

The `todonotes` package*

Henrik Skov Midtiby
henrikmidtiby@gmail.com

September 5, 2008

Abstract

The `todonotes` package allows you to insert to-do items in your document. At any point in the document a list of all the inserted to-do items can be listed with the `\listoftodos` command.


1 Introduction

The `todonotes` package makes three commands available to the user: `\todo[]{}{}`, `\missingfigure{}` and `\listoftodos`. `\todo[]{}{}` and `\missingfigure{}` makes it possible to insert notes in your document about things that has to be done later (`todonotes ...`). I developed the basic functionality of the package while I worked on my bachelor project.

1.1 Usage

`\todo` My most common usage of the `todonotes` package, is to insert an uncustomized `todonotes` somewhere in a latex document. An example of this usage is the command

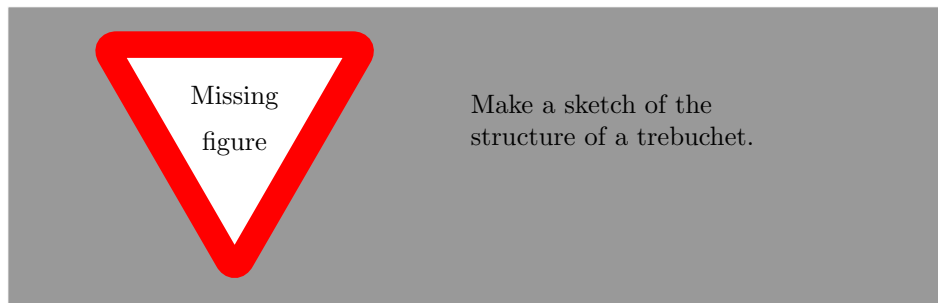
`\todo{Make a cake \ldots},`

which renders like . The `\todo` command has this structure: `\todo[options]{}{todo text}`. The `todo text` is the text that will be shown in the todonote and in the list of todos. The optional argument `options`, allows the user to customize the appearance of the inserted todonotes. For a description of all the options see section 1.3.

`\missingfigure` The `\missingfigure` command inserts an image containing an attention sign and the given text. The command takes only one argument `\missingfigure{<text>}`, a text string that could describe what the figure should consist of. An example of its usage could be

`\missingfigure{Make a sketch of the structure of a trebuchet.}`
which renders like.

*This document corresponds to `todonotes .dtx`, dated 2008/08/11.



`\listoftodos` The `\listoftodos` command inserts a list of all the todos in the current document. `\listoftodos` takes no arguments. For this document the list of to-do's looks like.

Todo list

■ Make a cake ...	1
Figure: Make a sketch of the structure of a trebuchet.	1
■ A red note.	3
■ And a green note	3
■ A note with no line connecting it to the placement in the original text. .	3
■ A todonote placed in the text	3
■ A note with a large fontsize.	3
■ Note with very small fontsize.	3

1.2 Package options

<code>disable</code>	If the option <code>disable</code> is passed to the package, the macros usually defined by the package (<code>\todo</code> , <code>\listoftodos</code> and <code>\missingfigure</code>) are defined as macros with no effect, and thus all inserted notes are removed.
<code>colorinlistoftodos</code>	Adds a small colored square in front of all items in the Todo list. The color of the square is the same as the fill color of the inserted todonote. This can be usefull if there are different types of todos (insert reference, explain in detail, ...) where the color of the inserted todonote marks the type of todo.
<code>color</code>	<code>color=val</code> sets the default background color of the inserted todonotes to <code>val</code> . The default value is <code>color=orange</code> . <code>val</code> must be a color understood by the <code>xcolor</code> package.
<code>bordercolor</code>	<code>bordercolor=val</code> sets the default border color of the inserted todonotes to <code>val</code> . The default value is <code>color=black</code> . <code>val</code> must be a color understood by the <code>xcolor</code> package.
<code>textwidth</code>	<code>textwidth=length</code> sets the width of a todo item in the margin to <code>length</code> . The width of inline todonotes will allways be the same as the current line width.
<code>dvistyle</code>	When a document with todonotes is compiled with plain latex (to a dvi-file),

there is an issue with the visual appearance¹. The option `dvistyle` changes the appearance of the inserted todonotes to avoid this problem.

