The colorframed package

Release 0.9b of 2022/10/01 report issues at https://github.com/jfbu/colorframed/issues

Jean-François Burnol

0.9 (2022/09/22) initial release
0.9a (2022/09/23) doc fixes, mention Longbox
0.9b (2022/10/01) - fix “placement of titled-frame title is not backwards compatible” (#2)
                 - \colorframedTitleBarFrame as workaround to upstream issues with titled-frame (#3, #4)
                 - add link to #5 for usage with list items, other doc improvements such as #1

1 Description

This package fixes various colour leaks one encounters with the environments from Donald Arseneau’s package framed. Typically, colour leaks occur if using \color (at top level or whose scope involves a page break) inside the environments, or more subtly also when using \textcolor with an argument ending up being split at a page break.

This latter type of colour leak (or colour disappearance) is the more challenging one as the fix requires modifications or replacements not only of some of the framed.sty macros (such as its \CustomFBox, which colorframed overwrites) but also to some \LaTeX internals, as some environments of framed.sty rely on usage of \fbox or \colorbox. Rather than overwriting internal \LaTeX macros such as \@frameb@x or \color@b@x, colorframed simply replaces \fbox and \colorbox in the framed.sty environments by appropriate substitutes.

I am aware tcolorbox package documentation explains at least one colour issue which looks similar to those fixed here in framed context, and that the fix overthere uses an extra colour stack, hence is not \LaTeX compatible currently.

The problems are solved here without involving an extra colour stack, hence the fixes work also with \LaTeX.

2 The environments from framed

We refer the reader to framed for the official documentation and provide here only a few relevant details, particularly regarding the `titled-frame' environment.

<table>
<thead>
<tr>
<th>A list of the environments from package framed</th>
</tr>
</thead>
<tbody>
<tr>
<td>This list indicates which boxing macros are used in the original, and their replacement.</td>
</tr>
<tr>
<td>\textbf{framed} (\fbox)</td>
</tr>
<tr>
<td>\textbf{oframed} (\CustomFBox)</td>
</tr>
<tr>
<td>\textbf{titled-frame} (\CustomFBox)</td>
</tr>
<tr>
<td>\textbf{shaded} (\colorbox)</td>
</tr>
<tr>
<td>\textbf{shaded*} (\colorbox)</td>
</tr>
</tbody>
</table>
The previous box gives an example of use of `titled-frame’ environment. It is an environment with one mandatory argument which provides the title of the frame, which is repeated after a page break with (cont) appended. The colours TFFrameColor and TFFTitleColor must have been defined by the user. colorframed adds the possibility to colorize separately the continuation label. For example we used
\renewcommand\colorframedTFconlabcolorcommand{\color{purple}}

and the default definition of this macro is to do nothing (and then the symbol obeys the TFFrameColor).

• The current box is an example of `shaded’ environment; The shadecolor must have been defined by user. See the repository issue #5 for some comments about how to use such environment with list items, if the environment is to start at start of the item, as here.

• This is an example of `snugshade’ environment. It shares with `shaded’ the usage of shadecolor.

• This is an example of `snugshade*’.

• The `titled-frame’ environment was in effect broken in recent LATEX which has modified how \smash behaves (the continuation label created a blank line interrupting the framing); colorframed fixes this infelicity in passing.

To customize further usage of `titled-frame’ one must renew its definition, which is left untouched by colorframed. Here is its source from framed.sty: (code and comments by Donald Arseneau)

% A particular type of titled frame with continuation marks.
% Parameter #1 is the title, repeated on each page.
\newenvironment{titled-frame}[1][% 
  \def\FrameCommand{\fboxsep8pt\fboxrule2pt 
  \TitleBarFrame{\textbf{#1}}}% 
  \def\FirstFrameCommand{\fboxsep8pt\fboxrule2pt 
  \TitleBarFrame{\textbf{#1}}}% 
  \def\MidFrameCommand{\fboxsep8pt\fboxrule2pt 
  \TitleBarFrame{\textbf{#1\ (cont)}}}% 
  \def\LastFrameCommand{\fboxsep8pt\fboxrule2pt 
  \TitleBarFrame{\textbf{#1\ (cont)}}}% 
  \MakeFramed{\advance\hsize-20pt \FrameRestore}}% 
% note: 8 + 2 + 8 + 2 = 20. Don’t use \width because the frame title
% could interfere with the width measurement.
{\endMakeFramed}

The continuation label used in the `titled-frame’ environment is $\blacktriangleright$ (which despite its name reacts to current text colour setting). It requires to the best of the author knowledge loading amsymb or some other math symbols package and it is up to user to do it.

• (one can not nest environments of framed.sty, else we would have done so here.)

