Onera/DTIM

Documentation of the `acmebeamer` package.

A presentation like/showcase document.

Table of contents

- **Package options**
- **Beamer features**
  - Title page
  - Table of contents
  - Page numbers
  - Head- and footline
  - Frametitle
- **Extra features**
  - Better `itemize` environment
  - Blocks on steroids
- **One more thing . . .**

Package options

`portrait` Swaps the height and width of the document.

Title page

Usual and basic way

`\maketitle` Modified to typeset the macro `\inserttitlepage` inside a `frame` environment.

`\titlegraphic` Sets the value of `\inserttitlegraphic`.

`\setbeamertemplate` Define the template to use for the title page. Same as `\titlegraphic` in this context.

Going a bit further

- `\maketitle` actually takes **two optional arguments**. The first one is passed to `\setuptitlepage` (see below), the second one is passed to the frame options. If you want to give the second argument without the first one, you must give an explictly empty first argument.

  Use as `\maketitle[<options to \setuptitlepage[...]>][<options to \begin{frame}[...]>]`.

- The package provides a way to setup the title page with the macro `\setuptitlepage[...]`, according to a few (for now) options of the default template. There are three options as of now, `lineoffset`, `logo` and `customlogo`.
  - `lineoffset`: sets the height of the line in the default template.
  - `logo`: two values only (for now), `none` and `institute`. One way guess what they do.
  - `customlogo`: sets its value to be the logo. The last one wins.

Table of contents

`\frametoc` Takes one mandatory argument (in braces) **and then**, an optional argument (in brackets). The mandatory argument is the title of the frame. The optional argument is passed to `\tableofcontents`.

Use as `\frametoc{<frame title>}[<options to \tableofcontents[...]>]`.

Page numbers

`\setuppagenumbers`

Takes one mandatory argument in square brackets, a key value list consisting of two possible keys: `style` and `custom`. The latter sets the page numbers command to typeset its argument. There are three values possible (for now) for the `style` key: `none`, `boxed` and `plain`. The last two typeset the pages as *<page number>/<total number of pages>*.

Use as `\setuppagenumbers[...]`.

Head- and footline

The `acmebeamer` package provides a flexible way to customize the head- and footline through two commands `\setupheadline` and `\setupfootline` described below.

`\setupheadline`
`\setupfootline`

Takes one mandatory argument in square brackets, a key value list. There are four key available: `left`, `center`, `right` and `frame`. The first three customize the respective places and `frame` is used to customize the width and the margins.

You can set those with `\setupheadline[frame={width=.8\hsize, margin=3pt, topmargin=5pt}]` for example. Available are `width`, `margin`, `leftmargin`, `rightmargin`, `topmargin` and `bottommargin`.

The other options have the same behaviour and options. The values start being `empty`.

- Standard options from `beamer`: `title`, `subtitle`, `author`, `date`, `section` and `section`.
- New options, the basic way: `page numbers` and `empty`.
- New options, the tricky way: don't use `right=...`, but `right/subsection=...` with the following options available: `text` (same as doing `right=subsection`), `bullets` and `squares`. At some point, the package may offer a direct solution.

The present document uses `\setupheadline[left=title, right=page numbers]` `\setupfootline[]`.

This particular slide uses `\setupheadline[left=section, center/subsection=bullets]` `\setupfootline[left=subtitle, center=author, right=date]`.

One can notice that the short forms are used for the standard options and that previous setups are not erased (see the page numbers on the right). If all `right`, `center` and `left` are empty, the head- or footline doesn't occupy any space.

Frametitle

Great feature added!

`template` frametitle

Has been modified so that if the frame title is empty, then the template is not used. Besides, the subtitle always occupies space if the template is used (may change in the future).

```
\begin{frame}
...
\end{frame}
```

Has been modified to be able to omit the title if one wants to use the same as the current section or subsection. Here are the rules:

- No title provided:
  - uses the subsection (`\subsecname`) if not empty;
  - or uses the section (`\secname`) if not empty;
  - or uses the template with an empty title (if a subtitle is provided, it will be typeset).
- A title has been provided: normal behaviour. As a remainder, one can specify a title and a subtitle according to the following:
  - as the first and second **braced group** if nothing happens before the groups except for spaces and comments `\begin{frame} {<title>} {<subtitle>}`;
  - as arguments to the commands `\frametitle` and `\framesubtitle` inside the `frame` environment.

  How can we add a subtitle to a frame that has the same title as the (sub)section? This slide uses `\begin{frame} \framesubtitle{...}`.

  There is a small catch however: you must use `\maketitle` or you can use `\acmeframetitlehack` before the frames you want to use this feature on. All the following frames will use the feature.