1.3 Options for the `todo` command

There are several options that can be given to the `\todo` command. All the options are described here and often I have included examples of the change in visual appearance.

`color`

`color=val` sets the background color of the inserted todonotes to `val`. The default value can be set by the `color` option when the todonotes package is loaded. The todo notes inserted in this paragraph is created with the commands `\todo[color=red!40]{A red note.}` and `\todo[color=green!40]{And a green note.}`. The color of the inserted note could be used to mark different types of tasks (insert references, explain something in detail, ...), this could be streamlined by defining new commands like below.

```
\newcommand{\insertref}[1]{\todo[color=green!40]{#1}}
\newcommand{\explainindetail}[1]{\todo[color=red!40]{#1}}
```

`line / noline`

If you want to get rid of the line connecting the inserted note with the place in the text where the note occurs in the latex code, the option `noline` can be used. `\todo[noline]{A note with no line ...}`

`inline / noinline`

It is possible to place a todonote inside the text instead of placing it in the margin, this could be desirable if the text in the note has a considerable length. `\todo[inline]{A todonote placed in the text}`

A todonote placed in the text

`size`

`size=val` changes the size of the text inside the todonote. The commands used to create the notes below are `\todo[size=\Large]{A note with a large fontsize.}` and `\todo[inline, size=\tiny]{Note with very small fontsize.}`

Note with very small fontsize.

`list / nolist`

When the option `nolist` is given, the todo item will not appear in the list of todos.

1.4 Options for the `missingfigure` command

Currently the `missingfigure` command takes no optional arguments.

1.5 Options for the `listoftodos` command

Currently the `listoftodos` command takes no optional arguments.

¹The problem is placement of text inside the colored boxes.

2 Implementation

Identifies the package and loads the packages dependences.

```
1 \ProvidesPackage{todonotes}[2008/08/11]
2 \RequirePackage{ifthen}
3 \RequirePackage{xkeyval}
4 \RequirePackage{hyperref}
5 \RequirePackage{xcolor}
6 \RequirePackage{tikz}
7 \RequirePackage{calc}
```

Some default values are set

```
8 \newcommand{\@backgroundcolor}{orange}
9 \newcommand{\@bordercolor}{black}
10 \newcommand{\@textwidth}{\marginparwidth}
```

2.1 Declaration of options for the package

In this part the various options for the package are defined.

Make it possible to disable the functionality of the package. If this option is given, the commands `\todo{}` and `\listoftodos` are defined as commands with no effect. (But you can still compile you document with these commands).

```
11 \newif{\ifdisabled}
12 \DeclareOptionX{disable}{\disabledtrue}
```

Show small boxes in the list of todos with the color of the inserted todonotes.

```
13 \newif{\ifcolorinlistoftodos}
14 \DeclareOptionX{colorinlistoftodos}{\colorinlistoftodostrue}
```

The default style behaves bad when compiled with latex (some text placement problems). The `dvistyle` option, changes the visual behavior to avoid this text placement problem.

```
15 \newif{\ifdviStyle}
16 \DeclareOptionX{dvistyle}{\dviStyletrue}
```

Make the background color of the notes as an option.

```
17 \define@key{todonotes.sty}%
18   {color}{\renewcommand{\@backgroundcolor}{#1}}
```

Make the color of the notes box color as an option.

```
19 \define@key{todonotes.sty}%
20   {bordercolor}{\renewcommand{\@bordercolor}{#1}}
```

Make the text width as an option.

```
21 \define@key{todonotes.sty}%
22   {textwidth}{\renewcommand{\@textwidth}{#1}}
```

Finally process the given options.

```
23 \ProcessOptionsX
```

2.2 Options for the todo command

In this part the various options for commands in the package are defined. Set an arbitrarily fill color

```
24 \newcommand{\fillcolor}{}%
25 \define@key{todonotes}{color}{\renewcommand{\fillcolor}{#1}}%
```

