1 Documentation

1.1 Installation

The installation consists in copying the .mf files from the mf subdirectory somewhere in Metafont’s search path and the .sty and .fd files from the latex subdirectory somewhere in \TeX{}’s search path.

Call TEXMF the base directory of your installation (on Unix this should be something like /usr/share/texmf or ~/texmf for a single user installation). Then copy the directories as follows:

- copy mf as TEXMF/fonts/source/public/cml1
- copy latex as TEXMF/tex/latex/cml1

If you want to install the fonts in Type1 format:

- copy type1 as TEXMF/fonts/type1/public/cml1
- copy tfm as TEXMF/fonts/tfm/public/cml1
- copy cml1.map as TEXMF/fonts/map/dvips/misc/cml1.map

You may have to update \TeX{}’s databases by running texhash or mktexlsr (this is for Kpathsea-based distributions like \TeX{}\TeX{}) or a similar command. You may also have to register the map file for the Type1 fonts using updmap, depending on your installation. Linux users using Debian or Ubuntu can avoid all this by using the package at http://iml.univ-mrs.fr/~beffara/soft/ (updated more often than \TeX{}Live).

1.2 Package loading

The package is loaded by simply saying \usepackage{cml1}. The variant of the font that should be used is chosen using the following package options:

- cm Computer Modern serif
- ss Computer Modern sans serif
- euler AMS Euler
- emu emulation mode (see section 1.4)
- auto automatic selection among the above (default, see section 1.4)
1.3 Symbols

The cmll font defines a handful of symbols useful in linear logic that were not defined in other fonts and packages.

Unary operators:

\[
\begin{align*}
! \oc & \quad ? \wn \\
\shpos & \quad \shneg \\
\shift
\end{align*}
\]

Binary operators:

\[
\begin{align*}
\parr & \quad \invamp \\
\with
\end{align*}
\]

Large operators:

\[
\begin{align*}
\bigparr & \quad \biginvamp \\
\bigwith & \quad \biginvamp
\end{align*}
\]

Binary relations:

\[
\begin{align*}
\coh & \quad \scoh \\
\incoh & \quad \sincoh \\
\Perp & \quad \simperp \\
\multimapinv & \quad \multimapboth \\
\multimap & \quad \multimapinv
\end{align*}
\]

Letter-like symbols:

\[
\begin{align*}
\Bot & \quad \simbot
\end{align*}
\]

The symbols !, ? and & are actually characters from the standard Computer Modern fonts declared with a new math code to get proper spacing. See the following examples:

\[
\begin{align*}
A \& B & A \& B \\
A = !B & A = !B \\
A = ?B & A = ?B
\end{align*}
\]

The names \parr and \invamp are synonyms, the names \bigparr and \biginvamp too. The macros \biginvampemu and \bigwithemu provide emulations for the \biginvamp and \bigwith symbols, built using the normal symbols at different sizes (using the relsize package). In emulation mode (i.e. with the package option emu) the names \biginvamp and \bigwith are synonyms for them.

1.4 Emulation and automatic selection

If you use a math font family other than Computer Modern or Euler, you may have an ampersand symbol that does not match the reversed ampersand from any variant of CMLL. In this case, the cmll package provides a “poor man” version of
the reversed ampersand and large ampersands, built using the standard \& symbol. Also provided are the “big” variants of \with and \invamp built using the normal version at a different size (using the relsize package). These versions can be used explicitly under the names \invampemu, \bigwithemu and \biginvampemu.

In the emulation mode, that is when the package is loaded with the emu option, these variants are used instead of the symbols provided by the font. Moreover, in emulation mode, any symbol that already exists is preserved by cmll, otherwise all symbols are redefined. This is useful for instance with txfonts, which already provides \invamp and \multimap both. Note that this feature works only if the package cmll is loaded after loading any package that might define one of the symbols provided by CMLL.

The automatic selection works as follows:

- if euler is loaded, use the Euler variant,
- if txfonts or pxfonts is loaded, use the emulation mode,
- if the default font is cmss, use the ss variant,
- otherwise use the cm variant.

1.5 History

2010-04-20 New symbols: \multimapinv, \nmultimap, \nmultimapinv, \nmultimapboth.

2009-01-23 New symbols: \multimapboth, \Perp, \Bot, \simperp, \simbot.

Synonyms \invamp and \biginvamp added for compatibility. New package option emu. Various fixes and code improvements.

2006-02-22 First public release.

2 Files

2.1 This document

1 (+driver)
2 \documentclass{ltxdoc}
3 \usepackage[array,cmll]
4 \newenvironment{symbols}[1]{%
5 \par%
6 \def\dosymbol##1{%<leavevmodeobox to .5\textwidth%
7 \kern.25\textwidth \hbox to 2em{\hss##1\hfil}%
8 \texttt{\string##1}\hss%
9 \flushleft%
10 #1\strut\}}{%endflushleft}
11 \begin{document}
12 \DocInput{cmll.dtx}
13 \end{document}
14 (/driver)
2.2 Font definitions

The font definition file is deduced from the ones for Computer Modern. We provide an NFSS entry named \texttt{cmllr} in medium and bold extended versions.