Better `itemize` environment

```
\begin{sitemize}
...
\end{sitemize}
```

Sometimes, we would like to list a bunch of things, but we do not want to start at level 1. This is the *raison d’être* of this environment. You can specify the level in an optional argument: allowed values are `1`, `2` and `3`. The values are **relative to the current nesting level**. The default level is `1` for compatibility reasons.

Example

1. Text goes here.
   - one-one (third level)
   - one-two (third level)
2. Text goes here.
   - two-one (second level)
   - two-two (second level)

     a a

     b b
   - two-three (second level)

Typeset with

```
\begin{enumerate}[<+->] % remove 'handout' to see
\item Text goes here.
  \begin{sitemize}
  \item one-one (third level)
  \item one-two (third level)
  \end{sitemize}
\item Text goes here.
  \begin{sitemize}
  \item two-one (second level)
  \item two-two (second level)
    \begin{enumerate}[a]
    \item a
    \item b
    \end{enumerate}
  \item two-three (second level)
  \end{sitemize}
\end{enumerate}
```

Blocks on steroids (1/4)

! Blocks are completely rewritten from scratch. They are still accessible with the same commands, but the arguments are somewhat different and are given differently too.

#1 the arguments

The beamer blocks take only one optional parameter **in braces** which is the title of the block, if any. Now, there is two optional arguments, one in braces, the title, **and then** one in square brackets. The latter are options destined to the new system of blocks.

`\begin{blockname} {<title>} [<options>]... \end{blockname}`.

#2 basic options

`width` Sets the width of the block.

`color` Sets the color of the block.

`align` Sets the alignment of the block, not the text inside the block (see below).

`bodyheight` Sets the height of the blockbody.

`title/bodycolor` Sets the color of the title/body, this can be a beamer color or just a color (like “yellow”).

`title/bodystyle` Sets the style of the title/body (like “`\bfseries`” or “`\slshape\large\color{green}`”).

`title/bodyalign` Sets the alignment of the text inside. Possible values are: `default`, `flushleft`, `flushright` and `center` horizontaly and `top`, `bottom` and `middle` vertically. You can give several (typically two) as a list in braces: `bodyalign={center,top}`. Better avoid spaces.

#3 advanced options

`title-/bodycustomframe`
`title-/bodybackground` Sets the frame Ti*k*Z path. You can disable the frame with `false` or anything that will be ignored in a `tikzpicture` environment. So `true` alos works, but is confusing. The same goes for the background. (Defaults on third next slide.)

`title-/bodyoptions` Options passed to the `framedtext` environment. (See next section.)

Blocks on steroids (2/4)

Examples

With a yellow title

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis accumsan varius nisl placerat volutpat.

```
\begin{block} {With a yellow title}
  [titlecolor=yellow]
  \smalllorem
\end{block}
```

Without title, bold centered body text.

```
\begin{alertblock}
  [bodyalign=center, bodystyle=\bfseries]
  Without title, bold centered body text.
\end{alertblock}
```

Left aligned title

Fixed body height to 20pt, bottom right aligned.

```
\begin{exampleblock} {Left aligned title}
  [titlealign=flushleft, bodyheight=20pt,
   bodyalign={bottom,flushright}]
  Fixed body height to 20pt, bottom and right aligned.
\end{exampleblock}
```

A new one, `emphasisblock`

- one
- two
- three

```
\begin{emphasisblock} {A new one, emphasisblock}
  \begin{sitemize}
  \item one
  \item two
  \item three
  \end{sitemize}
\end{emphasisblock}
```

Blocks on steroids (3/4)

The case of lists and enumerations

The use of lists in blocks is somewhat screwed up if there is text around. The package offers a dirty solution in the form of a macro to use between the text and the `\(s)itemize/enumerate` environments: `\fixvspace`.

Before text: `\fixvspace` macro between text and environment.

- one
- two

After text: no `\fixvspace` macro between text and environment.

- one
- two

Blocks on steroids (4/4)

Default setup

`\defineblock[...][...]`
`\defineblocks[...][...]`
Defines a block or several at once, either from an existing block, or a setup.

