencircle yscaled leafletlen
xscaled leafletwid rotated -35;
def border =
pickup thththickpen;
draw top lft (0,h)--top rt(w,h)--bot rt(w,0)--bot
lft(0,0)--cycle withcolor #3;
pickup ththickpen;
draw (top lft (0,h)--top rt(w,h)--bot rt(w,0)--bot
lft(0,0)--cycle) scaled 0.95 shifted (0.025w,0.025h)
withcolor #3;
enddef;
def leaf(expr p,s,r,t) =
path leafpath;
leafpath = ((p shifted (leafletlen/2,0))..
(p shifted (0,leafletwid/2))..
(p shifted (-leafletlen/2,0))..
(p shifted (0,-leafletwid/2))..cycle)
rotatedaround (p,s);
if t = 0:
fill leafpath withcolor r;
elseif t = 1:
fill leafpath reflectedabout
((w/2,h),(w/2,0)) withcolor r;
elseif t = 2:
fill leafpath reflectedabout
((0,h/2),(w,h/2)) withcolor r;
elseif t = 3:
fill leafpath reflectedabout
((0,h/2),(w,h/2)) reflectedabout ((w/2,h),(w/2,0))
withcolor r;
fi
enddef;
def branch(expr p,s,b) =
pickup thickpen;
pair t; t = p rotatedaround (p,s);
pair u; u = point 1.0 along (t{dir (s+90)}..
t shifted (0.3leaflen,leaflen) rotatedaround (t,s));
pair q; q = (t shifted (1.4leafletwid,0)) rotatedaround (t,s);
pair v; v = (q shifted (0.3leaflen,leaflen)) rotatedaround (q,s);
pair r; r = (t shifted (-1.4leafletwid,0)) rotatedaround (t,s);
pair a; a = (r shifted (0.3leaflen,leaflen)) rotatedaround (r,s);
if b = 0:
draw (t{dir (s+90)}..u) withcolor #3;
elseif b = 1:
draw (t{dir (s+90)}..u) reflectedabout ((w/2,h),(w/2,0))
```

```
withcolor #3;
elseif b = 2:
draw (t{dir (s+90)}..u) reflectedabout ((0,h/2),(w,h/2))
withcolor #3;
elseif b = 3:
draw (t{dir (s+90)}..u) reflectedabout ((0,h/2),(w,h/2))
reflectedabout ((w/2,h),(w/2,0)) withcolor #3;
fi
leaf((point 0.15 along (q{dir (s+90)}..v)),s,#3,b);
leaf((point 0.45 along (q{dir (s+90)}..v)),s,#3,b);
leaf((point 0.75 along (q{dir (s+90)}..v)),s,#3,b);
leaf((point 0.15 along (r{dir (s+90)}..a)),s,#3,b);
leaf((point 0.45 along (r{dir (s+90)}..a)),s,#3,b);
leaf((point 0.75 along (r{dir (s+90)}..a)),s,#3,b);
leaf((point 0.98 along (t{dir (s+90)}..u)),s+60,white,b);
leaf((point 1.00 along (t{dir (s+90)}..u)),s+60,#3,b);
enddef;
def football(expr p) =
pickup thinpen;
draw (z26..z30..z31..z26..z32..z33..cycle)
rotatedaround ((w/2,h/2),p) withcolor #3;
pickup ththickpen;
draw (z20..z24..z21) rotatedaround ((w/2,h/2),p)
withcolor white;
draw (z21..z25..z20) rotatedaround ((w/2,h/2),p)
withcolor white;
draw (z20..tension 1.4..z22..z23..tension 1.5..z20)
rotatedaround ((w/2,h/2),p) withcolor white;
draw (z21..tension 1.4..z22..z23..tension 1.5..z21)
rotatedaround ((w/2,h/2),p) withcolor white;
pickup medpen;
draw (z20..z24..z21) rotatedaround ((w/2,h/2),p)
withcolor #3;
draw (z21..z25..z20) rotatedaround ((w/2,h/2),p)
withcolor #3;
draw (z20..tension 1.4..z22..z23..tension 1.5..z20)
rotatedaround ((w/2,h/2),p) withcolor #3;
draw (z21..tension 1.4..z22..z23..tension 1.5..z21)
rotatedaround ((w/2,h/2),p) withcolor #3;
fill (z34..z36..z35..z37..cycle) rotatedaround
((w/2,h/2),p) withcolor #3;
fill (z38..z40..z39..z41..cycle) rotatedaround
((w/2,h/2),p) withcolor #3;
fill (z42..z44..z43..z45..cycle) rotatedaround
((w/2,h/2),p) withcolor #3;
enddef;
border;
z0 = (w-3.3leafletwid-(w/11),h-leaflen-(w/144));
z1 = (w/2,2h/3);%h-leafletwid-2pt);
z2 = z0 reflectedabout ((w/2,h),(w/2,0));
```

