NTG Document Class \texttt{brief} for \LaTeX\ version 2e

Victor Eijkhout    Johannes Braams

February 21, 2020

Contents

1 Introduction 2

2 Initial Code 2
  2.1 Setting Paper Sizes  . . . . . . . . . . . . . . . . . . . . . . 3
  2.2 Choosing the type size  . . . . . . . . . . . . . . . . . . . . . 4
  2.3 Two-side or one-side printing  . . . . . . . . . . . . . . . . 4
  2.4 Draft option  . . . . . . . . . . . . . . . . . . . . . . . . . . 4
  2.5 Equation numbering on the left  . . . . . . . . . . . . . . . . 4
  2.6 Flush left displays  . . . . . . . . . . . . . . . . . . . . . . 4
  2.7 Typewriter alignment  . . . . . . . . . . . . . . . . . . . . . 5
  2.8 Folding lines  . . . . . . . . . . . . . . . . . . . . . . . . . . 5
  2.9 Address placement  . . . . . . . . . . . . . . . . . . . . . . 5
  2.10 Support for different languages  . . . . . . . . . . . . . . . 5

3 Executing Options 5

4 Loading Packages 8

5 Document Layout 8
  5.1 Fonts  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
  5.2 Paragraphing  . . . . . . . . . . . . . . . . . . . . . . . . . 9
  5.3 Page Layout  . . . . . . . . . . . . . . . . . . . . . . . . . . 10
    5.3.1 Vertical spacing  . . . . . . . . . . . . . . . . . . . . . . 10
    5.3.2 The dimension of text  . . . . . . . . . . . . . . . . . . 10
    5.3.3 Margins  . . . . . . . . . . . . . . . . . . . . . . . . . . 10
    5.3.4 The address field  . . . . . . . . . . . . . . . . . . . . . 11
    5.3.5 Changing head and text heights  . . . . . . . . . . . . . 11
    5.3.6 Information in the foot  . . . . . . . . . . . . . . . . . 12
    5.3.7 Footnotes  . . . . . . . . . . . . . . . . . . . . . . . . . . 12
  5.4 Page Styles  . . . . . . . . . . . . . . . . . . . . . . . . . . 12
    5.4.1 Marking conventions  . . . . . . . . . . . . . . . . . . . 12
    5.4.2 Defining the page styles  . . . . . . . . . . . . . . . . . 13
1 Introduction

This file contains the document class brief that was made available by Working Group 13 of the NTG (Nederlandstalige \TeX Gebruikersgroep). It defines more commands than the standard document class letter, but a letter made with the letter document class is still processable with this document class.

2 Initial Code

In this part we define a few commands that are used later on.
\@ptsize This control sequence is used to store the second digit of the pointsize we are typesetting in. So, normally, it’s value is one of 0, 1 or 2.
1 \(^{(+\text{brief})}\)
2 \texttt{newcommand}\@ptsize{}

\@typhulp This switch is used to decide whether or not to put a small line on the paper that is used to align the paper in a typewriter.
3 \texttt{newif}@typhulp

\@streepjes A switch to indicate if the ‘folding lines’ should be printed
4 \texttt{newif}@streepjes

\@adresrechts This switch indicates if the addressing information is to be set on the left or on the right side of the letter.
5 \texttt{newif}@adresrechts

\@elfinch A switch to remember whether we are using A4 or letter paper. (possibly obsolete)
6 \texttt{newif}@elfinch

2.1 Setting Paper Sizes
The variables \texttt{\paperwidth} and \texttt{\paperheight} should reflect the physical paper size after trimming. For desk printer output this is usually the real paper size since there is no post-processing.
7 \texttt{\DeclareOption\{a4paper\}}
8 \{\setlength\paperheight {297mm}\%
9 \setlength\paperwidth {210mm}\@elfinchfalse\}
10 \texttt{\DeclareOption\{a5paper\}}
11 \{\ClassWarning{brief}{Paper size A5 not supported, using A4}\%
12 \setlength\paperheight {297mm}\%
13 \setlength\paperwidth {210mm}\@elfinchfalse\}
14 \texttt{\DeclareOption\{b5paper\}}
15 \{\ClassWarning{brief}{Paper size B5 not supported, using A4}\%
16 \setlength\paperheight {297mm}\%
17 \setlength\paperwidth {210mm}\@elfinchfalse\}
18 \texttt{\DeclareOption\{letterpaper\}}
19 \{\setlength\paperheight {11in}\%
20 \setlength\paperwidth {8.5in}\@elfinchtrue\}
21 \texttt{\DeclareOption\{USletter\}}
22 \{\setlength\paperheight {11in}\%
23 \setlength\paperwidth {8.5in}\@elfinchtrue\}
24 \texttt{\DeclareOption\{legalpaper\}}
25 \{\ClassWarning{brief}{\%}
26 \setlength\paperheight {14in}\%
27 \setlength\paperwidth {8.5in}\@elfinchtrue\}
28 \texttt{\ DeclareOption\{executivepaper\}}
29 \{\ClassWarning{brief}\%}
30 \{\ClassWarning{brief}\%}
2.2 Choosing the type size

The type size options are handled by defining \@ptsize to contain the last digit of the size in question and branching on \ifcase statements. This is done for historical reasons to stay compatible with other packages that use the \@ptsize variable to select special actions. It makes the declarations of size options less than 10pt difficult, although one can probably use 9 and 8 assuming that a class won't define both 8pt and 18pt options.

\DeclareOption{10pt}{\renewcommand*{\@ptsize}{0}}
\DeclareOption{11pt}{\renewcommand*{\@ptsize}{1}}
\DeclareOption{12pt}{\renewcommand*{\@ptsize}{2}}

2.3 Two-side or one-side printing

Two-sided printing was not supported in the \LaTeX\ 2.09 version of this document-class.
\if@compatibility
\DeclareOption{twoside}{\@latexerr{No 'twoside' layout for letters}\@eha}
\else
\DeclareOption{twoside}{\@twosidetrue \@mparswitchtrue}
\fi
\DeclareOption{oneside}{\@twosidefalse \@mparswitchfalse}

2.4 Draft option

If the user requests draft we show any overfull boxes. We could probably add some more interesting stuff to this option.
\DeclareOption{draft}{\setlength{\overfullrule}{5pt}}
\DeclareOption{final}{\setlength{\overfullrule}{0pt}}

2.5 Equation numbering on the left

The option leqno can be used to get the equation numbers on the left side of the equation.
\DeclareOption{leqno}{\input{leqno.clo}}

2.6 Flush left displays

The option fleqn redefines the displayed math environments in such a way that they come out flush left, with an indentation of \mathindent from the prevailing left margin.
\DeclareOption{fleqn}{\input{fleqn.clo}}
2.7 Typewriter alignment
\DeclareOption{typhulp}{\@typhulptrue}
\DeclareOption{geentyphulp}{\@typhulpfalse}

