A \LaTeX{} style option to attach
Line numbers on paragraphs

Stephan I. Böttcher
stephan@nevis.columbia.edu

Contents

1 Introduction 2

2 Put the line numbers to the lines 3

3 Control line numbering
   3.1 Display math 7

4 Line number references
   4.1 The linelabel command 9

5 The appearance of the line numbers
   5.1 Running line numbers 11
   5.2 Pagewise line numbers 12
   5.3 Numbering modulo 5 16

6 Package options
   6.1 Package Extensions 18
      6.1.1 displaymath 19
      6.1.2 Line numbers in internal vertical mode 19
      6.1.3 Line number references with offset 20
      6.1.4 Numbered quotation environments 21
      6.1.5 Frame around a paragraph 21
1 Introduction

This package provides line numbers on paragraphs. After \TeX has broken a paragraph into lines there will be line numbers attached to them, with the possibility to make references through the \TeX \texttt{\ref}, \texttt{\pageref} cross reference mechanism. This includes four issues:

\begin{itemize}
\item attach a line number on each line,
\item create references to a line number,
\item control line numbering mode,
\item count the lines and print the numbers.
\end{itemize}

The first two points are implemented through patches to the output routine. The third by redefining \texttt{\par}, \texttt{\@par} and \texttt{\@@par}. The counting is easy, as long as you want the line numbers run through the text. If they shall start over at the top of each page, the aux-file as well as \TeX's memory have to carry a load for each counted line.

I wrote this package for my wife Petra, who needs it for transcriptions of interviews. This allows her to precisely refer to passages in the text. It works well together with \texttt{\marginpar}s, but not to well with \texttt{displaymath}. \texttt{\footnotes} are a problem, especially when they are split, but we may get there. \texttt{lineno.sty} works surprisingly well with other packages, for example, \texttt{wrapfig.sty}. So please try if it works with whatever you need, and if it does, please tell me, and if it does not, tell me as well, so I can try to fix it.

This style option is written for \TeX $2\varepsilon$, later than November 1994, since we need the \texttt{\protect\write} macro.
2 Put the line numbers to the lines

The line numbers have to be attached by the output routine. We simply set
the \interlinepenalty to -100000. The output routine will be called after
each line in the paragraph, except the last, where we trigger by \par. The
\linenopenalty is small enough to compensate a bunch of penalties (e.g.,
with \samepage).

\newcount\linenopenalty\linenopenalty=-100000
\mathchardef\linenopenaltypar=25000

So let’s make a hook to \output, the direct way. The \LaTeX macro
@reinserts puts the footnotes back on the page.
(New v3.01) \reinserts badly screws up split footnotes. The bottom
part is still on the recent contributions list, and the top part will be put back
there after the bottom part. Thus, since lineno.sty does not play well with
\inserts anyway, we can safely experiment with \holdinginserts, without
making things much worse.
Or that’s what I thought, but: Just activating \holdinginserts while
doing the \par will not do the trick: The \output routine may be called
for a real page break before all line numbers are done, and how can we get
control over \holdinginserts at that point?
Let’s try this: When the \output routine is run with \holdinginserts=3
for a real page break, then we reset \holdinginserts and restart \output.
Then, again, how do we keep the remaining \inserts while doing further
line numbers?
If we find \holdinginserts=-3 we activate it again after doing \output.
(/New v3.01)
(New v3.02) To work with multicol.sty, the original output routine is
now called indirectly, instead of being replaced. When multicol.sty changes
\output, it is a toks register, not the real thing. (/New v3.02)

\let\@LN@output\output
\newtoks\output
\output=%\expandafter{\the\@LN@output}
\@LN@output={%
LineNoTest
  \if@tempswa
  LineNoHoldInsertsTest
  \if@tempswa
    \the\output
    \ifnum\holdinginserts=-3
      \global\holdinginserts 3
    fi
  fi
}
The float mechanism inserts \texttt{\interlinepenalty}s during \texttt{\output}. So carefully reset it before going on. Else we get doubled line numbers on every float placed in horizontal mode, e.g, from \texttt{\linelabel}.

Sorry, neither a \texttt{\linelabel} nor a \texttt{\marginpar} should insert a penalty, else the following linenumber could go to the next page. Nor should any other float. So let us suppress the \texttt{\interlinepenalty} altogether with the \texttt{\@nobreak} switch.