```
z3 = (2w/3,h/2);%w-leafletwid-2pt,h/2);
z4 = z0 reflectedabout ((w,h/2),(0,h/2));
z5 = z1 reflectedabout ((0,h/2),(w,h/2));
z6 = z4 reflectedabout ((w/2,h),(w/2,0));
z7 = z3 reflectedabout ((w/2,h),(w/2,0));
pickup thickpen;
draw z0{dir -120}..{left}z1{left}..{dir 120}z2 withcolor #3;
draw z0{dir -120}..{down}z3{down}..{dir -60}z4 withcolor #3;
draw z4{dir 120}..{left}z5{left}..{dir -120}z6 withcolor #3;
draw z6{dir 60}..{up}z7{up}..{dir 120}z2 withcolor #3;
branch(z0,-30,0);
branch(z0,-30,1);
branch(z0,-30,2);
branch(z0,-30,3);
z10 = (w/2,h-leafletwid-(w/72));
z11 = z10 rotatedaround ((w/2,h/2),90);
z12 = z10 rotatedaround ((w/2,h/2),180);
z13 = z10 rotatedaround ((w/2,h/2),270);
path greatcirc; greatcirc = z10..z11..z12..z13..cycle;
pickup thththickpen;
draw greatcirc withcolor white;
pickup ththickpen;
draw greatcirc withcolor #3;
z20 = z2 shifted (0,-leafletlen);
z21 = z6 shifted (0,leafletlen);
z22 = z11 shifted (leafletlen,0);
z23 = z7 shifted (-leafletlen,0);
z24 = 0.25[z11,z7];
z25 = 0.75[z11,z7];
z26 = 0.5[z11,z7];
z27 = 0.25[z2,z6];
z28 = 0.5[z2,z6];
z29 = 0.75[z2,z6];
z30 = 0.5[z11,z2];
z31 = 0.5[z7,z2];
z32 = 0.5[z6,z11];
z33 = 0.5[z6,z7];
z34 = z26 shifted (-leafletlen,0);
z35 = z26 shifted (leafletlen,0);
z36 = z26 shifted (0,leafletwid);
z37 = z26 shifted (0,-leafletwid);
z38 = z27 shifted (-0.8leafletwid,0);
z39 = z27 shifted (0.8leafletwid,0);
z40 = z27 shifted (0,0.8leafletlen);
z41 = z27 shifted (0,-0.8leafletlen);
z42 = z29 shifted (-0.8leafletwid,0);
z43 = z29 shifted (0.8leafletwid,0);
z44 = z29 shifted (0,0.8leafletlen);
z45 = z29 shifted (0,-0.8leafletlen);
football(0);
```

```
1183 football(90);
1184 football(180);
1185 football(270);
1186 z50 = z1 shifted (0,-leafletwid);
1187 z51 = z3 shifted (-leafletwid,0);
1188 z52 = z5 shifted (0,leafletwid);
1189 z53 = z7 shifted (leafletwid,0);
1190 z54 = 0.4[(w/2,h/2),(0,h)];
1191 z55 = 0.4[(w/2,h/2),(w,h)];
1192 z56 = 0.4[(w/2,h/2),(w,0)];
1193 z57 = 0.4[(w/2,h/2),(0,0)];
1194 pickup thickpen;
1195 draw z50..z51..z52..z53..cycle withcolor #3;
1196 pickup medpen;
1197 draw z50{left}..z54 withcolor #3;
1198 draw z50{right}..z55 withcolor #3;
1199 draw z51{up}..z55 withcolor #3;
1200 draw z51{down}..z56 withcolor #3;
1201 draw z52{right}..z56 withcolor #3;
1202 draw z52{left}..z57 withcolor #3;
1203 draw z53{down}..z57 withcolor #3;
1204 draw z53{up}..z54 withcolor #3;
1205 z60 = z50 shifted (0,-leafletwid);
1206 z61 = z51 shifted (-leafletwid,0);
1207 z62 = z52 shifted (0,leafletwid);
1208 z63 = z53 shifted (leafletwid,0);
1209 z64 = (w/2,h/2) shifted (0,leafletlen);
1210 z65 = (w/2,h/2) shifted (leafletlen,0);
1211 z66 = (w/2,h/2) shifted (0,-leafletlen);
1212 z67 = (w/2,h/2) shifted (-leafletlen,0);
1213 z68 = 0.5[z64,z65];
1214 z69 = 0.5[z65,z66];
1215 z70 = 0.5[z66,z67];
1216 z71 = 0.5[z67,z64];
1217 z72 = 0.5[z60,z61];
1218 z73 = 0.5[z61,z62];
1219 z74 = 0.5[z62,z63];
1220 z75 = 0.5[z63,z60];
1221 z76 = 0.2[z71,z75];
1222 z77 = point 0.4 along (z63{up}..{right}z60);
1223 z78 = point 0.6 along (z63{up}..{right}z60);
1224 z79 = 0.6[z63,z76];
1225 pickup thinpen;
1226 path innerbord; innerbord =
1227 z60{z64-z60}..z71..{z63-z67}z63{up}..{right}z60;
1228 draw innerbord withcolor #3;
1229 draw innerbord rotatedaround ((w/2,h/2),90) withcolor #3;
1230 draw innerbord rotatedaround ((w/2,h/2),180) withcolor #3;
1231 draw innerbord rotatedaround ((w/2,h/2),270) withcolor #3;
1232 label(btex {\drmdecinitfont at#2\unexpanded{\drminitfontcom} #5} etex,(w/2,h/2))
```

