The `notes.sty` package for marking special sections in a document with icons

Duncan Webb*

2002/10/29

Abstract

This package provides environments to highlight significant portions of text within a document by putting the text in a box and adding an icon in the margin. It has been designed specifically for double sided printing catering for the LaTeXpage numbering problem.

1 Introduction

The notes package displays a block of italic text in a frame and placed in the margin is an icon.

`importantnote`
`warningnote`
`informationnote`

Currently there are three environments defined `importantnote`, `warningnote` and `informationnote`.

Included with the package there are three icons and some LyX include files. The icons are provided as Xfig diagrams. A make file will generate the style sheet, this documentation and other graphic formats. The other formats include encapsulated postscript, portable document format and portable network graphics. The `.eps` files are used when the document is printed, the `.pdf` files are used when the document is converted to pdf and the `.png` files are used when the document is converted to HTML.

It may be necessary to remove the `.png` *files before a document is converted to pdf because in this format they look dreadful; they are best suited for HTML pages.*

The LyX files should be placed in your `~/.lyx` directory so that they are then available from LyX.

It is quite simple to change the icons or to add new note environments. To change the icons, either use your own and rename the files to the appropriate name.

To add a new enviroment just copy an existing one and change its name. Then you can change the icon name or the font.

*email: duncan@dwebb.ch

2 Usage

In the document preamble include the `usepackage[...]{notes}`. Generally, no options need to be specified.

To begin a note we use the one of the three environments, such as `importantnote`

```
\usepackage{notes}
...
\begin{importantnote}
This is how an important note is shown
\end{importantnote}
```

and the result is:

This is how an important note is shown

Here is an example of an warning note.

This is an example of a warning note.

There is no limitation on the amount of text in the box, it can be a single line or several lines. However, it is not recommended to have a page full of text as this is not the purpose of the package.

3 To Do

There are a number of things that need to be done, possibly others that I've not yet thought about.

① Use the `.aux` mechanism, this may be more reliable and allow cleaner code.

4 The Package

1 ⟨∗package⟩

Definable settings are listed below. `noteskipamount` is some glue that is printed before and after the note. `textwidth` is the length of the printable area. `marginparsep` is the amount of space between the icon and the main body of the text. `rulewd` and `ruleht` are the thickness of the frame arount the notes box. `textwd` is the amount of space that text in the note uses up. `vframegap` and `hframegap` are the gaps between the text and its frame.

```
2 \newskip\noteskipamount \noteskipamount=6pt plus 4pt minus 0pt
3 \newdimen\textwidth     \textwidth=\hsize
4 \newdimen\marparsep     \marparsep=4pt
5 \newdimen\rulewd        \rulewd=0.4pt
6 \newdimen\ruleht        \ruleht=0.4pt
7 \newdimen\iconwd        \iconwd=0.8cm
```

```
\newdimen\iconht        \iconht=0.8cm
\newdimen\textwd        \textwd=.8\textwidth
\newdimen\vframegap     \vframegap=3pt
\newdimen\hframegap     \hframegap=6pt
```

`debug` The `debug` option set the debug flag, which numbers each of the icons and write in the log debug information.

```
\newif\if@debug@
\DeclareOption{debug}{%
  \global\@debug@true
}%
```

`frames` The `frames` option sets the frame flag, which prints frames around the icon and the text.

```
\newif\if@frames@
\DeclareOption{frames}{%
  \global\@frames@true
}%
\ProcessOptions*
```

Here are some dimensions that are calculated during the processing of the note. `notecnt` keeps a count of the notes.

```
\newcounter{notecnt}
\setcounter{notecnt}{0}
\newdimen\containerwd
\newdimen\textframeht
\newdimen\iconframeht
\newdimen\containerht
\newdimen\notesmargin
\newdimen\dbgrulewd \dbgrulewd=0pt
\newdimen\dbgruleht \dbgruleht=0pt
\if@frames@
  \dbgrulewd=0.01pt \dbgruleht=0.01pt
\fi
\def\noteskip{\vskip\noteskipamount}
```

`boxdbg` The `boxdbg` command puts a frame around the argument if the debug flag is set. The mechanism is to define a width for the line, which is 0pt when the debug flag is *not* set and `dbgrulewd`, `dbgruleht` when the debug flag is set.

```
\def\boxdbg#1{\hbox{%
  \vrule width \dbgrulewd\vbox{%
    \hrule height \dbgruleht%
    \hbox{#1}\hrule height \dbgruleht%
  }%
  \vrule width \dbgrulewd}%
}
```