Since (ltspace.dtx, v1.2p)[1996/07/26], the \texttt{\@nobreaktrue} does it’s job globally. We need to do it locally here.

We have to return all the page to the current page, and add a box with the line number, without adding breakpoints, glue or space. The depth of our
line number should be equal to the previous depth of the page, in case the page breaks here, and the box has to be moved up by that depth.

The \interlinepenalty comes after the \vadjust from a \linelabel, so we increment the line number \emph{after} printing it. The macro \makeLineNumber produces the text of the line number, see section 5. Finally we put in the natural \interlinepenalty, except after the last line.

3 Control line numbering

The line numbering is controlled via \par. \LaTeX{} saved the \TeX{}-primitive \par in \@@par. We push it one level further out, and redefine \@@par to insert the \interlinepenalty needed to trigger the line numbering. And we need to allow pagebreaks after a paragraph.

New (2.05beta): the prevgraf test. A paragraph the ends with a displayed equation, a \noindent\par or \wrapfig.sty produce empty paragraphs. These should not get a spurious line number via \linenopenaltypar.
The basic commands to enable and disable line numbers. \@par and \par
are only touched, when they are \let to \@@@par/\linenumberpar. The line
number may be reset to 1 with the star-form, or set by an optional argument
[\langle number\rangle].

\def\linenumbers{\let\@@par\linenumberpar
\ifx\@par\@@@par\let\@par\linenumberpar\fi
\@ifnextchar[{{\resetlinenumber}}]{\@ifstar{{\resetlinenumber}}{}}
}
\def\nolinenumbers{\let\@@par\@@@par
\ifx\@par\linenumberpar\let\@par\@@@par\fi
\ifx\par\linenumberpar\let\par\@@@par\fi
}

What happens with a display math? Since \par is not executed, when break-
ing the lines before a display, they will not get line numbers. Sorry, but I do
not dare to change \interlinepenalty globally, nor do I want to redefine
the display math environments here.

display math

See the subsection below, for a wrapper enviroment to make it work. But
that requires to wrap each and every display in your LaTeX source.
The next two commands are provided to turn on line numbering in
a specific mode. Please note the difference: for pagewise numbering,
\linenumbers comes first to inhibit it from seeing optional arguments, since
re-/presetting the counter is useless.

\def\pagewiselinenumbers{\linenumbers\setpagewiselinenumbers}
\def\runninglinenumbers{\setrunninglinenumbers\linenumbers}

Finally, it is a \LaTeX style, so we provide for the use of environments, includ-
ing the suppression of the following paragraphs indentation.

\let\endlinenumbers\endlinenumbers
\expandafter\let\csname endlinenumbers\endcsname\endlinenumbers
\let\endnolinenumbers\endlinenumbers
3.1 Display math

Now we tackle the problem to get display math working. There are different options.

1. Precede every display math with a \par. Not too good.
2. Change \interlinepenalty and associates globally. Unstable.
3. Wrap each display math with a \{linenomath\} environment.

We’ll go for option 3. See if it works:

\[ \text{display math} \]

(1)

The star form \{linenomath*\} should also number the lines of the display itself,

\[ \text{multi line display math with array} \]

(2) (3) (4)

including multiline displays.

First, here are two macros to turn on linenumering on paragraphs preceding displays, with numbering the lines of the display itself, or without. The \ifx\ tests if line numbering is turned on. It does not harm to add these wrappers in sections that are not numbered. Nor does it harm to wrap a display twice, e.g. in case you have some \{equation\}s wrapped explicitly, and later you redefine \equation to do it automatically.

\begin{verbatim}
\newcommand\linenomathNonumbers{%
  \ifx\@@par\@@@par\else
    \ifnum\interlinepenalty>\-\linenopenalty\par
      \global\holdinginserts3%
      \advance\interlinepenalty \linenopenalty
      \advance\predisplaypenalty \linenopenalty
    \fi
  \fi
  \ignorespaces
}\
\newcommand\linenomathWithnumbers{%
  \ifx\@@par\@@@par\else
    \ifnum\interlinepenalty>\-\linenopenalty\par
      \global\holdinginserts3%
      \advance\interlinepenalty \linenopenalty
      \advance\predisplaypenalty \linenopenalty
    \fi
  \fi
  \ignorespaces
}\
\end{verbatim}
The \{linenomath\} environment has two forms, with and without a star. The following two macros define the environment, where the stared/non-stared form does/doesn't number the lines of the display or vice versa.