2.8 Folding lines
It is possible to print ‘folding lines’ on the far right side of the paper.
\DeclareOption{streepjes}{\@streepjestrue}
\DeclareOption{geenstreepjes}{\@streepjesfalse}

2.9 Address placement
The address information can be put either on the left or on the right side of the
letter
\DeclareOption{adreslinks}{\@adresrechtsfalse}
\DeclareOption{adresrechts}{\@adresrechtstrue}

2.10 Support for different languages
In the original document style \texttt{brief} the options to support the various languages
were all dutch words. To be compatible with both the old version of the document
class and with the recommended set of language options we have at least two
options for each language.
First Dutch,
\DeclareOption{nederlands}{\AtEndOfClass{dutchbrief}}
\DeclareOption{dutch}{\AtEndOfClass{dutchbrief}}

then British English,
\DeclareOption{engels}{\AtEndOfClass{englishbrief}}
\DeclareOption{english}{\AtEndOfClass{englishbrief}}
American English,
\DeclareOption{USengels}{\AtEndOfClass{americanbrief}}
\DeclareOption{american}{\AtEndOfClass{americanbrief}}

German
\DeclareOption{duits}{\AtEndOfClass{germanbrief}}
\DeclareOption{german}{\AtEndOfClass{germanbrief}}

and finally french.
\DeclareOption{frans}{\AtEndOfClass{frenchbrief}}
\DeclareOption{french}{\AtEndOfClass{frenchbrief}}
\DeclareOption{francais}{\AtEndOfClass{frenchbrief}}

3 Executing Options
Here we execute the default options to initialize certain variables.
\ExecuteOptions{a4paper,11pt,oneside,onecolumn,final,\%
\ExecuteOptions{a4paper,11pt,oneside,onecolumn,final,%

The \texttt{\ProcessOptions} command causes the execution of the code for every option \texttt{FOO} which is declared and for which the user typed the \texttt{FOO} option in his \texttt{\documentclass} command. For every option \texttt{BAR} he typed, which is not declared, the option is assumed to be a global option. All options will be passed as document options to any \texttt{\usepackage} command in the document preamble.

Now that all the options have been executed we can define the user-level size changing commands. Their definition depends on which of the 10pt, 11pt or 12pt options was specified.

The user level command for the main size is \texttt{\normalsize}. Internally \TeX{} uses \texttt{\@normalsize} when it refers to the main size. \texttt{\@normalsize} will be defined to work like \texttt{\normalsize} if the latter is redefined from its default definition (that just issues an error message). Otherwise \texttt{\normalsize} simply selects a 10pt/12pt size.

The \texttt{\normalsize} macro also sets new values for \texttt{\abovedisplayskip}, \texttt{\abovedisplayshortskip} and \texttt{\belowdisplayskip}:

\begin{verbatim}
\normalsize
\@normalsize
\end{verbatim}

\begin{verbatim}
\ifcase\@ptsize
\renewcommand*{\normalsize}{\@setfontsize{\normalsize}{\@xpt}{\@xiipt}
\abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
\abovedisplayshortskip \z@ \@plus3\p@
\belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
\belowdisplayskip \abovedisplayskip
\let\@listi\@listI}
\or
\renewcommand*{\normalsize}{\@setfontsize{\normalsize}{\@xipt}{13.6}\p@
\abovedisplayskip 11\p@ \@plus3\p@ \@minus6\p@
\abovedisplayshortskip \z@ \@plus3\p@
\belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
\belowdisplayskip \abovedisplayskip
\let\@listi\@listI}
\or
\renewcommand*{\normalsize}{\@setfontsize{\normalsize}{\@xiipt}{15}\p@
\abovedisplayskip 12\p@ \@plus3\p@ \@minus7\p@
\abovedisplayshortskip \z@ \@plus3\p@
\belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
\belowdisplayskip \abovedisplayskip
\let\@listi\@listI}
\fi
\end{verbatim}

Make \texttt{\@normalsize} a synonym for \texttt{\normalsize}.

\begin{verbatim}
\let\@normalsize\normalsize
\end{verbatim}
We initially choose the normalsize font.

\normalsize

We use \MakeRobust instead of \DeclareRobustCommand above to avoid a log entry for the redefinition. But if we are running in a rollback situation (prior to 2015) we don’t touch it.

\ifx\MakeRobust\@undefined \else
\fi

This is similar to \normalsize.

\ifcase\@ptsize
\else
\fi

This is similar to \normalsize.

\ifcase\@ptsize
\else
\fi
These are all much simpler than the previous macros, they just select a new 
font size, but leave the parameters for displays and lists alone.

This class file does not load additional packages.

In this section we are finally dealing with the nasty typographical details.

We use two fixed fonts in these letters.
5.2 Paragraphing

\lineskip These parameters control \TeX's behaviour when two lines tend to come too close together.
\normallineskip
\baselinestretch This is used as a multiplier for \baselineskip. The default is to \textit{not} stretch the baselines.
\setlength{\lineskip}{1\p@}
\setlength{\normallineskip}{1\p@}
\renewcommand{\baselinestretch}{}
\parskip \parindent gives extra vertical space between paragraphs and \parindent is the width of the paragraph indentation. Letters are typeset without paragraph indentation.
\setlength{\parskip}{0.7em \@plus .3em \@minus .2em}
\setlength{\parindent}{0\p@}
\@lowpenalty \@medpenalty \@highpenalty The commands \texttt{\nopagebreak} and \texttt{\nolinebreak} put in penalties to discourage these breaks at the point they are put in. They use \@lowpenalty, \@medpenalty or \@highpenalty, dependant on their argument.
\setlength{\@lowpenalty}{51}
\setlength{\@medpenalty}{151}
\setlength{\@highpenalty}{301}
\clubpenalty \widowpenalty These penalties are use to discourage club and widow lines. Because we use their default values we only show them here, commented out.
\setlength{\clubpenalty}{150}
\setlength{\widowpenalty}{150}
\displaywidowpenalty \predisplaypenalty \postdisplaypenalty Discourage (but not so much) widows in front of a math display and forbidding breaking directly in front of a display. Allow break after a display without a penalty.
\setlength{\displaywidowpenalty}{50}
\setlength{\predisplaypenalty}{10000}
\setlength{\postdisplaypenalty}{0}
\interlinepenalty Allow the breaking of a page in the middle of a paragraph.
\setlength{\interlinepenalty}{0}
\brokenpenalty We allow the breaking of a page after a hyphenated line.
\setlength{\brokenpenalty}{0}
5.3 Page Layout

All margin dimensions are measured from a point one inch from the top and lefthand side of the page.

5.3.1 Vertical spacing

\headheight The \headheight is the height of the box that will contain the running head. The \headsep is the distance between the bottom of the running head and the top of the text. \topskip is the \baselineskip for the first line on a page.