`buildtextframe` The `buildtextframe` command creates a frame around the argument. Space is placed around the items, the vertical space is defined by `vframegap` and the horizonal space is defined by the `hframegap`.

```
\def\buildtextframe#1{%
  \vbox{%
```

```
    \hrule height \ruleht%
    \hbox{%
      \vrule width \rulewd\kern\vframegap%
      \vbox{\kern\hframegap#1\kern\hframegap}%
      \kern\vframegap\vrule width \rulewd%
    }%
    \hrule height \ruleht%
  }%
}
```

`buildiconbox` The `buildiconbox` command creates an box around the icon, the icon is placed in a rectanglar box of `iconwd` by `iconht`. The default box is a 1cm sqaure. When the debug flag is set the note number is written at the bottom left of the icon.

```
\def\buildiconbox#1{
  \if@debug@
    \raisebox{0pt}[0pt][0pt]{\makebox[0pt][c]{\tiny\thenotecnt}}%
  \fi
  \vbox{%
    \hsize \iconwd \noindent \hbox to \iconwd{%
      \hfil\resizebox*{\iconwd}{\iconht}{\includegraphics{#1}}\hfil%
    }%
  }%
}
```

`buildiconframe` the `buildiconframe` command puts its argument is a vbox of `iconht`.

```
\def\buildiconframe#1{%
  \vbox to \iconht{\vfil\hbox{\buildiconbox{#1}}\vfil}%
}%
```

`calccontainerht` The `calccontainerht` command determines the greater height of the icon or the text. It returns the maximum of the two dimensions passed.

```
\def\calccontainerht#1#2{
  \ifnum #1 > #2 #1 \else #2 \fi
}
```

`buildvcontainer` The `buildvcontainer` command creates an vbox to contain both the icon and the text boxes. #1 is the container height, #2 is the box to be contained

```
\def\buildvcontainer#1#2{%
  \boxdbg{\vbox to #1{\vfil#2\vfil}}
}
```

`buildcontainer` The `buildcontainer` command creates a hbox of the icon and the text box. In a double sided document the icon is placed on the right hand side for odd pages numbers and the left hand side for even page numbers. In a single sided document the icon is placed in the left hand side margin. #1 is the text container, #2 is the icon container

```
\def\buildcontainer#1#2{%
  \if@twoside
    \ifodd\count0
      \hbox to \containerwd{%
        \hskip\notesmargin\hfil#1\hfil\hskip\marparsep#2%
```

```
      }
    \else
      \hbox to \containerwd{
        #2\hskip\marparsep\hfil#1\hfil\hskip\notesmargin%
      }
    \fi
  \else %single sided
    \hbox to \containerwd{%
      #2\hskip\marparsep\hfil#1\hfil\hskip\notesmargin%
    }
  \fi
}
```

`buildnotes` Now we start the complicated bit! It is not really complicated but because `\ifodd\count0` is not always correct in LaTeXit means that the amount of vertical space that the note takes up and the amount space left on the page must be calculated before the note is written to the page. LaTeXupdates the page when a vertical skip is sent to the page.

First some boxes and dimensions are defined.

```
\newbox\iconframe
\newbox\textframe
\newbox\iconcontainer
\newbox\textcontainer
\newbox\container
\newbox\notesbox
\newdimen\pageleft
\newdimen\notesboxht
```

The height of the text frame and the icon frame are calculated. From these dimensions the height of the note is calculated. Six `dbgruleht`s are added because there are three boxes drawn when the frame flag is set.