Set a relative font size

```
26 \newcommand{\sizecommand}{}%
27 \define@key{todonotes}{size}{\renewcommand{\sizecommand}{#1}}%
```

Should the todo item be included in the list of todos?

```
28 \newif\ifappendtolistoftodos%
29 \define@key{todonotes}{list}[]{\appendtolistoftodostrue}%
30 \define@key{todonotes}{nolist}[]{\appendtolistoftodosfalse}%
```

Should the todo item be displayed inline?

```
31 \newif\ifinlinenote%
32 \define@key{todonotes}{inline}[]{\inlinenotetrue}%
33 \define@key{todonotes}{noinline}[]{\inlinenotefalse}%
```

Should the note in the margin be connected to the insertion point in the text?

```
34 \newif\ifline%
35 \define@key{todonotes}{line}[]{\linetrue}%
36 \define@key{todonotes}{noline}[]{\linefalse}%
```

Preset values of the options

```
37 \presetkeys
38   {todonotes}
39   {color=\@backgroundcolor, noinline,
40    line, list, size=\normalsize}{}%
```

2.3 The main code part

Here is the actual macros defined. If the option "disable" was passed to the package define empty commands.

```
41 \ifdisabled
42   \newcommand{\listoftodos}{}
43   \newcommand{\todo}[2] [] {}
44   \newcommand{\missingfigure}[1] {}
45 \else % \ifdisabled
```

Define the \listoftodos command and define the appearance of the list of todos.

```
46 \newcommand{\listoftodos}
47   {\section*{Todo list} \@starttoc{tdo}}
48 \newcommand{\l@todo}
49   {\@dottedtocline{1}{0em}{2.3em}}
```

Define styles used by the todo command

```
50 \tikzstyle{notestyle} = [draw=\@bordercolor, line width=0.5pt,
51   text width = \@textwidth - 1.6 ex - 1pt, inner sep = 0.8 ex]
52 \tikzstyle{notestyleleft} = [notestyle, left]
```

```

53 \tikzstyle{connectstyle} = [thick]
54 \tikzstyle{inlinenotestyle} = [notestyle, text width=\textwidth - 1.6 ex - 1 pt]

\todo Define the todo command
55 \newcommand{\todo}[2] [] {%
56 \setkeys{todonotes}{#1}%

Add the item to the list of todos. When the option colorinlistoftodos is given
to the package a small colored square is added in front of the text.
57 \ifappendtolistoftodos%
58 \phantomsection%
59 \ifcolorinlistoftodos%
60 \addcontentsline{tdo}{todo}{\protect{%
61 \colorbox{\fillcolor}{\textcolor{\fillcolor}{\tiny i}} %
62 #2}}%
63 \else%
64 \addcontentsline{tdo}{todo}{\protect{#2}}%
65 \fi%
66 \fi%

Place the todonote as indicated by the options (inline or in a marginpar), below
is the code for the inline placement.
67 \ifinlinenote%
68 \ifdviStyle%
69 {\par\noindent\tikz[remember picture] \draw node[inlinenotestyle, %
70 fill=\fillcolor] {};\par}%
71 {\noindent \sizecommand #2}%
72 {\par\noindent\tikz[remember picture] \draw node[inlinenotestyle, %
73 fill=\fillcolor] {};\par}%
74 \else%
75 {\par\noindent\tikz[remember picture] \draw node[inlinenotestyle, %
76 fill=\fillcolor] {\sizecommand #2};\par}%
77 \fi%
78 \else%

When the todonote should be placed inside a marginpar, the code below is applied.
First is the current location in the document stored, this enables us later to connect
this point with the inserted todonote.
79 \begin{tikzpicture}[remember picture, baseline=-0.75ex]%
80 \node [coordinate] (inText) {};%
81 \end{tikzpicture}%
82 \marginpar[{} Draw note in left margin
83 \ifdviStyle%
84 \tikz[remember picture] \draw node[notestyle, fill=\fillcolor] {}; \\\ %
85 \begin{minipage}{\@textwidth}%
86 \sizecommand #2%
87 \end{minipage} \\\ %
88 \tikz[remember picture] \draw node[notestyle, fill=\fillcolor] (inNote) {};%
89 \ifline%
90 \begin{tikzpicture}[remember picture, overlay]%

