The dynblocks package*

Claudio Fiandrino
claudio.fiandrino@gmail.com

September 18, 2014

Abstract
The dynblocks package allows to fully customize blocks aspect and
dimension inside a presentation.
The package originated from this question in TeX.SE. The core
functionalities of the package are based on this answer.

Contents
1 Introduction 1
2 Usage 2
  2.1 Basic usage 2
  2.2 Change text alignment 4
  2.3 Text opacity and word alert 5
3 Options and advanced examples 7
  3.1 The shadow and the rounded corners 7
  3.2 Customized fill colors 8
  3.3 Color adaptation to the Beamer theme 10

1 Introduction
The purpose of the package is to provide an instrument to customize several
aspects of blocks (here called dynblocks):

- the width;
- the color

---

*This package has version number v0.2b of 187/09/2014; it is released under and subject
to the \LaTeX Project Public License (LPPL).
● of the background;
● of the border;

● the text:
   ○ alignment;
   ○ opacity

Notice that dynblocks defined by dynblocks differ from usual beamer’s blocks because no title is given.

The package has the following requirements:

● TikZ;
● etoolbox;
● xparse.

I would like to thank Enrico Gregorio for having pointed out an issue after the release of TikZ 3.0.0 and xparse version 2014/08/25.

2 Usage

To load the package use as usual: \usepackage[⟨options⟩]{dynblocks}.

The different options that can be adopted will be analysed in detail in section 3.

2.1 Basic usage

Using the package in basic mode allows to define a block with:

● justified alignment;
● width equal to \textwidth;
● border color \textcolor{blue} and fill color \textcolor{blue!10}.

thanks to the command \opaqueblock⟨⟨overlay spec⟩⟩[⟨width⟩]⟨⟨text⟩⟩. Notice that ⟨⟨overlay spec⟩⟩ could be:

● a single number: <1>;
● multiple numbers separated by commas and delimited by braces: ⟨{1,2,3}>;
● a single number followed by a dash: <1->.
Moreover, it is also possible to make it *invisible*, that is to force colors to become gray by means of \invblock⟨overlay spec⟩. For example, the following code, generates the two frames shown in figures 1a and 1b.

\documentclass{beamer}
\usepackage{dynblocks}
\usetheme{Luebeck}
\begin{document}
\begin{frame}{The frame title}
\begin{columns}[T]
\begin{column}{0.4\textwidth}
\begin{dynblock}
\opaqueblock<1>[0.8\textwidth]{hello this is a dynamic block with an itemize environment:
\begin{itemize}
\item hello
\item hello again
\end{itemize}}
\end{dynblock}
\end{column}
\begin{column}{0.4\textwidth}
\begin{dynblock}
\opaqueblock<2>{hello this is another dynamic block}
\end{dynblock}
\end{column}
\end{columns}
\end{frame}
\end{document}

The frame title
hello this is a dynamic block with an itemize environment:
hello
hello again
hello this is another dynamic block

(a) First frame
(b) Second frame

*Figure 1*: The basic example

In this example, it is possible to notice that the second \opaqueblock has no specified width; the default value is \textwidth, but if the block
is placed inside a `column` environment it automatically inherits the width given there. To set different values of width, it is necessary to specify the optional argument as did for the first `opaqueblock` of the example.

The presence of an `invblock` makes the first `opaqueblock` invisible; this command needs to be placed immediately after an `opaqueblock` because of two facts:

- it automatically inherits the width from the `opaqueblock`;
- it automatically shows the text of the previous `opaqueblock`.

Finally, both `opaqueblock` and `invblock` need to be placed inside a `dynblock` environment.

### 2.2 Change text alignment

Thanks to the command `\setalignment{⟨text spec⟩}` the text alignment changes according to the ⟨text spec⟩ (for all possibilities please refer to the pgfmanual section 17.4.3 - version December 20, 2013).

Suppose, for instance, to modify the previous example in order to display the second block with center aligned text. The only change to do is:

```latex
\begin{dynblock}
\setalignment{center}
\opaqueblock<2>{hello this is another dynamic block}
\end{dynblock}
```

and this will lead to figure 2.