LaTeXdoes not update the page total, the `noteskip` command flushes the value. The `pageleft` dimension is initially set to the `pagegoal` dimension (the total space on the page) and then it reduced by `pagetotal` (the space used by the text). `pageleft` is then reduced by the `noteskipamount`, `notesboxht`, `noteskipamount`. Finally a magic number is added to the amount left. I've no idea where this magic number comes from, but it was found by seeing that the amount of space left on the page was not zero. Trial and lots of errors discovered that this was a constant. #1 is the text, #2 is the icon.

```
\def\buildnotes#1#2{%
  \stepcounter{notecnt}
  \if@debug@
    \typeout{NOTE \thenotecnt \space started}
  \fi
  \setbox\textframe=\buildtextframe{#1}
  \setbox\iconframe=\buildiconframe{#2}
  \containerht=\calccontainerht{\ht\iconframe}{\ht\textframe}
  % check the page remaining
  \notesboxht=\containerht
  \advance\notesboxht by \dbgruleht
  \advance\notesboxht by \dbgruleht
  \advance\notesboxht by \dbgruleht
  \advance\notesboxht by \dbgruleht
```

```
\advance\notesboxht by \dbgruleht
\advance\notesboxht by \dbgruleht
\noteskip % this forces the pagetotal to be updated
\pageleft=\pagegoal
\advance\pageleft by -\pagetotal
\advance\pageleft by -\noteskipamount %
\advance\pageleft by -\notesboxht
\advance\pageleft by -\noteskipamount %
\advance\pageleft by \pageshrink % I'm not sure about this
% increase the space left on the page by a magic number.
\advance\pageleft by 0.87083pt
\if@debug@
  \typeout{pageleft(0)=\the\pageleft}
  \typeout{wd textframe=\the\wd\textframe}
  \typeout{wd iconframe=\the\wd\iconframe}
\fi
\setbox\textcontainer=\buildvcontainer{\containerht}{\box\textframe}
\setbox\iconcontainer=\buildvcontainer{\containerht}{\box\iconframe}
\notesmargin=\marparsep
\advance\notesmargin by \wd\iconcontainer
% calc the container width
\containerwd=\textwidth
\containerwd=355pt % \textwidth can be strange
\advance\containerwd by \notesmargin
\advance\containerwd by \notesmargin
\if@debug@
  \typeout{notesboxht=\the\notesboxht}
  \typeout{containerht=\the\containerht}
  \typeout{containerwd=\the\containerwd}
  \typeout{pageshrink(1)=\the\pageshrink}
  \typeout{pagetotal(1)=\the\pagetotal}
  \typeout{pageleft(1)=\the\pageleft}
  \typeout{textframeht=\the\textframeht}
  \typeout{iconframeht=\the\iconframeht}
  \typeout{wd iconcontainer=\the\wd\iconcontainer}
\fi
% if there is insufficient space left eject the page
\ifnum \pageleft < 0 \eject \fi
% build the container box
\setbox\container=\vbox{%
  \buildcontainer{\boxdbg{\copy\textcontainer}}{%
    \boxdbg{\copy\iconcontainer}%
  }%
}
\setbox\notesbox=\boxdbg{\copy\container}
\notesboxht=\ht\notesbox
\if@debug@\typeout{notesboxht=\the\notesboxht}\fi
% build the container box
\moveleft \notesmargin\copy\notesbox\noteskip
% check the page remaining
\pageleft=\pagegoal
\advance\pageleft by -\pagetotal
\advance\pageleft by -\notesboxht
\if@debug@
```

```
    \typeout{pagetotal(2)=\the\pagetotal}
    \typeout{pageleft(2)=\the\pageleft}
    \typeout{NOTE \thenotecnt \space finished}
  \fi
}

\newbox\notebox
```

`importantnote` The `importantnode` enviroment is defined here. A vbox is placed around the text, which is formatted to `textwd`. The `hand` icon is passed with the vbox to the `buildnotes` command.

```
\newenvironment{importantnote}%
{%
\begingroup%
\setbox\notebox=\vbox\bgroup\hsize\textwd\noindent\bgroup\it%
}{
\egroup
\egroup
\buildnotes{\box\notebox}{hand}
\endgroup%
}
```

`warningnote` The `warningnote` enviroment is defined here; the `warn` image is used for this environment.

```
\newenvironment{warningnote}%
{%
\begingroup%
\setbox\notebox=\vbox\bgroup\hsize\textwd\noindent\bgroup\it%
}{
\egroup
\egroup
\buildnotes{\box\notebox}{warn}
\endgroup%
}
```

`informationnote` The `informationnote` enviroment is defined here; the `info` image is used for this environment.

```
\newenvironment{informationnote}%
{%
\begingroup%
\setbox\notebox=\vbox\bgroup\hsize\textwd\noindent\bgroup\it%
}{
\egroup
\egroup
\buildnotes{\box\notebox}{info}
\endgroup%
}
```

⟨/package⟩

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.