```
withcolor #4;
\end{mpost}
}
```

And that's the end. Thanks for reading, folks; please email me with any suggestions or improvements.

A The Secret History: Building DRM

HERE YOU CAN GET ALL the answers to questions about DRM that you didn't have and never asked. This appendix is essentially an exercise in self-gratification, to explain a few things about the fonts and why I made some of the choices that I did. As such, it'll probably be interesting to few, if any; but here it all is anyway.

A.1 About the Name

When I started this font, I was trying to ape an old-style Caslon specimen that I'd found on the Internet. (If you search for one, you'll doubtlessly find the one I was going for; it's littered all over the place.) You can still see certain traces of this, particularly in the long tail of the Q (there it is!), and in the serifs on the E and F. On the other hand, even at the very beginning, before the font had taken on a character of its own, I was doing a pretty poor job of imitating this other one. My serifs were fairly prominent, but only slightly bracketed; there was a pretty drastic distinction between thick and thin strokes; it had a vertical orientation. Before long, it was clear that I had a very different font.

So the name was "DRM," for "Day Roman Modern." But this didn't really accurately describe the font, and it didn't keep this meaning for long. (Maybe a few days; the original files were titled "dayroman," an even more inaccurate appellation.) I've since backronymed this to "Don's Revised Modern," which still isn't strictly correct, but it's pretty well ensconced at the moment. There are a limited number of descriptors with those initials, and I've grown pretty fond of those initials; it would be difficult for me to think of the font with any other name. But if somebody has a better backronym, I'd love to hear it.

A.2 Why METAFONT?

So why METAFONT? Isn't METAFONT horribly out of date, unconscionably producing nasty, decrepit bitmapped glyphs instead of shiny, futuristic outlines? Doesn't it somehow involve hatred of mom's apple pie, summertime barbecues, and the girl next door?

Well, in some ways METAFONT is certainly out of date. It's limited to eight bits (2^8 characters), for example, and that limit can't be circumvented by any trivial means. Due to the brilliance of its author, it has several similar limits which, while seeming arbitrary and capricious to us today, were absolutely

necessary for allowing METAFONT to run on the machines available at the time it was produced. There's really no denying this, and I'd be the last to try.

On the other hand, METAFONT is not out of date for the reasons most people who eloquently pronounce its obsolescence believe it is. The bitmapped glyph issue, for example; there really is nothing wrong with this. In fact, in some ways it's a benefit. Scaling is not really an issue, particularly in this age when METAFONT can be run automatically when TEX encounters a size that it doesn't already have on hand; we can easily acquire fonts of whatever size we need. And, much like METAFONT's eight-bit stricture, whatever memory benefits come from storing fonts as outlines rather than bitmaps is surely irrelevant in this day and age.

Bitmaps are beneficial in that they remind us that optical sizing is still important; outline fonts have made us lazy, preventing the development of real font families with many optical sizes. Too many amateurs (a term I use without derision, and proudly apply to myself) think they can avoid designing optical sizes because their outlines can be automatically scaled. This leads to poor results.

But most importantly, METAFONT makes writing parameter-based fonts easy. The **bold** and light versions of the DRM fonts, for example, were produced by modifying only a few parameters of the base DRM roman font; the actual letterforms remain the same. This is a powerful tool that assists greatly in the creation of *families* of fonts.

METAFONT also lends itself quite nicely to customization. As a command-line program, I can easily script it to produce proofs, or to produce real fonts for inclusion in test documents, or to produce font charts, or all of the above. For example, your author used scripts to compile proofs as well as working fonts to produce this documentation; it was relatively trivial to produce a script which would, inelegantly but quite effectively, output proof sheets and sample texts along with charts of each individual font, from simple roman text to quite complex math. Below is an example of the (rather messy, but functional) script I used to produce font charts and sample texts for the fonts:

