The \texttt{nccrules} package\footnote{This file has version number v1.0, last revised 2005/05/13.}

Alexander I. Rozhenko
rozhenko@oapmg.sscru
2005/05/13

This package implements \texttt{\dashrule} and \texttt{\dashrulefill} commands, that simplify composing of dashed lines and dashed multilines. Two kinds of footnote rule generation commands are also introduced: \texttt{\newfootnoterule} creates a footnote rule with an arbitrary contents and \texttt{\newfootnotedashrule} creates a footnote rule based on dash rule.

\tableofcontents

1 User Interface

1.1 Dash Rules

\begin{verbatim}
dashrule[(raise)]{⟨h-pattern⟩}{⟨v-pattern⟩}
\end{verbatim}

prepares a dash rule. Its syntax is quite similar to the \texttt{\rule} command except that \texttt{⟨h-pattern⟩} and \texttt{⟨v-pattern⟩} can contain a list of sizes delimited with spaces. List sizes are interpreted as follows: size, space, size, space, etc. In other words, every odd size is a size of rule part and every even size is a space between neighbour parts. If the last size in the list is even (means a space), its space value is divided by two and is added before the first rule part and after the last rule part. Units in sizes can be omitted. In this case, pt-units are supposed. Examples:
The command
\dashrulefill[\langle\text{raise}\rangle][\langle\text{leader-type}\rangle][\langle\text{h-pattern}\rangle][\langle\text{v-pattern}\rangle]
fills a free space with a dash rule. The rule is composed from the 1st, 3rd, and 4th arguments of the command and is repeated as more times as necessary to fill a free horizontal space. The rest of space after filling is stored depending on \langle\text{leader-type}\rangle used. Default leader type is “aligned leaders”: every rule is aligned to the multiple of its width counted from the left boundary of filled box. Other cases are: c means centered leaders (all extra spaces are collected at both ends of fill area), x means distributed leaders (extra spaces are uniformly distributed between every dash rule), and s means stretched leaders (the space in h-pattern going after the last rule size is considered as stretchable space and rules are stretched to the whole width of the fill area).

Example:

\noindent x\dashrulefill{2 2}{2}\mbox{}\
xxxxx\dashrulefill[0.4ex]{2 2}{2}\mbox{}\
\mbox{}\dashrulefill[0.4ex][c]{4 4}{2}\mbox{}\
\mbox{}\dashrulefill[[4 4]{2}\mbox{}\
\mbox{}\dashrulefill[.4ex][x]{4 4}{2}\mbox{}\
\mbox{}\dashrulefill[.4ex][s]{4 4}{2}\mbox{}\
\mbox{}\dashrulefill[.4ex][s]{4 2 1 2 4}{1}\mbox{}\
\mbox{}\dashrulefill{5 3 1 3 5}{1 1 0.4 1 1}\mbox{}\
\mbox{}\dashrulefill[0mm][c]{5 3 1 3 5}{1 1 0.4 1 1}\mbox{}\hrulefill

It produces the following:

x .................................................................
xxxxx ........................................................................

\dashrulefill{10mm}{1pt}x
\dashrulefill{3 3 5mm}{.4}x
\dashrulefill{3 2 3}x
\dashrulefill{2 2}x
\dashrulefill{5 3 1 3 10}{1 1 0.4 1 1}x
\dashrulefill[.4ex]{2 2}x
\dashrulefill[.4ex][c]{4 4}x
\dashrulefill[.4ex][x]{4 4}x
\dashrulefill[.4ex][s]{4 4}x
\dashrulefill[.4ex][s]{4 2 1 2 4}x
\dashrulefill{5 3 1 3 5}x
\dashrulefill{5 3 1 3 5}x
\hrulefill

\hrulefill

2
1.2 Custom Footnote Rules

In the `manyfoot` package, you can separate footnote levels with custom footnote rules. In this package, we provide two ways for creation such rules. All footnote rules should follow the following contract: they must be prepared as \hrule or \hbox; the total height of rule or box must be compensated with kerns. Usually a negative kern is inserted before a footnote rule and a nonnegative kern is inserted after the rule. The sum of both kerns and of total height of rule must be equal to zero. We follow this contract in footnote rules created with this package.

In the `manyfoot` package, every footnote rule must have a name \footnoterule. So, we need to specify the prefix only when a new footnote rule is constructed. Next that we need to specify is a rule width. If width is omitted, the default width of 0.4\columnwidth is used. The footnote rule generation commands described below are used in the preamble only.

\newfootnoterule

produces a custom user-defined rule. The \footnoterule must be stretchable to the given width. The \mboxfill command from the `mboxfill` package is useful in it.

Examples:

\newfootnoterule{starred}{\mboxfill[1.5\width][s]{*$*$}}
\newfootnoterule{dotted}{\mboxfill[2\width][a]{.}}

The \starredfootnoterule and \dottedfootnoterule commands are created here. They produce the following rules:

\begin{itemize}
  \item \starredfootnoterule
  \item \dottedfootnoterule
\end{itemize}

\newfootnotedashrule

produces a dash footnote rule. The special case, when \h-pattern is empty means the solid rule. Examples:

\newfootnotedashrule{double}{.4 1 .4}
\newfootnotedashrule{dashed}{3 3}{.4}
\newfootnotedashrule{dotdashed}{4 2 1 2 4 -4}{.6}
\newfootnotedashrule{ddotdashed}{6 2 1 2 1 2 6 -6}{.6}