\newcommand\linenumberdisplaymath{\%
  \def\linenomath{\linenomathWithnumbers}\%
  \@namedef{linenomath*}{\linenomathNonumbers}\%
}
\newcommand\nolinenumberdisplaymath{\%
  \def\linenomath{\linenomathNonumbers}\%
  \@namedef{linenomath*}{\linenomathWithnumbers}\%
}
\def\endlinenomath{\%
  \global\holdinginserts0
  \@ignoretrue
}
\expandafter\let\csname endlinenomath\endcsname\endlinenomath

The default is not to number the lines of a display. But the package option \texttt{mathlines} may be used to switch that behavior.

4 Line number references

The only way to get a label to a line number in a paragraph is to ask the output routine to mark it.

We use the marginpar mechanism to hook to \texttt{output} for a second time. Marginpars with number $-1$, we fake marginpars with No $-2$. Originally, every negative numbered float was considered to be a marginpar. The float box number \texttt{@currbox} is used to transfer the label name in a macro called \texttt{@LNLE<box-number>}. A \texttt{newlabel} is written to the aux-file. The reference is to \texttt{\thelineNumber}, \textit{not} \texttt{\thelinenumber}. This allows to hook in, as done below for pagewise line numbering.
(New v3.03) The \LN@ExtraLabelItems are added for a hook to keep packages like \{hyperref\} happy. (/New v3.03)

\let\LN@addmarginpar\@addmarginpar
\def\@addmarginpar{\ifnum\count\@currbox>-2\relax
\expandafter\LN@addmarginpar\else\@cons\@freelist\@currbox\protected@write\@auxout{}{%
\string\newlabel\{{\csname @LNL@\the\@currbox\endcsname}{{\theLineNumber}{\thepage}\LN@ExtraLabelItems}}%
\fi}
\let\LN@ExtraLabelItems\@empty

4.1 The linelabel command

To refer to a place in line \ref{⟨foo⟩} at page \pageref{⟨foo⟩} you place a \linelabel{⟨foo⟩} at that place.

If you use this command outside a \linenumbers paragraph, you will get references to some bogus line numbers, sorry. But we don’t disable the command, because only the \par at the end of a paragraph may decides whether to print line numbers on this paragraph or not. A \linelabel may legally appear earlier than \linenumbers.

\linelabel, via a fake float number −2, puts a \penalty into a \vadjust, which triggers the pagebuilder after putting the current line to the main vertical list. A \write is placed on the main vertical list, which prints a reference to the current value of \thelinenumber and \thepage at the time of the \shipout.

A \linelabel is allowed only in outer horizontal mode. In outer vertical mode we start a paragraph, and ignore trailing spaces (by fooling \@esphack).

The argument of \linelabel is put into a macro with a name derived from the number of the allocated float box. Much of the rest is dummy float setup.

\def\linelabel#1{%
\ifvmode\ifinner \else
\leavevmode \@bsphack \@savsk\p@
\fi\else
\fi
5 The appearance of the line numbers

The line numbers are set as \texttt{\tiny\sffamily\arabic{linenumber}}, 10pt left of the text. With options to place it right of the text, or . . .

. . . here are the hooks:

\begin{verbatim}
def\makeLineNumberLeft{\hss\linenumberfont\LineNumber\hskip\linenumbersep}
def\makeLineNumberRight{\linenumberfont\hskip\linenumbersep\hbox to\linenumberwidth{\hss\LineNumber}\hss}
def\linenumberfont{\normalfont\tiny\sffamily}
\newdimen\linenumbersep
\newdimen\linenumberwidth
\linenumberwidth=10pt
\linenumbersep=10pt
\def\switchlinenumbers{\@ifstar
  {\let\makeLineNumberOdd\makeLineNumberRight
   \let\makeLineNumberEven\makeLineNumberLeft}
  {\let\makeLineNumberOdd\makeLineNumberLeft}
}
\end{verbatim}

Margin switching requires \texttt{pagewise} numbering mode, but choosing the left or right margin for the numbers always works.
\let\makeLineNumberEven\makeLineNumberRight
}

\def\setmakelinenumbers#1{\@ifstar
  \let\makeLineNumberRunning#1
  \let\makeLineNumberOdd#1
  \let\makeLineNumberEven#1
\fi%
}

\def\leftlinenumbers{\setmakelinenumbers\makeLineNumberLeft}
\def\rightlinenumbers{\setmakelinenumbers\makeLineNumberRight}

\leftlinenumbers*

\LineNumber is a hook which is used for the modulo stuff. It is the command to use for the line number, when you customizes \makeLineNumber. Use \thelinenumber to change the outfit of the digits.