\footskip The distance from the baseline of the box which contains the running footer to the baseline of last line of text is controlled by the \footskip. Bottom of page:

\maxdepth The \text height primitive register \maxdepth has a function that is similar to that of \topskip. The register \@maxdepth should always contain a copy of \maxdepth. In both plain \TeX and \LaTeX 2.09 \maxdepth had a fixed value of 4pt; in native \LaTeX2e mode we let the value depend on the typesize. We set it so that \maxdepth + \topskip = typesize \times 1.5. As it happens, in these classes \topskip is equal to the typesize, therefore we set \maxdepth to half the value of \topskip.

5.3.2 The dimension of text

\textwidth The dimensions of the text are fixed; they are defined in the NEN norm which this class implements.

\textheight \rightskip \@rightskip

\oddsidemargin \evensidemargin \marginparwidth

5.3.3 Margins
The horizontal space between the main text and marginal notes is determined by \marginparsep, the minimum vertical separation between two marginal notes is controlled by \marginparpush.

\topmargin is the distance between the top of ‘the printable area’ –which is 1 inch below the top of the paper– and the top of the box which contains the running head.

5.3.4 The address field

The address information has to be put on a specific place.

5.3.5 Changing head and text heights

This class has a much higher head on the first page of a letter than on subsequent pages.
5.3.6 Information in the foot

We also reserve some space at the bottom of the paper to print some information about the sender of the letter.

\footsep The distance between the text and this foot information
\newdimen\footsep
\setlength\footsep{15mm}

5.3.7 Footnotes

The distance between the text and this foot information
\setlength\footsep{15mm}

5.4 Page Styles

The page style foo is defined by defining the command \ps@foo. This command should make only local definitions. There should be no stray spaces in the definition, since they could lead to mysterious extra spaces in the output (well, that’s something that should be always avoided).

\@evenhead \@oddhead \@evenfoot \@oddfoot

The \ps@... command defines the macros \@oddhead, \@oddfoot, \@evenhead, and \@evenfoot to define the running heads and feet—e.g., \@oddhead is the macro to produce the contents of the heading box for odd-numbered pages. It is called inside an \hbox of width \textwidth.

5.4.1 Marking conventions

To make headings determined by the sectioning commands, the page style defines the commands \chaptermark, \sectionmark, ..., where \chaptermark{\(TEXT\)} is called by \chapter to set a mark, and so on.

The \...mark commands and the \...head macros are defined with the help of the following macros. (All the \...mark commands should be initialized to no-ops.)

\LaTeX\ extends \TeX's \mark facility by producing two kinds of marks, a ‘left’ and a ‘right’ mark, using the following commands:

\markboth{(LEFT)}{(RIGHT)}: Adds both marks.
\markright{(RIGHT)}: Adds a ‘right’ mark.
\leftmark: Used in the \@oddhead, \@oddfoot, \@evenhead or \@evenfoot macros, it gets the current ‘left’ mark. \leftmark works like \TeX's \botmark command.
\rightmark: Used in the \@oddhead, \@oddfoot, \@evenhead or \@evenfoot macros, it gets the current ‘right’ mark. \rightmark works like \TeX’s \firstmark command.

The marking commands work reasonably well for right marks ‘numbered within’ left marks—e.g., the left mark is changed by a \chapter command and the right mark is changed by a \section command. However, it does produce somewhat anomalous results if two \markboth’s occur on the same page.

Commands like \tableofcontents that should set the marks in some page styles use a \markboth command, which is \let by the pagestyle command (\@mkboth) to \markboth for setting the heading or to \@gobbletwo to do nothing.

5.4.2 Defining the page styles

The pagestyles empty and plain are defined in the \LaTeX kernel (ltpage.dtx), but these definitions are changed to a simpler version for this document class.

\ps@headings

The definition of the page style headings has to be different for two sided printing than it is for one sided printing.

\if@twoside
\def\ps@headings{%

The running feet contain some information about the sender of the letter. The feet are the same for even and odd pages.

\def\@oddfoot{\voetregel\hss}%
\let\@evenfoot\@oddfoot

The running head contains some information about this letter. The head is the same for even and odd pages.

\def\@oddhead{%
\vbox to \@otherheadheight
{\vervolghoofd\vfil
  \if@streepjes\streepjes{\@firstheadheight}\fi}\hss}
\let\@evenhead\@oddhead

For one sided printing we don’t need to define \@evenhead so the definition is somewhat simpler.
\else
\def\ps@headings{%
\def\@oddfoot{\voetregel\hss}%
\def\@oddhead{%
\vbox to \@otherheadheight
{\vervolghoofd\vfil
  \if@streepjes\streepjes{\@otherheadheight}\fi}\hss}
\fi
\fi

\ps@firstpage

On the first page the head contains much more than on other pages, therefore the height of the head and text need to be adapted.
The definition of the page style \texttt{empty} is simple: No running head or foot at all.

\begin{verbatim}
\ps@empty
\def\ps@empty{\let\@oddfoot\@empty\let\@oddhead\@empty
\let\@evenfoot\@empty\let\@evenhead\@empty}
\end{verbatim}

The definition of the page style \texttt{plain} is again simple.

\begin{verbatim}
\ps@plain
\def\ps@plain{\let\@oddhead\@empty
\def\@oddfoot{\normalfont\hfil\thepage}
\def\@evenfoot{\normalfont\hfil\thepage}}
\end{verbatim}

\section{Document Markup}

\subsection{Global Declarations}

The following declarations, shown with examples, give information about the sender:

- \texttt{\name{Dr. L. User}}: to be used for the return address on the envelope.
- \texttt{\signature{Larry User}}: goes after the closing.
- \texttt{\address{3245 Foo St. GNU York}}: used as the return address in the letter and on the envelope. If not declared, then an institutional standard address is used.
- \texttt{\location{Room 374}}: Acts as modifier to the standard institutional address.
- \texttt{\telephone{(415) 123-4567}}: Just in case some style puts it on the letter.

\begin{verbatim}
\newcommand*{\name}[1]{\def\fromname{#1}}
\newcommand{\fromname}{}
\newcommand*{\ondertekening}[1]{\def\fromsig{#1}}
\newcommand{\fromsig}{}
\end{verbatim}

This macro stores the signature.
\makelabels
The \makelabels declaration causes mailing labels to be made.

\begin{document}
\makelabels
\AtBeginDocument{% \let\@startlabels=startlabels \let\@mlabel=mlabel \if@filesw \immediate\write\@mainaux{\string\@startlabels}\fi}%
\AtEndDocument{% \if@filesw\immediate\write\@mainaux{\string\clearpage}\fi}}
\makelabels is allowed only before the \begin{document} command.