![Frame Title](The frame title)

![Dynamic Block](hello this is a dynamic block with an itemize environment:
- hello
- hello again)

![Dynamic Block](hello this is another dynamic block)

**Figure 2:** Second frame with text center aligned for the second block
Notice that the alignments can be local or global: if there is a definition in the preamble (global), all the \texttt{dynblocks} will be set according to this definition. When the definition is set inside a group (a \texttt{columns} environment or even simpler, the \texttt{dynblock} environment), it affects only locally the \texttt{dynblocks}. It is even possible set a global definition and then make local changes.

2.3 Text opacity and word alert

By default:

- \texttt{opaqueblocks} have an opacity set to 0.9;
- \texttt{invblocks} have an opacity set to 0.4.

To change these values two commands have been introduced:

- \texttt{\setvisopacity{\langle opacity spec\rangle}};
- \texttt{\setinvopacity{\langle opacity spec\rangle}};

where \texttt{\langle opacity spec\rangle} is a value in the interval \([0,1]\). Also these commands can be set locally or in a global fashion in the preamble.

Due to the opacity, the usual \texttt{\alert} command is not more useful with \texttt{invblocks}. The package provides a method to alert a word even in this case. The command to be used is \texttt{\dynalert{\langle overlay spec\rangle}}\{\langle text\rangle\}. Note that with the proper usage of \texttt{(overlay spec)}, \texttt{\dynalert} must not fall inside a \texttt{opaqueblock}. This is a limitation because the purpose for which it has been developed is different.

Assume, for example, to modify the reference example such that the opacity of \texttt{invblock} will be set to 0.1; furthermore, for the first block, the word “itemize” will be alerted with \texttt{\alert}, while “dynamic block” with \texttt{\dynalert} and for the second block (that does not have the correspondent \texttt{invblock}) it is shown what happens with a wrong usage of \texttt{\dynalert}. The code is:

\begin{verbatim}
\documentclass { beamer }
\usepackage { dynblocks }
\usetheme { Luebeck }
\setinvopacity {0.1}

\begin{document}
\begin{frame}{ The frame title }
\begin{columns}[[T]
\begin{column}{0.4 \textwidth}
\begin{dynblock}
\opaqueblock <1>[ 0.8 \textwidth ]{ hello this is a
\dynalert <2>{ dynamic block } with an \alert <1,2>{ itemize}
\end{dynblock}
\end{column}
\begin{dynblock}
\opaqueblock <1>[0.8 \textwidth ]{ hello this is a
\dynalert <2>{ dynamic block } with an \alert <1,2>{ itemize}
\end{dynblock}
\end{columns}
\end{frame}
\end{document}
\end{verbatim}
\begin{itemize}
  \item hello
  \item hello again
\end{itemize}

\begin{dynblock}
\opaqueblock<2>{hello this is another
dynamic block}
\end{dynblock}

The frame title
hello this is a dynamic block with an itemize environment:
hello
hello again

The frame title
hello this is another dynamic block

\begin{frame}
\begin{columns}
\begin{column}{0.4\textwidth}
\begin{dynblock}
\opaqueblock<2>{hello this is another
dynamic block}
\end{dynblock}
\end{column}
\end{columns}
\end{frame}

Figure 3: Example with different opacity and alerts

As it is possible to see from figures 3a and 3b, the usual \texttt{\alert}, when used inside an \texttt{\invblock}, is set with the opacity of the block while the proper \texttt{\dynalert} no. Anyway, a wrong usage of \texttt{\dynalert} lead to the output shown in figure 3b: the subsequent text of the alerted word is set with the opacity of an \texttt{\invblock}.

The suggested use, in conclusion, is:

- \texttt{\alert} with \texttt{⟨overlay spec⟩} equal specified in the related \texttt{\opaqueblock};
- \texttt{\dynalert} to highlight words inside an \texttt{\invblock};
- never do something like: \texttt{\dynalert<1,2>{word}} if the \texttt{\opaqueblock<2>{word}} is shown in \texttt{⟨overlay spec⟩} = 1 and the \texttt{\invblock} in \texttt{⟨overlay spec⟩} = 2.

To change colors:
• for \alert the usual Beamer command works:
  \setbeamercolor{alerted text}{⟨color spec⟩};

• for \dynalert a different command has been introduced to differentiate them from standard Beamer’s alerts: \setwordscolor{⟨color⟩}; the default value is set to blue.

3 Options and advanced examples

In this section the package’s options are introduced with examples. They allow to customize more deeply the aspect of dynblocks:

• adding the shadow and the rounded corners (subsection 3.1);
• customizing the fill color (subsection 3.2);
• adapting the fill color to the current Beamer theme used (subsection 3.3).