We will implement two modes of operation:

- numbers running through (parts of) the text
- pagewise numbers starting over with one on top of each page.

Both modes have their own count register, but only one is allocated as a \LaTeX counter, with the attached facilities serving both.

\newcounter{linenumber}
\newcount\c@pagewiselinenumber
\let\c@runninglinenumber\c@linenumber

Only the running mode counter may be reset, or preset, for individual paragraphs. The pagewise counter must give a unique anonymous number for each line.

\newcommand\resetlinenumber[1][1]{\c@runninglinenumber#1}

5.1 Running line numbers

Running mode is easy, \LineNumber and \thelineNumber produce \thelinenumber, which defaults to \arabic{linenumber}, using the \c@runninglinenumber counter. This is the default mode of operation.
5.2 Pagewise line numbers

Difficult, if you think about it. The number has to be printed when there is no means to know on which page it will end up, except through the aux-file. My solution is really expensive, but quite robust.

With version v2.00 the hashsize requirements are reduced, because we no longer need one controlsequence for each line any more. But this costs some computation time to find out on which page we are.

\texttt{\textbackslash makeLineNumber} gets a hook to log the line and page number to the aux-file. Another hook tries to find out what the page offset is, and subtracts it from the counter \texttt{\textbackslash c@linenumber}. Additionally, the switch \texttt{\textbackslash ifoddNumberedPage} is set true for odd numbered pages, false otherwise.

Each numbered line gives a line to the aux file

\texttt{\textbackslash @LN\{\langle line\rangle\}\{\langle page\rangle\}}

very similar to the \texttt{\newlabel} business, except that we need an arabic representation of the page number, not what there might else be in \texttt{\thepage}.
From the aux-file we get one macro \LN@P\langle page\rangle for each page with line numbers on it. This macro calls four other macros with one argument each. These macros are dynamically defined to do tests and actions, to find out on which page the current line number is located.

We need sort of a pointer to the first page with line numbers, initialized to point to nothing:

\def\LastNumberedPage{first}
\def\LN@Pfirst{\nextLN\relax}

The four dynamic macros are initialized to reproduce themselves in an \xdef

\let\lastLN\relax % compare to last line on this page
\let\firstLN\relax % compare to first line on this page
\let\pageLN\relax % get the page number, compute the linenumber
\let\nextLN\relax % move to the next page

During the end-document run through the aux-files, we disable @LN. I may put in a check here later, to give a rerun recommendation.

\AtEndDocument{\let@LN@gobbletwo}

Now, this is the tricky part. First of all, the whole definition of @LN is grouped, to avoid accumulation on the save stack. Somehow \csname cs\endcsname pushes an entry, which stays after an \xdef to that cs.

If \LN@P\langle page\rangle is undefined, initialize it with the current page and line number, with the pointer-to-the-next-page pointing to nothing. And the macro for the previous page will be redefined to point to the current one.

If the macro for the current page already exists, just redefine the last-line-number entry.

Finally, save the current page number, to get the pointer to the following page later.

\def@LN#1#2{{\expandafter@LN@csname LN@P#2\endcsname{#1}{#2}}}
\def@LN#1#2#3{{\ifx#1\relax\expandafter@LN@csname LN@P\LastNumberedPage\endcsname#1\else\expandafter@LN@csname LN@P\LastNumberedPage\endcsname#1\fi\def#1{#1}\gdef\LastNumberedPage{#3}}
The previous page macro gets its pointer to the current one, replacing the \relax with the cs-token \LN@P\langle\text{page}\rangle. In case of page number mismatch, \TeX will trip here, because the argument string for \nextLN is not \relax. I think it's difficult to do a reasonable intercept here, because this is running in an \xdef. Does \PackageError{} work in there?

Now, to print a line number, we need to find the page, where it resides. This will most probably be the one where the last one came from or maybe the next. However, it can be a completely different one. We maintain a cache, which is let to the last accessed pages macro. But for now it is initialized to expand \LN@first, where the pointer to the first numbered page has been stored in.

To find out on which page the current \c@linenumber is, we define the four dynamic macros to do something usefull and execute the current cache macro. \lastLN is run first, testing if the line number in question may be on a later page. If so, disable \firstLN, and go on to the next page via \nextLN.