6.2 The generic letter commands

\brief
The \brief environment creates a new letter, starting from page 1. (The first page is unnumbered.) It has a single argument, which is the addressee and his address, as in

\begin{brief}{Sam Jones \\ Institute for Retarded Study\\ Princeton, N.J.}
Local declarations, such as \address, can follow the \begin{brief}.
\begin{environment}{brief}{1} %
\newpage
\if@twoside \ifodd\c@page
\else\thispagestyle{empty} \hbox{}\newpage\fi
\fi
\c@page\@ne
\interlinepenalty=200 \% smaller than the TeXbook value
The \leavevmode and \ignorespaces commands are there for protecting against an empty argument.
\process{\leavevmode\ignorespaces #1}
Now we can start filling in the various fields in the references line. First the
adressee.
\defrefveld\@Ad\{\geadresseerdetekst\}{\toname}
Then the date. When nothing was specified we use \vandaag.
\ifdim\wd\@Dt=0cm \defrefveld\@Dt\{\datumtekst\}{\vandaag}\fi
Now we can prepare the letterheads. It couldn’t be done earlier because the user
can specify that he uses a different kind of ‘window envelope’.
\prepareerhoofden
We may need to adapt the height of the head and the text body on the following
pages. Therefore we measure the height of the head on those pages.
\setbox\@tempboxa\vervolghoofd
\@tempdima\ht\@tempboxa
\advance\@tempdima by -\@otherheadheight
\ifdim\@tempdima>0\p@
\global\advance\@otherheadheight by \@tempdima
\global\advance\@othertextheight by -\@tempdima
\fi}
We have to do the same for the foot of the letter.
\setbox\@tempboxa=\vbox{\voetregel}
\global\footskip=\ht\@tempboxa
\global\advance\footskip by \footsep}

The end of the environment possibly writes the address information on the .aux
file.
\stopletter\@par\pagebreak\@par
\if@filesw
\begingroup
\let\\relax
\let\protect\relax
\immediate\write\@auxout{\string\@mlabel{\returnaddress}{\toname\\toaddress}}\endgroup
\fi}

\begin{letter}
The letter environment is a synonym for the brief environment, to provide com-
patibility with the standard letter document class.
\end{letter}
\@processto
\@xproc
\@yproc
\@processto gets the \toname and \toaddress from the letter environment’s
macro argument. \@xproc and \@yproc are auxiliary macros.
\long\def\@processto#1{\@xproc #1\relax\let\\unexpandable\protect
\immediate\write\@auxout{\string\@mlabel{\returnaddress}{\toname\\toaddress}}\endgroup
\@yproc
\long\def\@xproc #1{\@unexpandable\protect
\long\def\@yproc #1##1{\long\def\@xproc \#1\#2000{\def\toname{##1}\def\toaddress{##2}}
\@else
\@xproc #1\#1000\fi}
\@enddef\@xproc #1##1{\long\def\@yproc #1##2000{\def\toaddress{##2}}
\@enddef\@yproc #1##1{\long\def\@yproc #1##2000{\def\toaddress{##2}}
\@enddef
The command \antwoordadres takes the return address as an argument. The various parts of the address should be separated by `\`, which will be turned into bullets.

\newif\if@antwoordadres
\newcommand*\antwoordadres[1]{% 
  \@antwoordadrestrue\renewcommand*\@antwoordadres{#1}}
\newcommand*\@antwoordadres{}
\let\replyaddress\antwoordadres

\section{The address window}

The address for the letter will be placed in such a way that a ‘window envelope’ can be used to send the letter.

\adresveldbreedte The width of the address window.
\adresveld This command formats the address window.

\newdimen\adresveldbreedte
\newcommand*\adresveld{\hbox{}\kern-\topskip\kern\@vensterskip\begingroup
  \compute\width\addresswindow
  \if@adresrechts
    \setlength\adresveldbreedte{4\refveldbreedte}\addtolength\adresveldbreedte{-76mm}\def\@tempa{\moveright 76mm}\
  \else
    \let\@tempa\relax
    \setlength\adresveldbreedte{83mm}\
  \fi
  \store\address\in\abox.
  \setbox\@tempboxa\vtop{\hsize\adresveldbreedte\@normalsize\parindent\z@\parskip\z@
    \rightskip0\p@\adresveldbreedte\let\\@nobreakcr\toname\\toaddress}
\if@antwoordadres
  \@tempa\vbox to \z@{\hb@xt@\adresveldbreedte{\kleinvet\def\\\@antwoordadres\hfil}\
    \kern2\p@\hrule\vss}\fi

Format the return address if one was given.
Print a small rule as typing aid if required.
\if@typelp
\tempa\llap{\vbox to \z@{\vskip9mm\streepje\vss}}\fi

And finally print the address information. Note that this way of position the box which contains the address information has the advantage that no matter how high or deep the box is, the following information will always be printed in the same spot on the paper.
\kern9mm \kern-\ht\@tempboxa \@tempdima=\dp\@tempboxa
\@tempa\box\@tempboxa \kern-\@tempdima
\vskip31mm} \endgroup}

6.2.2 The reference line

The width of the various fields in this line. It is determined in NEN 3516
\newdimen\refveldbreedte
\setlength\refveldbreedte{38mm}
\@defrefveld A macro to help in defining the various fields.
\def\@defrefveld#1#2#3{
\setbox#1\@refveld{#2}{#3}}
\@refveld The macro \@refveld stores the formatted field in a box.
\def\@refveld#1#2{
\vtop{\hsize\refveldbreedte
\parskip\z@\parindent\z@
\everypar{}% 
\lineskiplimit\z@\baselineskip12\p@ 
\lineskip\z@
\rightskip0\p@ \@plus \refveldbreedte \@minus .5\refveldbreedte 
\vbox{\refkopfont\baselineskip10\p@\@par}
\kern2\p@ 
\strut #2}}
\@UB \@UK \@OK \@Dt
We allocate four box registers to store the four fields in
\@UB \@UK \@OK \@Dt
\uwbrief The command \uwbrief can be used to show the date of the letter to which your letter is an answer
\newcommand*{\uwbrief}[1]{\@defrefveld{\@UB}{\uwbrieftekst}{#1}}
\let\yourletterof=\uwbrief
\uwkenmerk The command \uwkenmerk can be used to show the reference of the letter to which your letter is an answer
\newcommand*{\uwkenmerk}[1]{\@defrefveld{\@UK}{\uwkenmerktekst}{#1}}
\let\yourreference=\uwkenmerk
\onskenmerk Store our reference in a box register.
\newcommand*{\onskenmerk}[1]{\@defrefveld{\@OK}{\onskenmerktekst}{#1}}
To store the date in a box register. When the user gives an empty argument no date will be printed. When he doesn’t use \datum he will get today’s date.

\newcommand{\datum}{\def@tempa{}\def@tempb{#1}%
  \ifx@tempa@tempb
  \setbox@Dt\hbox{ }
  \else
  \edef\ref@veld{\@Dt}\datumtekst{#1}%
  \fi}
\let\date\datum

This collects all the information for the reference line.