3.1 The shadow and the rounded corners

To load:

• the shadow option use: \usepackage[shadow]{dynblocks}; it is possible to set the shadow opacity by means of the following command: \setshadowopacity{⟨opacity spec⟩} (default value 0.4);

• the option to have rounded corners for dynblocks use: \usepackage[roundedcorners]{dynblocks}.

For example:

\documentclass{beamer}
\usepackage[shadow,roundedcorners]{dynblocks}
\usetheme{Luebeck}
\begin{document}
\begin{frame}{The frame title}
\begin{columns}[T]
\begin{column}{0.4\textwidth}
\begin{dynblock}
\opaqueblock<1>[0.8\textwidth]{hello this is a}
\dynalert<2>{dynamic block} with an
\alert<1,2>{itemize} environment:
\begin{itemize}
\item hello
\item hello again
\end{itemize}
\end{dynblock}
\end{column}
\end{columns}
\end{frame}
\end{document}
allows to get the frames shown in figures 4a and 4b.

3.2 Customized fill colors

By activating this option it is possible to fully customize the dynblocks colors because several command become available:

- \setblockcolor{⟨color spec⟩} and \setbordercolor{⟨color spec⟩} for the \opaqueblocks (default values are blue!10 and blue respectively);
- \setinnercolor{⟨color spec⟩} and \setoutercolor{⟨color spec⟩} for the \fancyblocks (default values are white and blue!10 respectively);
- \settopcolor{⟨color spec⟩} and \setbottomcolor{⟨color spec⟩} for the \vshadeblocks (default values are white and blue!10 respectively);
- \setleftcolor{⟨color spec⟩} and \setrightcolor{⟨color spec⟩} for the \oshadeblocks (default values are white and blue!10 respectively).
Similarly to the shadow and the roundedcorners options, to load the customcolors option use \usepackage\{customcolors\}\{dynblocks\}.

In the following example, all dynblocks types are used and it is possible to see how local and global setting work.

\documentclass\{beamer\}
\usepackage\{shadow, roundedcorners, customcolors\}\{dynblocks\}
\setblockcolor\{red!10\}
\setbordercolor\{red\}
\setbottomcolor\{orange!40\}
\setrightcolor\{orange!40\}
\usetheme\{Luebeck\}
\begin\{document\}
\begin\{frame\}{The frame title}
\begin\{columns\}[T]
\begin\{column\}{0.4 \textwidth}
\begin\{dynblock\}
\opaqueblock\{ 0.8 \textwidth \}{ hello this is a \dynalert\{ dynamic block \} with an \alert\{ itemize \} environment: \begin\{itemize\} \item hello \item hello again \end\{itemize\} \}
\invblock\{2-\}
\end\{dynblock\}
\end\{column\}
\begin\{column\}{0.4 \textwidth}
\setalignment\{center\}
\begin\{dynblock\}
% default settings since no \setinnercolor or \setoutercolor % are there \fancyblock\{ hello this is another dynamic block \}
\invblock\{3-\}
\end\{dynblock\}
\\[2ex\]
\setbordercolor\{orange\} % local definition % that overwrites the global one \begin\{dynblock\}
\vshadeblock\{replica: hello this a another dynamic block\}
\invblock\{4-\}
\end\{dynblock\}
\\[2ex\]
\begin\{dynblock\}
\oshadeblock\{replica 2: hello this a another dynamic block\}
\end\{dynblock\}
\end\{columns\}
\end\{frame\}
\end\{document\}
The four frames obtained by this example are shown in figures 5a, 5b, 5c and 5d; more in detail:

- an example of `\opaqueblock` customization through `\setblockcolor` and `\setbordercolor` could be seen in figure 5a;
- an example of `\fancyblock` with default settings could be seen in figure 5b (notice that it inherits the `bordercolor` from the global setting);
- an example of `\vshadeblock` with customization of `\setbottomcolor` and locally `\setbordercolor` (`\settopcolor` at default value) could be seen in figure 5c;
- finally, an example of `\oshadeblock` with `\setrightcolor` and local `\setbordercolor` customization (`\settopcolor` at default value) could be seen in figure 5d.