Else, if \firstLN finds out that we need an earlier page, we start over from the beginning. Else, \nextLN will be disabled, and \pageLN will run \gotNumberedPage with two arguments: the first line number on this page, and the page number.

We start with \pageLN disabled and \nextLN defined to continue the search with the next page.
When we switch to another page, we first have to make sure that it is there. If we are done with the last page, we probably need to run \TeX{} again, but for the rest of this run, the cache macro will just return two zeros. This saves a lot of time, for example if you have half of an aux-file from an aborted run, in the next run the whole page-list would be searched in vain again and again for the second half of the document.

If there is another page, we iterate the search.

To separate the official hooks from the internals there is this equivalence, to hook in later for whatever purpose:

So, now we got the page where the number is on. We establish if we are on an odd or even page, and calculate the final line number to be printed.

You might want to run the pagewise mode with running line numbers, or you might not. It’s your choice:
For line number references, we need a protected call to the whole procedure, with the requested line number stored in the \c@linenumber counter. This is what gets printed to the aux-file to make a label:

\def\thePagewiseLineNumber{\protect\getpagewiselinenumber{\the\c@linenumber}}%

And here is what happens when the label is referred to:

\def\getpagewiselinenumber#1{{\c@linenumber #1\relax\testNumberedPage \helinenumber}}

A summary of all per line expenses:

**CPU:** The \texttt{output} routine is called for each line, and the page-search is done.

**DISK:** One line of output to the aux-file for each numbered line

**MEM:** One macro per page. Great improvement over v1.02, which had one control sequence per line in addition. It blew the hash table after some five thousand lines.

### 5.3 Numbering modulo 5

Most users want to have only one in five lines numbered. \texttt{\LineNumber} is supposed to produce the outfit of the line number attached to the line, while \texttt{\thelinenumber} is used also for references, which should appear even if they are not multiples of five.

\newcount\c@linenumbermodulo

\def\themodulolinenumber{{\@tempcnta\c@linenumber
\divide\@tempcnta\c@linenumbermodulo
\multiply\@tempcnta\c@linenumbermodulo
\ifnum\@tempcnta=\c@linenumber\thelinenumber\fi}}
The user command to set the modulo counter:

```
\newcommand{\modulolinenumbers}[1][0]{%
  \let\LineNumber{\themodulolinenumber}
  \ifnum#1>1\relax
    \c@linenumbermodulo#1\relax
  \else\ifnum#1=1\relax
    \def\LineNumber{\thelinenumber}%
  \fi\fi
}
\setcounter{linenumbermodulo}{5}
\modulolinenumbers[1]
```

6 Package options

There is a bunch of package options, all of them executing only user commands (see below).

Options `left` (`right`) put the line numbers on the left (right) margin. This works in all modes. `left` is the default.

```
\DeclareOption{left}{\leftlinenumbers*}
\DeclareOption{right}{\rightlinenumbers*}
```

Option `switch` (`switch*`) puts the line numbers on the outer (inner) margin of the text. This requires running the pagewise mode, but we turn off the page offset subtraction, getting sort of running numbers again. The pagewise option may restore true pagewise mode later.

```
\DeclareOption{switch}{\setpagewiselinenumbers
  \switchlinenumbers
  \runningpagewiselinenumbers}
\DeclareOption{switch*}{\setpagewiselinenumbers
  \switchlinenumbers*%
  \runningpagewiselinenumbers}
```

The options `pagewise` and `running` select the major linenumber mechanism. `running` line numbers refer to a real counter value, which can be reset for any paragraph, even getting multiple paragraphs on one page starting with line number one. `pagewise` line numbers get a unique hidden number within the document, but with the opportunity to establish the page on which they finally come to rest. This allows the subtraction of the page offset, getting
the numbers starting with 1 on top of each page, and margin switching in
twoside formats becomes possible. The default mode is running.

The order of declaration of the options is important here pagewise must
come after switch, to override running pagewise mode. running comes last,
to reset the running line number mode, e.g., after selecting margin switch
mode for pagewise running. Once more, if you specify all three of the options
[switch,pagewise,running], the result is almost nothing, but if you later
say \pagewiselinenumbers, you get margin switching, with real pagewise
line numbers.

\DeclareOption{pagewise}{\setpagewiselinenumbers
\realpagewiselinenumbers}
\DeclareOption{running}{\setrunninglinenumbers}

The option modulo causes only those linenumbers to be printed which are
multiples of five.