\def\referentieregel{\hbox
  {\hb@xt@\refveldbreedte{\copy@UB\hfil}\
   \hb@xt@\refveldbreedte{\copy@UK\hfil}\
   \hb@xt@\refveldbreedte{\copy@OK\hfil}\
   \hb@xt@\refveldbreedte{\copy@Dt\hfil}\hss}}

On the second and following pages a simple reference line can be printed. It contains the address information, the date and the page number.

\newbox@Ad
\def\vervolgreferentieregel{%
  \hbox{\
    \hb@xt@\refveldbreedte{\copy@Ad\hfil}%
    \hskip\refveldbreedte
    \hb@xt@\refveldbreedte{\copy@Dt\hfil}%
    "\@refveld{\bladnummertext}{\thepage}\hss}}

The headings are empty by default.

\newcommand{\briefhoofd}{}
\newcommand{\vervolghoofd}{\vbox{}}

The usage of this command creates non-empty headers.

\newcommand{\maakbriefhoofd}{\@ifstar{\@kortvervolgbriefhoofd}{\@langvervolgbriefhoofd}}
\let\makeheader\maakbriefhoofd

This creates a shortened heading for following pages

\newcommand{\@kortvervolgbriefhoofd}{%}
\@maakbriefhoofd{#1}{#2}%
\def\vervolghoofd{\vbox{\hspace{4}\refveldbreedte
  \hb@xt@\normalfont\sffamily\Large\strut\hfil
  \hrule \kern2mm \vervolgreferentieregel}}}

\@Ad
For this purpose we need to allocate another box register.

\newbox{\@Ad}
\def\vervolgreferentieregel{\hbox
  {\hb@xt@\refveldbreedte{\copy@Ad\hfil}%
   \@refveld{\bladnummertext}{\thepage}\hss}}

The headers and footers

\briefhoofd
\vervolghoofd
\maakbriefhoofd
\@kortvervolgbriefhoofd
\@langvervolgbriefhoofd  This creates a long heading for following pages by just using \briefhoofd.
\@maakbriefhoofd  This was used in the two preceding macros; it defines \briefhoofd.
\@voetruimte  A box to store the footer in.
\@voetteller  We need to know how many items are placed in the footer.
\voetregel  \voetregel just copies the box \@voetruimte.
\voetitem  A command to add an information field to the footer.
\streepje  A shorthand for one little rule.
\streepjes  This prints the folding rules
Then we can print a rule on the left side of the paper, half way down to align for a perforator.
\llap{\perfstreepje\kern24mm}\hfill
The folding rules are printed on the right hand side of the paper.
\rlap{\kern24mm\vouwstreepjes}\vss}}

\perfstreepje Prints a \streepje halfway down the paper. A4 paper is 297 mm high; we start from a position 13mm below the edge of the paper. Hence the \kern 135mm.
\newcommand*{\perfstreepje}{\vtop{\kern\z@ \kern 135mm \streepje}}
\vouwstreepjes This prints two folding rules.
\newcommand*{\vouwstreepjes}{\vtop{\kern\z@
\kern 95mm \% 108-13 \streepje \% denk maar dat dit geen dikte heeft \kern 45mm \% 155-150 \streepje}}

6.2.5 Page breaking control
\stopbreaks
\newcommand*{\stopbreaks}{\interlinepenalty @M\def\par{\@@par \nobreak}\let\\=\@nobreakcr
\let\vspace\@nobreakvspace}
\@nobreakvspace
\DeclareRobustCommand{\@nobreakvspace}{\@ifstar{\@nobreakvspacex}{\@nobreakvspacex}}
\@nobreakvspacex{\ifvmode\nobreak\vskip #1\relax\else\@bsphack\vadjust{\nobreak\vskip #1}\@esphack\fi}
\@nobreakcr{\let\reserved@e\relax\let\reserved@f\relax\vadjust{\nobreak}\@ifstar{\@xnewline}{\@xnewline}}
\startbreaks
\def\startbreaks{\let\\=\@normalcr\interlinepenalty 200\def\par{\@par\penalty 200\relax}}
\opening Text is begun with the \opening command, whose argument generates the salutation, as in
\opening{Dear Henry,}
This should produce everything up to and including the ‘Dear Henry,’ and a
command that follows. Since there’s a \vfil at the bottom of every page, it
can add vertical fil to position a short letter. It should use the following commands:

- \toname : name part of ‘to’ address. Will be one line long.
- \toaddress : address part of ‘to’ address. The lines separated by \\
- \fromname : name of sender.
- \fromaddress : argument of current \address declaration– null if none. Should use standard institutional address if null.
- \fromlocation : argument of current \location declaration–null if none.
- \telefonenum : argument of current \telephone declaration–null if none.

\opening
\thispagestyle{firstpage}
\adresveld
\prevdepth=-1000\p@ \vskip-2\p@ \@dosubject #1\par
\nobreak}
\@dosubject
\@dosubject This prints the subject of the letter if one was specified.
\@dosubject
\def\@dosubject{% 
\ifx\@empty\@subject 
\else 
\par\noindent \parbox[t]{\textwidth}{\@hangfrom{\refkopfont \betrefttekst \enspace}normalfont\rmfamily\ignorespaces \@subject\strut}% 
\par 
\fi} 
\afsluiting
The body of the letter follows, ended by a \afsluiting command, as in
\afsluiting{Yours truly,} 
\closing
This commands generates the closing matter, and the signature. An obvious
thing to do is to use a \parbox for the closing and the signature. Should use the
following:

- \fromsig : argument of current \signature declaration or, if null, the
- \stopbreaks : a macro that inhibits page breaking.

\afsluiting
\stopbreaks
\fromsig
\stopbreaks
\def\ondertekening##1{\def\fromsig{##1}\@afsluiting{#1}}%
The internal command \@afsluiting takes care of printing the closing text.

\newcommand*{\@afsluiting}[1]{% 
\def\en{\strut\egroup\open@af} %
\let\and\en
\parbox{.5\textwidth}{
\raggedright \ignorespaces #1\[6\medskipamount]\%}
\leavevmode\open@af \fromsig \strut\egroup}}

Of these three, only \medskipamount is actually used above.

\betreft

The command \betreft (\re) stores the subject of the letter.

\newcommand*{\betreft}[1]{\def\@subject{#1}}
\let\onderwerp\betreft
\let\subject\betreft
\def\@subject{}
\let\re\betreft

After the \closing you can put arbitrary stuff, which is typeset with zero \parindent and no page breaking. Commands designed for use after the closing are:

\cc(Tinker\Evers\Chance)

which produces:

cc: Tinker
Evers
Chance

Note the obvious use of \parbox.

\bijlage

which produces:

bijlagen: Foo(2)
Bar

bijlagen: Foo(2)
Bar

\encl

bijlagen: Foo(2)
Bar
The only thing \ps needs to do is call \startbreaks, which allows page breaking again.