\begin{verbatim}
\DeclareFontShape{U}{cmllr}{m}{n}{<5><6><7><8><9>gen*cmllr% <10><10.95>cmllr10% <12><14.4>cmllr12% <17.28->cmllr17%}
\end{verbatim}

\begin{verbatim}
\DeclareFontShape{U}{cmllr}{bx}{n}{<5><6><7><8><9>gen*cmllbx% <10><10.95>cmllbx10% <12->cmllbx12%}
\end{verbatim}

The following is a definition for the sans-serif version, named \texttt{cmllss}.

\begin{verbatim}
\DeclareFontShape{U}{cmllss}{m}{n}{<8>cmllss8% <9>cmllss9% <10>cmllss10 <12><14.4>cmllss12% <17.28->cmllss17%}
\end{verbatim}

\begin{verbatim}
\DeclareFontShape{U}{cmllss}{bx}{n}{<->cmllssbx10}{
\end{verbatim}

The following is a definition for the Euler-style version, named \texttt{eull}.

\begin{verbatim}
\DeclareFontShape{U}{eull}{m}{n}{<5><6><7><8><9>gen*eullr% <10->eullr10}{
\end{verbatim}

\begin{verbatim}
\DeclareFontShape{U}{eull}{bx}{n}{<5><6><7><8><9>gen*eullbx% <10->eullbx10}{
\end{verbatim}

2.3 The package

\begin{verbatim}
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{cmll}[2010/04/20 Linear Logic symbols for Computer Modern]
\end{verbatim}

With the package option \texttt{emu}, symbols already defined are used and \texttt{\invamp} (if undefined) is made by rotation using an ampersand. This requires the \texttt{graphicx} package.
The font is declared as a symbol font named llsymbols, in normal and bold versions. We provide package options to switch between the standard, sans-serif and Euler-style variants.

The following code is used to detect which family should be used. Euler is detected if its package is loaded, emulation is activated for txfonts and pxfonts, sans-serif is detected by looking at the default font family name.

The following macros are used to set up the font families and symbols from other fonts.

Here is the sans-serif variant.

And here is the Euler variant.
Finally we activate the proper variant.

In some cases it is preferable to emulate the \invamp by rotating the ampersand symbol. Here is a robust definition of this rotation:

\begin{verbatim}
\newcommand\invampemu{\mathbin{\mathchoice\{otatebox[origin=c]{180}{$\&$}\}\}}
\end{verbatim}

We may also want to emulate the big versions of the ampersand. In this case we change their text size appropriately for each style, using the relsize package. This version is reasonable at normal size and becomes approximative when math is composed in small or large sizes.

\begin{verbatim}
\newcommand\bigwithemu{\mathop{\mathchoice\{\vcenter{\hbox{\relsize{+4}$\&$}}\}}}
\end{verbatim}

For the large inversed ampersand, we call the \invamp macro, which is supposed to be defined (either as a proper character or as an emulation as above). When it is emulated, this makes two nested \mathchoices, which is not very efficient.

\begin{verbatim}
\newcommand\biginvampemu{\mathop{\mathchoice\{\vcenter{\hbox{\relsize{+4}$\invamp$}}\}}}
\end{verbatim}

The actual commands \invamp, \bigwith and \biginvamp can be defined in various ways depending on the setup.

\begin{verbatim}
\@ifundefined{invamp}{\RequirePackage{graphicx}\let\invamp=\invampemu}{%}
\end{verbatim}

Already existing symbols are preserved in emulation mode.

\begin{verbatim}
\@ifundefined{\symbol#1#2#3#4}{%\expandafter\DeclareMathSymbol\expandafter{#1}{#2}{#3}{#4}}%
\end{verbatim}
The new symbol definitions are the same for all variants.
\cmll@declare@symbol{invamp}{\mathbin}{llsymbols}{0}
\let\parr=\invamp
\cmll@declare@symbol{shpos}{\mathord}{llsymbols}{1}
\cmll@declare@symbol{shneg}{\mathord}{llsymbols}{2}
\cmll@declare@symbol{shift}{\mathord}{llsymbols}{3}
\cmll@declare@symbol{coh}{\mathrel}{llsymbols}{4}
\cmll@declare@symbol{scoh}{\mathrel}{llsymbols}{5}
\cmll@declare@symbol{shift}{\mathord}{llsymbols}{3}
\cmll@declare@symbol{incoh}{\mathrel}{llsymbols}{6}
\cmll@declare@symbol{sincoh}{\mathrel}{llsymbols}{7}
\cmll@declare@symbol{bigwith}{\mathop}{llsymbols}{8}
\cmll@declare@symbol{biginvamp}{\mathop}{llsymbols}{9}
\let\bigparr=\biginvamp
\cmll@declare@symbol{multimapboth}{\mathrel}{llsymbols}{12}
\cmll@declare@symbol{Bot}{\mathord}{llsymbols}{13}
\cmll@declare@symbol{Perp}{\mathrel}{llsymbols}{13}
\cmll@declare@symbol{simbot}{\mathord}{llsymbols}{14}
\cmll@declare@symbol{simperp}{\mathrel}{llsymbols}{14}
\cmll@declare@symbol{multimapinv}{\mathrel}{llsymbols}{15}
\cmll@declare@symbol{nmultimap}{\mathrel}{llsymbols}{16}
\cmll@declare@symbol{nmultimapinv}{\mathrel}{llsymbols}{17}
\cmll@declare@symbol{nmultimapboth}{\mathrel}{llsymbols}{18}
\endinput