(Here is an attempt to use the `framed’ environment around a list item; we can see an unexpected induced indentation of the paragraph. And, by the way, there is no `snugframe’
To customize the `titled-frame' output, one may simply redefine the environment via suitable changes e.g. to the setting of \fboxsep and \fboxrule. But doing so, one quickly discovers that the layout from \TitleBarFrame has problems (see on this the repository issues #3 and #4). So colorframed 0.9b provides \colorframedTitleBarFrame, which can be used as a replacement to \TitleBarFrame in a re-defined custom `titled-framed' environment. Check the source code for comments. This alternative obeys added customizing macros whose defaults are shown here:

\def\colorframedTFtitlesep{\fboxsep}% hor. distance of title from left border
\def\colorframedTFconlabsep{3pt}% hor. distance of continuation label from border

This variant handles differently than its model the horizontal and vertical placement of the title. It is provided as an exception to the general rule that this package should take care only of fixing colour problems (addition of new environments may be considered in future, but fixing existing environments will not be done beyond what is described here).

As was already mentioned `titled-frame' had been broken some years ago by an upstream \LaTeX{} change and we did not change it beyond fixing this problem. The alternative \colorframedTitleBarFrame must be opted-for explicitly by user in a re-definition of `titled-frame' environment, the default remaining with the problematic \TitleBarFrame. Or the user can do \let\TitleBarFrame\colorframedTitleBarFrame and use `titled-frame', not a new environment.

Already we have mentioned

\colorframedTFconlabcolorcommand

whose default definition let it expands to nothing at all (the continuation label then inherits the TFFrameColor colour).

Regarding the `framed' environment, colorframed adds the

\colorframedbordercolorcommand

which expands by default to \normalcolor. It influences the colour of the framing done by environment `framed', the original having no such customizability. Do not redefine it to do nothing, it always should set some colour, else the borders may display colour loss after a page break.

For example

\setlength{\FrameRule}{5pt}
\setlength{\FrameSep}{9pt}
\renewcommand{\colorframedbordercolorcommand}{\color{gray!50}}

configures the next usage of `framed'.
3 Technicalities, musings about future

I first developed the analysis of the colour leaks/losses at page breaks with \texttt{framed}
on occasion of some contributions I made to the \texttt{Sphinx project} and I am transferringhere the gained knowledge.

The key thing is that the boxes handled by \texttt{framed}.sty may contain isolated colourpush or colour pop. We must make sure an isolated colour push, if followed by colourchanges, is always followed by paired ones, and never by a colour pop from a colourcommand originated "prior" and symmetrically that a sole colour pop does not followa yet unpaired colour push from some other colour command related to the framing.

This means we can not ever use things such as \texttt{\fbox}, \texttt{\colorbox} or \texttt{\fcolorbox} (whetheror not \texttt{xcolor} is loaded) as they will cause colour pops after a possibly isolatedcolour push contained in the (already boxed) contents which are fetched to them asarguments by \texttt{framed}.sty internals. We must re-implement them otherwise. This iswhat is done here, without touching the \LaTeX{} internal macros themselves.

Some portion of the macro file is occupied with the "titled-frame" environment of\texttt{framed}.sty, which was broken by an upstream \LaTeX{} change related to \texttt{\smash}, and hasobvious problems apart from that. Further "obvious problems" do exist for the other\texttt{framed}.sty environments but it is not the matter of this package to fix them. Per-haps in future another package will be provided based upon the work here and addinga completely new environment fixing the "obvious problems" (some of them being simplylack of customizability).

The author has developed based upon usage of \texttt{pict2e} breakable boxes with round cor-ners, background colour, optional shadow (possibly inset), and other goodies and wasplanning on incorporating this environment into the package at some time in future.

But after initial release made it to CTAN on 2022/09/22 I became aware of \texttt{longibox}which provides already such \texttt{pict2e} breakable boxes with rounded corners (even ellipti-cal arcs), and furthermore with a CSS-like interface which is exactly what I haddoned on my side too... I need to check more \texttt{longibox} before a decision is made here!Perhaps it will be better to keep \texttt{colorframed} as it is currently and extras such as new\texttt{pict2e}-based boxes with a key-value interface to 'inline' or 'display' boxing macrosand environments should make it to another package (loading \texttt{colorframed} of course).

Of course, such a package would remain limited in its graphics abilities (and cus-tomizability) in comparison to the customizable boxing environments provided by pack-age \texttt{mdframed} or \texttt{tcolorbox} but testing with my own \texttt{pict2e}-based boxes showed signifi-cant speed-up in build time compared to using \texttt{framed+tcbox} or the "breakable" optionof \texttt{tcolorbox}.

\begin{center}
\texttt{\fbox}, \texttt{\colorbox} or \texttt{\fcolorbox}
\end{center}

\vspace{1cm}

\vbox to0pt{
\hrulefill
\vfill
\parbox{\textwidth}{
\textbf{End of documentation here}}
\vfill
\hrulefill
\vbox to0pt{