\stopletter The \stopletter command is called by \endletter to do the following:

- Add any desired fil or other material at the end of the letter.
- Define \returnaddress to be the return address for the mailing label. More precisely, it is the first argument of the \mlabel command described below. It should be defined to null if the return address doesn’t appear on the labels. Any command, other than \, that should not be expanded until the \mlabel command is actually executed must be preceded by \protect. Whenever possible, \protect commands in the definition of \returnaddress—it’s much more efficient that way. In particular, when the standard return address is used, you should define \returnaddress to something like \protect\standardreturnaddress.

6.3 Customizing the labels

Commands for generating the labels are put on the .AUX file, which is read in and processed by the \end{document} command. You have to define the following two commands:

- \startlabels: Should reset the page layout parameters if necessary.
- \mlabel{⟨return address⟩}{⟨to address⟩}: Command to generate a single label.

\returnaddress

\labelcount

\startlabels The following \startlabels command sets things up for producing labels in two columns of five 2” × 4-1/4” labels each, suitable for reproducing onto Avery brand number 5352 address labels.
\let\@texttop\relax
\topmargin -50\p@
\headsep \z@
\oddsidemargin -35\p@
\evensidemargin -35\p@
\textheight 10in
\@colht\textheight \@colroom\textheight \vsize\textheight
\textwidth 550\p@
\columnsep 26\p@
\ifcase\@ptsize\relax
\normalsize
\or
\small
\or
\footnotesize
\fi
\baselineskip \z@
\lineskip \z@
\boxmaxdepth \z@
\parindent \z@
\twocolumn\relax}

\@startlabels
\@startlabels is the command name that is written to the .aux file. It is a no-op at first, and defined to be the same as \startlabels in the \begin{document} hook.
\let\@startlabels=\relax

\mlabel
This command prints an address label; it is used when the user specified \makelabels in the preamble of his document. The command \mlabel takes two arguments; the second argument is supposed to be the address; the first argument can be used to print a return address. In this document class we ignore the first argument. Also the labels are supposed to be 2 inch high and 3.6 inch wide. When your address labels have a different width you will have to defined your own \mlabel command.
\newcommand*\mlabel[2]{\parbox[b][2in][c]{262\p@}{\strut\ignorespaces #2}}

\@mlabel
\@mlabel is written to the .aux file in place of \mlabel. That allows to define it as a no-op per default, and activate it in the \begin{document} hook.
\let\@mlabel=\@gobbletwo

6.4 Lists
6.4.1 General List Parameters
The following commands are used to set the default values for the list environment’s parameters. See the \LaTeX{} manual for an explanation of the meanings
of the parameters. Defaults for the list environment are set as follows. First, \rightmargin, \listparindent and \itemindent are set to 0pt. Then, for a Kth level list, the command \@listK is called, where ‘K’ denotes ‘i’, ‘ii’, ‘iii’, ‘iv’, ‘v’, ‘vi’. (I.e., \@listiii is called for a third-level list.) By convention, \@listK should set \leftmargin to \leftmarginK.

\leftmargini For efficiency, level-one list’s values are defined at top level, and \@listi is defined to set only \leftmargini.

\leftmarginii The following three are calculated so that they are larger than the sum of \labelsep and the width of the default labels (which are ‘(m)’, ‘vii.’ and ‘M.’).

\leftmarginiii \setlength\leftmargini {2.5em}
\leftmarginiv \setlength\leftmarginii {2.2em}
\leftmarginv \setlength\leftmarginiii {1.87em}
\leftmarginvi \setlength\leftmarginiv {1.7em}
\leftmarginvii \setlength\leftmarginv {1em}
\leftmarginviii \setlength\leftmarginvi {1em}
Here we set the top level leftmargin.

\leftmargin The following three are calculated so that they are larger than the sum of \labelsep and the width of the default labels (which are ‘(m)’, ‘vii.’ and ‘M.’).

\labelsep \setlength\labelsep {5\p@}
\labelwidth \setlength\labelwidth\leftmargini
\addtolength\labelwidth{-\labelsep}
\partopsep When the user leaves a blank line before the environment an extra vertical space of \partopsep is inserted, in addition to \parskip and \topsep.
\topsep \setlength\topsep{0\p@}
\parskip Extra vertical space, in addition to \parskip, added above and below list and paragraphing environments.
\@beginparpenalty These penalties are inserted before and after a list or paragraph environment.
\@endparpenalty They are set to a bonus value to encourage page breaking at these points.
\@itempenalty This penalty is inserted between list items.
\@listI \@listi \@listI defines top level and \@listi values of \leftmargin, \parsep, \topsep, and \itemsep.
\@listI These values have been taken from the ones in the document class artikel3.
\@listi \def\@listi\@leftmargin\leftmargini
\labelsep.5em
\labelwidth\leftmargin
\@listi \def\@listi\@leftmargin\leftmargini
\labelsep.5em
\labelwidth\leftmargin
We have to initialise these parameters.

Here are the same macros for the higher level lists.

6.4.2 Enumerate

The enumerate environment uses four counters: \textit{enumi}, \textit{enumii}, \textit{enumiii} and \textit{enumiv}, where \textit{enumN} controls the numbering of the \textit{N}th level enumeration.

\texttt{\renewcommand*{\theenumi}{\@arabic{\c@enumi}}} \texttt{\renewcommand*{\theenumii}{\@alph{\c@enumii}}}
The label for each item is generated by the commands \labelenumi ... \labelenumiv.

The expansion of p@enumN \theenumN defines the output of a \ref command when referencing an item of the Nth level of an enumerated list.

6.4.3 Itemize

Itemization is controlled by \labelitemi, \labelitemii, \labelitemiii, and \labelitemiv, which define the labels of the various itemization levels: the symbols used are bullet, bold endash, asterisk and centred dot.

\labelitemfont The default definition for \labelitemfont is to reset the font to \normalfont so that always the same symbol is produced regardless of surrounding conditions. A possible alternative would be
\renewcommand\labelitemfont{\% \fontseries\seriesdefault \fontshape\shapedefault\selectfont} which resets series and shape doesn’t touch the family.

6.4.4 Description

The description environment is defined here – while the itemize and enumerate environments are defined in the L\LaTeX\ kernel (ltlists.dtx).

description The description environment is defined here – while the itemize and enumerate environments are defined in the L\LaTeX\ kernel (ltlists.dtx).

\descriptionlabel To change the formatting of the label, you must redefine \descriptionlabel.
\renewcommand*\descriptionlabel[1]{\hspace{\labelsep}normalfont\bfseries #1}
6.5 Defining new environments

6.5.1 Verse

The verse environment is defined by making clever use of the list environment’s parameters. The user types `\` to end a line. This is implemented by `\let`’ing `\` equal `\@centercr`.