The following rules are created here: \doublefootnoterule, \dashedfootnoterule, \dotdashedfootnoterule, and \ddotdashedfootnoterule. They produce the following rules:
2 The Implementation

The `mboxfill` package is required here:

```latex
\begin{verbatim}
\makeatletter
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
(\makeatother)
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
```

This command is a composition of the `mboxfill` and `\dashrule` commands. A trick is applied here: we decrease dash rule width on the value of last space in it and then apply `mboxfill`.

```latex
\begin{verbatim}
\makeatletter
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
(\makeatother)
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
```

`\NCC@composedash` parses the pattern and calls the action when a rule size is parsed. At this point, the `\@tempdimb` contains a size parsed and `\@tempdimc` contains the previous space. In `\@tempdimc`, the whole rule size is calculated.

```latex
\begin{verbatim}
\makeatletter
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
\begin{verbatim}
(\makeatother)
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
\end{verbatim}
```

Pattern parser:
\def\NCC@parsedash#1 \@nil#3{%  
\if#1/\else % Empty arg. ignored  
\ifx#1!\else % Exclamation mark ends the list  
\@defaultunits\@tempdima#1pt\relax\@nil  
#3\NCC@parsedash#2\@nil(#3)#3%  
\fi  
\fi  
}\NCC@vdash\NCC@vdash{⟨h-pattern⟩} is applied when a vertical dash is composed. \@tempdimb contains the required skip, \@tempdima contains the rule height.

\def\NCC@vdash#1{%  
\setbox\@tempboxa\vbox{%  
\unvbox\@tempboxa \vskip\@tempdimb  
\setbox\@tempboxa\hbox{}%  
\@tempskipa\@tempdima % Save rule height in \@tempskipa  
\NCC@composedash{\NCC@hdash}{#1}%  
\hb@xt\@tempdimc{\hss\unhbox\@tempboxa\hss}%  
}\NCC@fnoterule This is the footnote rule producing command. As usual, two kerns must be inserted surround the rule: a negative kern before the rule and a nonnegative kern after the rule. The sum of kern values and of the total rule height must vanish. Kerns are calculated in such a way to vertically center the rule relative to 2.8pt distance top to the current position. If the total height of rule is greater than 5.6pt, the rule is moved up on the total rule height (the kern after the rule is zero in this case).

\def\NCC@fnoterule#1#2{%  
\setlength\@tempdima{#1}%  
\setbox\@tempboxa\hb@xt@\@tempdima{#2}%  
\@tempdima\ht\@tempboxa \advance\@tempdima\dp\@tempboxa  
\@tempdimb 2.8\p@ \@tempdimc .5\@tempdima  
\ifdim\@tempdimc<\@tempdimb \@tempdimb \@tempdimb\@tempdimc \fi  
\advance\@tempdimb\@tempdimc  
\kern-\@tempdimb  
\box\@tempboxa  
\advance\@tempdimb -\@tempdima  
\NCC@fnoterule{⟨width⟩}{⟨rule code⟩}
\NCC@fnotedashrule \NCC@fnotedashrule{⟨width⟩}{⟨h-pattern⟩}{⟨v-pattern⟩}

 Produces the rule based on \dashrulefill. A special case of empty ⟨h-pattern⟩
 means the solid (maybe multi-line) rule.

 \def\NCC@fnotedashrule#1#2#3{%
  \NCC@fnoterule{#1}{\def\@tempa{#2}%
  \ifx\@tempa\@empty
  \@tempskipb\@tempdima \dashrule{\@tempskipb}{#3}%
  \else
  \dashrulefill[\z@][s]{#2}{#3}%
  \fi
  %
  %}
%

\NCC@fnotecreate \NCC@fnotecreate{⟨prefix⟩}{⟨body⟩}

 Creates a footnote rule whose name is composed from the given ⟨prefix⟩ and
 footnoterule. For example if the prefix is dashed, the \dashedfootnoterule
 command will be created.

 \def\NCC@fnotecreate#1{%
  \edef\@tempa{\noexpand\newcommand*%
    \expandafter\noexpand\csname #1footnoterule\endcsname}%
  \@tempa
  %
  %
  \@onlypreamble\NCC@fnotecreate

\NCC@fnoteprepare \NCC@fnoteprepare{⟨driver⟩}{⟨prefix⟩}{⟨width⟩}

 Prepares a footnote rule command. The ⟨driver⟩ is a command that will be called
 for collecting rule code (it gets more arguments from the input). When a driver
 prepares the code, it calls the \NCC@fnotecreate command. If ⟨width⟩ is omitted,
 the standard width of 0.4\textwidth is used.

 \def\NCC@fnoteprepare#1#2{%
  \ifnextchar[#{#1}{#2}[.4\textwidth]%

  %
  %
  \@onlypreamble\NCC@fnoteprepare

 Footnote rule creation driver has the following syntax:

 ⟨driver⟩{⟨prefix⟩}{⟨width⟩}{⟨more arguments⟩}

\newfootnoterule \newfootnoterule{⟨prefix⟩}{⟨width⟩}{⟨rule code⟩}

 \newcommand\newfootnoterule{\NCC@fnoteprepare{\NCC@fnotedriver}}

 \def\NCC@fnotedriver#1#2#3{%
  \NCC@fnotecreate{#1}{\NCC@fnoterule{#2}{#3}}%
  %
  %
  \@onlypreamble\newfootnoterule

  %
  %
  \@onlypreamble\NCC@fnotedriver