### 3.3 Color adaptation to the Beamer theme

The purpose of this option is to use the Beamer’s color of the theme currently adopted; as it will be possible to see, the `getthemecolors` option should be used with particular care. To load the option there is the usual `\usepackage[getthemecolors]{dynblocks}`.

This option is defined inside the package as:

```latex
\DeclaoreOption{getthemecolors}{
% redefinition opaqueblock
\renewcommand{\thecol}{structure fg!10}
\renewcommand{\thebordercol}{structure fg}
% redefinition fancyblock
\def@setinnercolor{white}
\def@setoutercolor{structure fg!10}
% redefinition vshadeblock
\def@settopcolor{white}
\def@setbottomcolor{structure fg!10}
% redefinition oshadeblock
\def@setleftcolor{white}
\def@setrightcolor{structure fg!10}
}
```

thus it works properly if the current beamercolortheme set the `structure` definition.

For example:
Figure 5: Example with customcolor option and all \textit{dynblocks} types

\documentclass{beamer}
\usepackage[shadow, roundedcorners, getthemecolors, customcolors]{dynblocks}
\usetheme{CambridgeUS}
\begin{document}
\begin{frame}{The frame title}
\begin{columns}[T]
\begin{column}{0.4 \textwidth}
\begin{dynblock}
\opaqueblock<1>[0.8 \textwidth]{hello this is a dynamic block with an \textit{itemize} environment:
\begin{itemize}
\item hello
\item hello again
\end{itemize}}
\invblock<2->
\end{dynblock}
\end{column}
\end{columns}
\end{frame}
\end{document}
will lead to figures 6a and 6b.

\begin{figure}[h]
\centering
\begin{tabular}{cc}
\begin{minipage}[t]{0.4\textwidth}
\begin{dynblock}
\opaqueblock<2>{hello this is another dynamic block}
\end{dynblock}
\end{minipage} & \begin{minipage}[t]{0.4\textwidth}
\begin{dynblock}
\opaqueblock<2>{hello this is another dynamic block}
\end{dynblock}
\end{minipage}
\end{tabular}
\caption{(a) First frame (b) Second frame}
\end{figure}

\textbf{Figure 6:} Example with a theme that does not define structure color

The result can be improved in the following way:

\documentclass[beamer]
\usepackage[shadow, roundedcorners, getthemecolors, customcolors]{dynblocks}
\usetheme{CambridgeUS}
% definition of structure
\setbeamercolor*{structure}{parent=palette primary}

\begin{frame}{The frame title}
\begin{columns}[T]
\begin{column}{0.4\textwidth}
\begin{dynblock}
\opaqueblock<1>[0.8\textwidth]{hello this is a dynamic block with an itemize environment:}
\begin{itemize}
  \item hello
  \item hello again
\end{itemize}
\end{dynblock}
\end{column}
\end{columns}
\end{frame}
obtaining as result the frames shown in figures 7a and 7b.

Figure 7: Example of a theme with a posteriori structure color definition

Here is another example:
The decision of adopting *structure* as reference is due to the fact that this parameter is one of the most relevant while customizing a Beamer theme. In the following example, it is shown a color customization of the Szeged theme and a particular effect that can be realized thanks to multiple *dynblocks* inside the same dynblock environment:

```latex
\documentclass{beamer}
\usepackage[getthemecolors,roundedcorners,shadow]{dynblocks}
\usetheme{Szeged}
\setbeamercolor{structure}{bg=red!20,fg=red}
\begin{document}
\begin{frame}{A title}
\begin{center}
\begin{dynblock}
\opaqueblock<1>[0.6\textwidth]{hello this is a dynamic block with an itemize environment:
\begin{itemize}
\item hello
\item hello again
\end{itemize}}
\invblock<2->
\setalignment{center}
\opaqueblock<2>{hello this is another dynamic block}
\end{dynblock}
\end{center}
\end{frame}
\end{document}
```

**Figure 8:** Second example of a theme with a posteriori structure color definition
hello this is a dynamic block with an itemize environment:
▶ hello
▶ hello again

hello this is another dynamic block

(a) First frame

(b) Second frame

Figure 9: Example of a customized theme

\end{dynblock}
\end{center}
\end{frame}
\end{document}

The two frames obtained as outcome are shown in figures 9a and 9b.

Note that the getthecolors option has some drawbacks when it is used with particular Beamer color themes like:

- albatross;
- beetle.