\DeclareOption{modulo}{\modulolinenumbers\relax}

The package option mathlines switches the behavior of the \linenomath
environment with its star-form. Without this option, the \linenomath
environment does not number the lines of the display, while the star-form
does. With this option, its just the opposite.

\DeclareOption{mathlines}{\linenumberdisplaymath}

displaymath now calls for wrappers of the standard LaTeX display math
environment. This was previously done by mlineno.sty.

\let\do@mlineno\relax
\DeclareOption{displaymath}{\let\do@mlineno\@empty}

The hyperref package, via nameref, requires three more groups in the second
argument of a newlabel. Well, why shouldn’t it get them? TODO: This
should be automagically detected \AtBeginDocument.

\DeclareOption{hyperref}{\def@LN@ExtraLabelItems{{}{}{}}}
\ProcessOptions

6.1 Package Extensions

The extensions in this section were previously supplied in separate .sty files.
6.1.1 \textit{displaymath}

The standard \LaTeX\ display math environments are wrapped in a \{linenomath\} environment.

\begin{verbatim}
\ifx\do@mlineno\@empty
  \renewenvironment{displaymath}
    {\linenomath\[
    \]}
    {\endlinenomath}
\renewenvironment{equation}
    {\linenomath$$\refstepcounter{equation}$$}
    {\eqno\hbox{@eqnnum}$$\endlinenomath$$}
\let\LN@eqnarray\eqnarray
\let\LN@endeqnarray\endeqnarray
\renewenvironment{eqnarray}
    {\linenomath\LN@eqnarray}
    {\LN@endeqnarray\endlinenomath}
\fi
\end{verbatim}

6.1.2 Line numbers in internal vertical mode

The command \texttt{\internallinenumbers} adds line numbers in internal vertical mode, but with limitations: we assume fixed baseline skip.

\begin{verbatim}
\def\internallinenumbers{\setrunninglinenumbers
\let\@par\internallinenumberpar
\ifx\@par\@@@par\let\@par\internallinenumberpar\fi
\ifx\par\@@@par\let\par\internallinenumberpar\fi
\let\@@par\internallinenumberpar
\@ifnextchar[{esetlinenumber}%\]
\ifx\@par\@@@par\let\@par\internallinenumberpar\fi
\ifx\par\@@@par\let\par\internallinenumberpar\fi
\ifx\@par\linenumberpar\let\@par\internallinenumberpar\fi
\ifx\par\linenumberpar\let\par\internallinenumberpar\fi
\@ifnextchar[{esetlinenumber}%\]
  {\@ifstar{\let@c@linenumber@c@internallinenumber
            c@linenumber@ne}{}}%}
}\let\endinternallinenumbers\endlinenums
\@namedef{internallinenumbers*}{\internallinenumbers*}
\expandafter\let\csname endinternallinenumbers*\endcsname\endlinenums
\newcount@c@internallinenumber
\newcount@c@internallinenumber
\def\internallinenumberpar{\ifvmode\@@@par\else\ifinner\@@@par\else\@@@par
\end{verbatim}
6.1.3 Line number references with offset

This extension to defines macros to refer to line numbers with a offset, e.g., to refer to a line which cannot be labeled directly (display math). This was formerly known as rllineno.sty.

To refer to a pagewise line number with offset:

\linerefp\[⟨OFFSET⟩\]{⟨LABEL⟩}

To refer to a running line number with offset:

\linerefr\[⟨OFFSET⟩\]{⟨LABEL⟩}

To refer to a line number labeled in the same mode as currently selected:

\lineref\[⟨OFFSET⟩\]{⟨LABEL⟩}
This goes deep into \TeX's internals.

6.1.4 Numbered quotation environments

The \{numquote\} and \{numquotation\} environments are like \{quote\} and \{quotation\}, except there will be line numbers.

An optional argument gives the number to count from. A star * (inside or outside the closing \}) prevent the reset of the line numbers. Default is to count from one.

6.1.5 Frame around a paragraph

The \{bframe\} environment draws a frame around some text, across page breaks, if necessary.

This works only for plain text paragraphs, without special height lines. All lines must be \baselineskip apart, no display math.
7 The final touch

There is one deadcycle for each line number.
8 The user commands

The user command to turn on and off line numbering are

\linenumbers

Turn on line numbering in the current mode.