```
#!/bin/bash
# +AMDG

re="^drm([m|b|bx|c|sym]*)([n|it|sl|sc]*)([0-9]{1,2})$";
ifmath="mmi";
ifmathsym="sy[0-9]";
font=$1;
fname="drm";
fenc="T1";
commands="\\sample\\bye";
if [[ $font =~ $ifmath ]]; then
 fenc="OML";
 commands="\\table\\math\\bye";
fi
if [[ $font =~ $ifmathsym ]]; then
```

```
 fenc="OMS";
 commands="\\table\\math\\bye";
fi
[[ $font =~ $re ]] && fweight="${BASH_REMATCH[1]}" &&
 fshape="${BASH_REMATCH[2]}" && fsize="${BASH_REMATCH[3]}";
if [[ $fshape == "" ]]; then
 fshape="m";
fi
if [[ $fweight == "" ]]; then
 fweight="n";
elif [[ $fweight == "sym" ]]; then
 fweight="n";
 fname="drmsym";
fi
#echo "fweight = $fweight; fshape = $fshape; fsize = $fsize";
nfssfontin=$(cat <<ENDFONTIN
$font
$commands
$fenc
$fname
$fweight
$fshape
$fsize
$commands
ENDFONTIN
)
echo $nfssfontin;

rm $1*pk; rm $1*gf;
mf "\mode=localfont; input $1";
gftopk $1.600gf $1.pk;
echo "$nfssfontin" | pdflatex nfssfont;
```

This little gem took a single argument, the name of the font that I wanted compiled; it then determined the appropriate parameters to hand over to `nfssfont`, including what type of sample was needed (text or math), and compiled it for me, which meant that with a single command (`./allcomp fontname`) I could get a complete chart of the font I was working on, along with a sample text to help judge kerning and general appearance. Doing the same with more "modern" font programs, particularly GUI ones, is doubtlessly more difficult.

Finally, pens. Pens are *endlessly* superior to defining points along outlines. The degree to which grokking and employing METAFONT's pen metaphor simplified the task of drawing these characters, particularly the more calligraphic varieties thereof, simply cannot be adequately expressed. Defining points along outlines and connecting them with zero-width lines did fine for *most* of the roman characters, but would have been painfully sluggish with, for example, the italic fonts.

Your author emphasized "most" above for good reason: while the points-

and-outlines approach worked quite effectively for the stately forms of roman characters, *modifying* those characters was sometimes much more difficult. Take, for example, the very basic different between "o" and "ø." Visually, of course, these are almost identical shapes, the latter simply having a line drawn through it; however, by outlines these shapes are so extremely different that drawing the latter would more easily be done from scratch than by a simple modification of the former. Using METAFONT's pen metaphor, though, the shape of "ø" could be drawn exactly as we would draw it on paper: by forming an "o," and then drawing a slash through it. And so your author accomplished it.

And though your author put off the development of italic until he'd become really proficient with METAFONT's pens, knowing that such intricate shapes as "*f*" and "*Q*" would be quite challenging with points-and-outlines, he was able to race through drawing the italics with ease, and wound up using METAFONT's pens much more frequently in the development of the remaining fonts than points-and-outlines, as drawing shapes with this metaphor is much more intuitive and easily visualized, at least to him, than the alternatives.

B The LaTeX Project Public License, v1.3c

The LaTeX Project Public License

LPPL Version 1.3c 2008-05-04

Copyright 1999, 2002–2008 LaTeX3 Project

> Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

Preamble

The LaTeX Project Public License (LPPL) is the primary license under which the LaTeX kernel and the base LaTeX packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is TeX-related (such as a LaTeX package), but it is written in such a way that you can use it even if your work is unrelated to TeX.

The section 'WHETHER AND HOW TO DISTRIBUTE WORKS UNDER THIS LICENSE', below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the LaTeX3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with

whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document 'cfgguide.tex' and 'modguide.tex' in the base LaTeX distribution for suggestions.

Definitions

In this license document the following terms are used:

Work Any work being distributed under this License.

Derived Work Any work that under any applicable law is derived from the Work.

Modification Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

Modify To apply any procedure that produces a Derived Work under any applicable law.

Distribution Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun's Network File System (NFS).

Compiled Work A version of the Work that has been processed into a form where it is directly usable on a computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

Current Maintainer A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the 'Copyright Holder' under any applicable law.

Base Interpreter A program or process that is normally needed for running or interpreting a part or the whole of the Work.

A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a 'LaTeX-Format' or in the case of files belonging to the 'LaTeX-format' a program implementing the 'TeX language'.

Conditions on Distribution and Modification

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.

2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.

3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.

4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.

5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.

6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.

   (1) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.

   (2) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the De-

rived Work and that contains a complete and accurate log of the changes.

(3) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.

(4) You distribute at least one of the following with the Derived Work:

1. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
2. Information that is sufficient to obtain a complete, unmodified copy of the Work.

7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.