\newenvironment{verse}{\let\=\@centercr\list{}{\setlength\itemsep{\z@}\setlength\itemindent{-15\p@}\setlength\listparindent{\itemindent}\setlength\rightmargin{\leftmargin}\addtolength\leftmargin{15\p@}}\item[]}{{\endlist}}

6.5.2 Quotation

The quotation environment is also defined by making clever use of the list environment’s parameters. The lines in the environment are set smaller than `\textwidth`. The first line of a paragraph inside this environment is indented.

\newenvironment{quotation}{\list{}{\setlength\listparindent{1.5em}\setlength\itemindent{\listparindent}\setlength\rightmargin{\leftmargin}}\item[]}{{\endlist}}

6.5.3 Quote

The quote environment is like the quotation environment except that paragraphs are not indented.

\newenvironment{quote}{\list{}{\setlength\rightmargin{\leftmargin}}\item[]}{{\endlist}}

6.5.4 Theorem

This document class does not define its own theorem environments, the defaults, supplied by \LaTeX kernel (\texttt{ltthm.dtx}) are available.

6.6 Setting parameters for existing environments

6.6.1 Array and tabular

`\arraycolsep` The columns in an array environment are separated by `2\arraycolsep`.

\setlength\arraycolsep{5\p@}

29
\tabcolsep  The columns in an tabular environment are separated by \tabcolsep.
646 \setlength{\tabcolsep}{6\p@}
\arrayrulewidth  The width of vertical rules in the array and tabular environments is given by \arrayrulewidth.
647 \setlength{\arrayrulewidth}{.4\p@}
\doublerulesep  The space between adjacent rules in the array and tabular environments is given by \doublerulesep.
648 \setlength{\doublerulesep}{2\p@}

6.6.2 Tabbing
\tabbingsep  This controls the space that the `\' command puts in. (See \LaTeX{} manual for an explanation.)
649 \setlength{\tabbingsep}{\labelsep}

6.6.3 Minipage
\@minipagerestore  The macro \@minipagerestore is called upon entry to a minipage environment to set up things that are to be handled differently inside a minipage environment. In the current styles, it does nothing.
\@mpfootins  Minipages have their own footnotes; \skip\@mpfootins plays same rôle for footnotes in a minipage as \skip\footins does for ordinary footnotes.
650 \skip\@mpfootins = \skip\footins

6.6.4 Framed boxes
\fboxsep  The space left by \fbox and \framebox between the box and the text in it.
\fboxrule  The width of the rules in the box made by \fbox and \framebox.
651 \setlength{\fboxsep}{3\p@}
652 \setlength{\fboxrule}{.4\p@}

6.6.5 Equation and eqnarray
\theequation  The equation counter will be typeset using arabic numbers.
653 \renewcommand{\theequation}{\@arabic{\c@equation}}
\jot  \jot is the extra space added between lines of an eqnarray environment. The default value is used.
654 % \setlength{\jot}{3pt}
\@eqnnum  The macro \@eqnnum defines how equation numbers are to appear in equations. Again the default is used.
655 % \def{\@eqnnum}{(\theequation)}
6.7 Font changing

Here we supply the declarative font changing commands that were common in \LaTeX version 2.09 and earlier. These commands work in text mode and in math mode. They are provided for compatibility, but one should start using the \text... and \math... commands instead. These commands are redefined using \@renewfontswitch, a command with three arguments: the user command to be defined; \LaTeX commands to execute in text mode and \LaTeX commands to execute in math mode.

\begin{verbatim}
\def\rmfamily{\normalfont}\rm
\end{verbatim}

\begin{verbatim}
\def\sfamily{\normalfont}\sf
\end{verbatim}

\begin{verbatim}
\def\ttfamily{\normalfont}\tt
\end{verbatim}

\bf

The command to change to the bold series. One should use \mdseries to explicitly switch back to medium series.

\begin{verbatim}
\def\bfseries{\normalfont}\bf
\end{verbatim}

\sl

And the commands to change the shape of the font. The slanted and small caps shapes are not available by default as math alphabets, so those changes do nothing in math mode. One should use \upshape to explicitly change back to the upright shape.

\begin{verbatim}
\def\itshape{\normalfont}\it
\end{verbatim}

\begin{verbatim}
\def\ssshape{\relax}\ss
\end{verbatim}

\cal

The commands \cal and \mit should only be used in math mode, outside math mode they have no effect. Currently the New Font Selection Scheme defines these commands to generate warning messages. Therefore we have to define them ‘by hand’.

\begin{verbatim}
\def\mathcal{\@fontswitch{\relax}{\mathcal}}
\end{verbatim}

\begin{verbatim}
\def\mathnormal{\@fontswitch{\relax}{\mathnormal}}
\end{verbatim}

6.8 Footnotes

\footnoterule

Usually, footnotes are separated from the main body of the text by a small rule. This rule is drawn by the macro \footnoterule. We have to make sure that the rule takes no vertical space (see plain.tex) so we compensate for the natural height of the rule of 0.4pt by adding the right amount of vertical skip.

To prevent the rule from colliding with the footnote we first add a little negative vertical skip, then we put the rule and make sure we end up at the same point where we begun this operation.

\begin{verbatim}
\renewcommand*{\footnoterule}{%
\kern-\p@
\hrule \@width .4\columnwidth
\kern .6\p@}
\end{verbatim}
Footnotes are numbered within chapters in the report and book document styles.

The footnote mechanism of \texttt{\LaTeX} calls the macro \texttt{@makefntext} to produce the actual footnote. The macro gets the text of the footnote as its argument and should use \texttt{@makefnmark} to produce the mark of the footnote. The macro \texttt{@makefntext} is called when effectively inside a \texttt{parbox} of width \texttt{columnwidth} (i.e., with \texttt{\hsize = \columnwidth}).

An example of what can be achieved is given by the following piece of \TeX code.

\begin{verbatim}
\long\def\@makefntext#1{\%\setpar\@@par
  \@tempdima = \hsize
  \advance\@tempdima-10pt
  \parshape \@ne 10pt \@tempdima
  \parindent 1em\noindent
  \hb@xt@\z@{\hss\@makefnmark}#1}
\end{verbatim}

The effect of this definition is that all lines of the footnote are indented by 10pt, while the first line of a new paragraph is indented by 1em. To change these dimensions, just substitute the desired value for ‘10pt’ (in both places) or ‘1em’. The mark is flush right against the footnote.

In these document classes we use a simpler macro, in which the footnote text is set like an ordinary text paragraph, with no indentation except on the first line of a paragraph, and the first line of the footnote. Thus, all the macro must do is set \texttt{\parindent} to the appropriate value for succeeding paragraphs and put the proper indentation before the mark.

\begin{verbatim}
\long\def\@makefntext#1{\noindent\hb@xt@\leftmargini{\normalfont\@thefnmark.\hfil}#1}
\end{verbatim}

The footnote markers that are printed in the text to point to the footnotes should be produced by the macro \texttt{@makefnmark}.