\linenumbers*

and reset the line number to 1.

\linenumbers[⟨number⟩]

and start with ⟨number⟩.

\nolinenumbers

Turn off line numbering.

\runninglinenumbers*[⟨number⟩]

Turn on running line numbers, with the same optional arguments as \linenumbers. The numbers are running through the text over page-breaks. When you turn numbering off and on again, the numbers will continue, except, of course, if you ask to reset or preset the counter.

\pagewiselinenumbers

Turn on pagewise line numbers. The lines on each page are numbered beginning with one at the first pagewise numbered line.

\resetlinenumber[⟨number⟩]

Reset [Set] the line number to 1 [⟨number⟩].

\setrunninglinenumbers

Switch to running line number mode. Do not turn it on or off.

\setpagewiselinenumbers

Switch to pagewise line number mode. Do not turn it on or off.

\switchlinenumbers*

Causes margin switching in pagewise modes. With the star, put the line numbers on the inner margin.

\leftlinenumbers*


\rightlinenumber

Set the line numbers in the left/right margin. With the star this works for both modes of operation, without the star only for the currently selected mode.

\runningpagewiselinenumbers

When using the pagewise line number mode, do not subtract the page offset. This results in running line numbers again, but with the possibility to switch margins. Be careful when doing line number referencing, this mode status must be the same while setting the paragraph and during references.

\realpagewiselinenumbers

Reverses the effect of \runningpagewiselinenumbers.

\modulolinenumbers[\langle number\rangle]

Give a number only to lines which are multiples of \langle number\rangle. If \langle number\rangle is not specified, the current value in the counter linenumbermodulo is retained. \langle number\rangle=1 turns this off without changing linenumbermodulo. The counter is initialized to 5.

\linenumberdisplaymath

Number the lines of a display math in a \{linenomath\} environment, but do not in a \{linenomath\*} environment. This is used by the package option \[displaymath\].

\nolinenumberdisplaymath

Do not Number the lines of a display math in a \{linenomath\} environment, but do in a \{linenomath\*} environment. This is the default.

\linelabel

Set a \linelabel{\langle foo\rangle} to the line number where this command is in. Refer to it with the L\TeX\ referencing commands \ref{\langle foo\rangle} and \pageref{\langle foo\rangle}.

The commands can be used globally, locally within groups or as environments. It is important to know that they take action only when the \par is executed. The \end\{mode\}linenumbers commands provide a \par. Examples:
8.1 Customization hooks

There are several hooks to customize the appearance of the line numbers, and some low level hooks for special effects.

\thelinenumber

This macro should give the representation of the line number in the \LaTeX-counter \texttt{linenumber}. The default is provided by \LaTeX:

\begin{verbatim}
\texttt{\arabic{linenumber}}
\end{verbatim}

\makeLineNumberLeft

This macro is used to attach a line number to the left of the text page. This macro should fill an \texttt{\hbox to 0pt} which will be placed at the left margin of the page, with the reference point aligned to the line to which it should give a number. Please use the macro \texttt{\LineNumber} to refer to the line number.

The default definition is

\begin{verbatim}
\texttt{\hss\linenumberfont\LineNumber\hskip\linenumbersep}
\end{verbatim}

\makeLineNumberRight

Like \texttt{\makeLineNumberLeft}, but for line numbers on the right margin. The default definition is

\begin{verbatim}
\texttt{\linenumberfont\hskip\linenumbersep\hskip\textwidth}
\hbox to\linenumberwidth{\hss\LineNumber}\hsss
\end{verbatim}
\linenumberfont

This macro is initialized to \normalfont\tiny\sffamily

\linenumbersep

This dimension register sets the separation of the linenumber to the text. Default value is 10pt.

\linenumberwidth

This dimension register sets the width of the line number box on the right margin. The distance of the right edge of the text to the right edge of the line number is \linenumbersep + \linenumberwidth. The default value is 10pt.

\thelineNumber (for wizards)

This macro is called for printing a \newlabel entry to the aux-file. Its definition depends on the mode. For running line numbers it’s just \thelinenumber, while in pagewise mode, the page offset subtraction is done in here.

\makeLineNumber (for wizards)

This macro produces the line numbers. The definition depends on the mode. In the running line numbers mode it just expands \makeLineNumberLeft.

\LineNumber (for wizards)

This macro is called by \makeLineNumber to typeset the line number. This hook is changed by the modulo mechanism.