```

```

91          \draw[connectstyle, draw=\fillcolor]%
92          ([yshift=-0.2cm] inText)%
93          -| ([xshift=0.2cm] inNote.east)%
94          -| (inNote.east);%
95      \end{tikzpicture}%
96  \fi%
97 \else%
98      \tikz[remember picture] \draw node[notestyle, fill=\fillcolor]
99 (inNote) {\sizecommand #2};%
100  \ifline%
101      \begin{tikzpicture}[remember picture, overlay]%
102          \draw[connectstyle, draw=\fillcolor]%
103          ([yshift=-0.2cm] inText)%
104          -| ([xshift=0.2cm] inNote.east)%
105          -| (inNote.east);%
106      \end{tikzpicture}%
107  \fi%
108 \fi%

```

In the book documentclass (which is a twoside layout), the `\marginpar` macro takes two arguments `\marginpar []`, it is possible to let latex decide in which

```

109 }]{% Draw note in right margin
110 \ifdviStyle%
111     \tikz[remember picture] \draw node[notestyle, fill=\fillcolor] {}; \\\ %
112     \begin{minipage}{\@textwidth}%
113     \sizecommand #2%
114     \end{minipage} \\\ %
115     \tikz[remember picture] \draw node[notestyle, fill=\fillcolor] (inNote) {};%
116     \ifline%
117         \begin{tikzpicture}[remember picture, overlay]%
118             \draw[connectstyle, draw=\fillcolor]%
119             ([yshift=-0.2cm] inText)%
120             -| ([xshift=-0.2cm] inNote.west)%
121             -| (inNote.west);%
122         \end{tikzpicture}%
123     \fi%
124 \else%
125     \tikz[remember picture] \draw node[notestyle, fill=\fillcolor]
126 (inNote) {\sizecommand #2};%
127     \ifline%
128         \begin{tikzpicture}[remember picture, overlay]%
129             \draw[connectstyle, draw=\fillcolor]%
130             ([yshift=-0.2cm] inText)%
131             -| ([xshift=-0.2cm] inNote.west)%
132             -| (inNote.west);%
133         \end{tikzpicture}%
134     \fi%
135 \fi%
136 }%
137 \fi%

```

```

138 }%

\missingfigure Defines the \missingfigure macro.
139 \newcommand{\missingfigure}[1]{
140 \addcontentsline{tdo}{todo}{Figure: \protect{#1}}%
141 \par
142 \noindent
143 \begin{tikzpicture}
144 \draw[fill=black!40, draw = white, line width=0pt]
145     (-3, -2.5) rectangle +(\textwidth, 4cm);
146 \draw (3, 0) node[right, text width=4cm] {#1};
147 \draw[red, fill=white, rounded corners = 5pt, line width=10pt]
148     (30:2cm) -- (150:2cm) -- (270:2cm) -- cycle;
149 \draw (0, 0.3) node {Missing};
150 \draw (0, -0.3) node {figure};
151 \end{tikzpicture}
152 }% Ending \missingfigure command
153 \fi % Ending \ifdisabled

```

Change History

0.1	General: The first version of the package	1		an optional argument to the todo command for adding inline todonotes (Idea from Patrick Toche)	1
0.2	General: Updated the option handling of the package	1	0.4	General: Modified the behaviour of the inline todonotes, to avoid empty lines around the inline todonotes.	1
0.2.1	General: Slightly modified by Kjell Magne Fauske to support notes in the left margin (for documentstyle book).	1	0.4.1	General: Added the option colorinlistoftodos which inserts a small box with the used fillcolor of the todonotes in the list of todos. . .	1
0.2.2	General: Added a missingfigure command	1	0.4.2	General: Fixed a bug with the disable option to the package. . .	1
0.2.3	General: Made a dependency on the calc package	1	0.5	General: Created a dtx file containing both source code and documentation of the package	1
0.3	General: Delayed the requirements for the hyperref package until begin document and added				