\section{Words}

This document class supports a number of languages. All words that will be printed by the class code are stored in commands which can be redefined if you want to use a different language.

This stores dutch strings.

\begin{verbatim}
\newcommand*{\dutchbrief}{\%
  \def\uwbrieftekst{Uw brief van}
  \def\uwkenmerktekst{Uw kenmerk}
  \def\onskenmerktekst{Ons kenmerk}}
\end{verbatim}
\englishbrief This stores English strings.
\americanbrief This stores American English strings.
\emph{\germanbrief} This stores the German versions of the strings.
\begin{verbatim}
\newcommand*{\germanbrief}{% 
\def\uwbrieftekst{Ihr Brief vom} 
\def\uwkenmerktekst{Ihr Zeichen} 
\def\onskenmerktekst{Unser Zeichen} 
\def\datumtekst{Datum} 
\def\geadresseerdetekst{An} 
\def\bladnummertekst{Seite}\ifcase\month\or Januar\or Februar\or März\or April\or Mai\or Juni\or Juli\or August\or September\or Oktober\or November\or Dezember\fi 
\space\number\year} 
\def\betrefttekst{Betrifft:} 
\def\ccname{Kopien an} 
\def\bijlagetekst{Anlage:} 
\def\bijlagentekst{Anlagen:} 
\def\telefoontekst{Telefon}}
\end{verbatim}

\emph{\frenchbrief} And finally to store the french strings
\begin{verbatim}
\newcommand*{\frenchbrief}{% 
\def\uwbrieftekst{Votre lettre du} 
\def\uwkenmerktekst{Vos r\'ef\'erences:} 
\def\onskenmerktekst{Nos r\'ef\'erences:} 
\def\datumtekst{Date:} 
\def\geadresseerdetekst{\`A l’attention de} 
\def\bladnummertekst{Page}\ifcase\month\or janvier\or février\or mars\or avril\or mai\or juin\or juillet\or août\or septembre\or octobre\or novembre\or décembre\fi 
\space\number\year} 
\def\betrefttekst{Objet:} 
\def\ccname{Copie \`a} 
\def\bijlagetekst{Pi\`ece jointe:} 
\def\bijlagentekst{Pi\`eces jointes:} 
\def\telefoontekst{Telefon}}
\end{verbatim}

\subsection{Two column mode}
\begin{verbatim}
\setlength{\columnsep}{10\p@} 
\setlength{\columnseprule}{1\p@} 
\end{verbatim}
This gives the distance between two columns in two column mode.

This gives the width of the rule between two columns in two column mode. We have no visible rule.
6.11 The page style

We have headings pages in this document class by default. We use arabic pagination.

\pagestyle{headings}
\pagenumbering{arabic}

6.12 Single or double sided printing

We don’t try to make each page as long as all the others.

\raggedbottom
\let\texttop\relax

We always start in one column mode.

\onecolumn
\pagebreak[3]

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.

<table>
<thead>
<tr>
<th>Symbols</th>
<th>\beginparpenalty . 562</th>
<th>\evenhead .......</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;</td>
<td>. . . . . . . . . . 736</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>\</td>
<td>746, 747, 753, 755, 760</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@par</td>
<td>308, 372, 451, 466</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@A</td>
<td>294, 393, 396</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@B</td>
<td>295, 375</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@D</td>
<td>383, 385, 392, 398</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@DK</td>
<td>375, 380, 391</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@UB</td>
<td>375, 376, 389</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@UK</td>
<td>375, 378, 390</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@antwoordadres</td>
<td>52</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@antwoordadresttrue</td>
<td>53</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@afsluiting</td>
<td>. . . . . . . . . . 485, 487, 497</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@antwoordadres</td>
<td>. . . . . . . . . . 325, 326, 353</td>
<td>@beginparpenalty . 562</td>
</tr>
<tr>
<td>@auxout</td>
<td>. . . . . . . . . . 313</td>
<td>@beginparpenalty . 562</td>
</tr>
</tbody>
</table>

\@beginparpenalty . 562
\@evenhead .......
Change History

v2.0b

\textregistered: Can't use (re)newcommand for \textregistered as that breaks the test against \textregistered. 23

v2.0d

\textregistered: Can't use (re)newcommand for \textregistered as that breaks the test against \textregistered. 23
\labelitemiii to 
\labelitemiii \hfill 28
v2.0e
\ps@headings: Removed a typo
(\othertheadheight) \hfill 13
v2.0g
\mit: Now define \cal and \mit
using
\DeclareRobustCommand* \hfill 31
\mlabel: Redefined \mlabel to not
use the \setbox primitive \hfill 25
General: Use \newcommand* instead
of \newcommand in most places \hfill 1
brief: No longer redefine
\protect but use one of the
available settings \hfill 16
v2.0h
General: Added a \relax to
prevent an incomplete \ifcase
error \hfill 6
v2.0i
\englishbrief: Repaired typo \hfill 33
\v2.0j
\englishbrief: Repaired typo \hfill 33
\v2.0k
\englishbrief: Repaired typo \hfill 33
\v2.0l
\englishbrief: Repaired typo \hfill 33
\v2.0m
\vrefveld: Also set \parindent to
zero \hfill 18
\adresveld: Set both \parskip
and \parindent to zero in
\adresveld \hfill 17
v2.0n
\adresveld: Repaired a typo
(\parskip) \hfill 17
\v2.0o
\labelitemiv: Changed -- to
\textendash following
classes.dtx \hfill 28
Did similar for the bullet and
centered dot. \hfill 28
\v2.0p
\@nobreakcr: Added setting of
\reserved@e and \reserved@f
as this is now needed for \LaTeX. \hfill 21
\labelitemiv: Now also
\textasteriskcentered \hfill 28
v2.0q
\@makefnmark: Use the default
definition for \@makefnmark \hfill 32
\@makefntext: As we want to have
different appearances of the
footnotemarker in the text and
in the footnotes, we can't use
\@makefnmark here \hfill 32
\adresveld: Put \textbullet in a
group to keep the font change
local \hfill 17
\closing: Added \leavevmode to
get the signatures on one line \hfill 23
\streepje: Make the 'streepje's a
little smaller \hfill 20
\vouwstreepjes: Change the
positioning of the 'streepje's a
little \hfill 21
General: Added a few more
synonimes for commands \hfill 1
\v2.0s
\@nobreakvspace: Made robust
(\LaTeX pt/2049) \hfill 21
\normalsize: Roll back handling
(gh/201) \hfill 7
\labelitemfont: Normalize label
fonts \hfill 28
\texttt{\textbackslash small}: Use \texttt{\textbackslash DeclareRobustCommand} instead of \texttt{\textbackslash newcommand*} \hspace{0.5cm} \textit{General: Synchronised with the standard document classes} \hspace{0.5cm} \textbf{1}

41