8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.

9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.

10. (1) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.

(2) If a Derived Work is distributed under a different license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.

9. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.

10. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work 'as is', without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless required by applicable law or agreed to in writing will The Copyright Holder, or any author named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

Maintenance of The Work

The Work has the status 'author-maintained' if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that it is 'author-maintained'.

The Work has the status 'maintained' if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status 'maintained' to 'unmaintained' if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.

2. If this search is successful, then enquire whether the Work is still maintained.

   (1) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.

   (2) If the search is unsuccessful or no action to resume active maintenance is taken by the Current Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a LaTeX work, this could be done, for example, by posting to `comp.text.tex`.)

3. (1) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.

   (2) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.

4. If you make an 'intention announcement' as described in 22 above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.

5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 32 or 4, then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 22 above.

Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as 'you' in this section.

Choosing This License or Another License

If for any part of your work you want or need to use *distribution* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document 'modguide.tex' in the base LaTeX distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing LaTeX under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to LaTeX, the discussion in 'modguide.tex' may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2005 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
%   http://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
```

```
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status 'maintained'.
%
% The Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the 'Work' referring to the three files '`pig.dtx`', '`pig.ins`', and '`pig.sty`' (the last being generated from '`pig.dtx`' using '`pig.ins`'), the 'Base Interpreter' referring to any 'LaTeX-Format', and both 'Copyright Holder' and 'Current Maintainer' referring to the person 'M. Y. Name'.

If you do not want the Maintenance section of LPPL to apply to your Work, change 'maintained' above into 'author-maintained'. However, we recommend that you use 'maintained' as the Maintenance section was added in order to ensure that your Work remains useful to the community even when you can no longer maintain and support it yourself.

Derived Works That Are Not Replacements

Several clauses of the LPPL specify means to provide reliability and stability for the user community. They therefore concern themselves with the case that a Derived Work is intended to be used as a (compatible or incompatible) replacement of the original Work. If this is not the case (e.g., if a few lines of code are reused for a completely different task), then clauses 6b and 6d shall not apply.

Important Recommendations

Defining What Constitutes the Work The LPPL requires that distributions of the Work contain all the files of the Work. It is therefore important that you provide a way for the licensee to determine which files constitute the Work. This could, for example, be achieved by explicitly listing all the files of the Work near the copyright notice of each file or by using a line such as:

```
% This work consists of all files listed in manifest.txt.
```

in that place. In the absence of an unequivocal list it might be impossible for the licensee to determine what is considered by you to comprise the Work and, in such a case, the licensee would be entitled to make reasonable conjectures as to which files comprise the Work.

C The SIL Open Font License, v1.1

Preamble

The goals of the Open Font License (OFL) are to stimulate worldwide development of collaborative font projects, to support the font creation efforts of academic and linguistic communities, and to provide a free and open framework in which fonts may be shared and improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and redistributed freely as long as they are not sold by themselves. The fonts, including any derivative works, can be bundled, embedded, redistributed and/or sold with any software provided that any reserved names are not used by derivative works. The fonts and derivatives, however, cannot be released under any other type of license. The requirement for fonts to remain under this license does not apply to any document created using the fonts or their derivatives.

Definitions

"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting — in part or in whole — any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

Permission & Conditions

Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute,

and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1. Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2. Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3. No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4. The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5. The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to remain under this license does not apply to any document created using the Font Software.

Termination

This license becomes null and void if any of the above conditions are not met.

Disclaimer

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Z