`\setupblock[...][...]`
`\setupblocks[...][...]`
`\setupblocks[...]`
Setups a block, several blocks at once or setup the default for all blocks. Unless specified otherwise, blocks inherit their parameters from `blocks`. If defined from another block, they inherit from it. All parameter set explicitely is specific to the block.

```
  \defineblock
    [default] [color=structure]
  \defineblock
    [alert] [color=alerted text]
  \defineblock
    [example] [color=example text]
  \defineblock
    [emphasis] [color=emphasis]

  % color of the bullets
  \setbeamercolor {defaultblockitems}
    {parent=structure}
  \setbeamercolor {alertblockitems}
    {parent=alerted text}
  \setbeamercolor {exampleblockitems}
    {parent=example text}
  \setbeamercolor {emphasisblockitems}
    {parent=emphasis}

  % map defaultblock to LaTeX block
  \let\defaultblock\block
  \let\defaultendblock\endblock

% example
  \defineblock
    [newexpample] [example]
  \defineblocks
    [one,two, three] [default]
  \defineblocks
    [four,five,six] [width=.5\hsize]
```

```
\setupblocks[% no spaces after [ or before commas
  width=\hsize,
  align=center,
  customframe=
   {\path [draw] (0,0) rectangle ++(\framedboxwd,-\framedboxht);},
  custombackground=
   {\path [fill] (0,0) rectangle ++(\framedboxwd,-\framedboxht);},
  frameoptions=
   {offset=0pt,
   tikzoptions={draw=fg, fill=bg, thick, rounded corners=2pt}},
  titlecustomframe=none,
  titlecustombackground=
   {\path [fill=fg]
   {[sharp corners] (0,0) -- ++(0,-\framedboxht) -- ++(\framedboxwd,0)}
   {[rounded corners=2pt] -- ++(0,\framedboxht) -- cycle};},
  titleoptions={offset=4pt, bottomoffset=2pt, left=\strut},
  titlecolor=white,
  titlestyle=\usebeamerfont{block title},
  titlealign={center},
  bodycustomframe=none,
  bodycustombackground=false,
  bodyoptions={offset=4pt},
  bodycolor=normal text,
  bodystyle=\usebeamerfont{block body},
  bodyheight=-\maxdimen,
  bodyalign={default, middle}]

% example
\setupblocks[default,alert][align=left]
```

One more thing . . . (1/2)

The `framed` command and the `framedtext` environment

The `acmebeamer` package relies on a package named `acmetoolbox` which provides (among other things) facilities to frame text similarly to packages like `bclogo` or `fancybox`. These facilities come as two commands for short snippets and an environment. They rely on Ti*k*Z to do the framing.

`\framed` Frames a text with a `\hbox`. It takes one optional argument in square brackets and one mandatory argument which is the text to frame. Use as `\framed[<options>]{...}`.

`\inframe` Ditto as `\framed`, except the baseline of the text inside is the same as the text outside.

```
\beging{framedtext}
...
\end{framedtext}
```

Frames a text with a `\vbox`. It takes one optional argument in square brackets.

You actually already saw some framed text: the notice on slide 13. This was an example of inframed text.

There is yet another (convenient) way to highlight text in paragraphs and formulas.

`\hl` Highlights text. This command is overlay-aware! `\hl[<tikz options>]{...}`

`\placehlmark` Place a named mark. The name is optional and is specified in square brackets. The mandatory argument is placed in a Ti*k*Z node named after the name supplied (`mark` by default). `\placehlmark[<name of mark>]{<text to be marked>}`

`\hlmarks` Used to highlight several marks. This command is overlay-aware too! An optional argument can be supplied in square brackets to indicate the shape to use to hightlight the marks, this is `ellipse` by default. (See the source to understand how it is used.)

`\hlmarks[<tikz options>]{(some) (marks) (to) (ellipse)}`

One more thing . . . (2/2)

The available options

`left, right, top, bottom` Put something at the left, right, top and bottom respectively arround the text (which will be boxed by that time).

`left/right/top/bottom/frame` Take values `on` and `off` to activate/deactivate the left, right, top, bottom or the whole frame. The code is clever enough to use the whole frame if all sides are `on`.

`left/right/top/bottom/margin` Sets the margins around the frame (in addition to the width).

`left/right/top/bottom/offset` Sets the offsets inside the frame (part of the width).

`width, minwidth, maxwidth` Sets the width, minimum or maximum width of the resulting frame.

`height, minheight, maxheight` Sets the height, minimum or maximum height of the resulting frame.

`align` Defines the alignment of the text in the frame. (See title/bodyalign in previous slides.)

`background` Takes values `on` and `off` to activate or deactivate the background.

`customframe` Custom Ti*k*Z path to be used to draw the frame.

Default is `\path [draw] (0,0) rectangle ++(\framedboxwd,-\framedboxht);`

`custombackground` Custom Ti*k*Z path to be used to draw the background.

Default is `\path [fill] (0,0) rectangle ++(\framedboxwd,-\framedboxht);`

`tikzoptions` Options passed to the `tikzpicture` environment used to draw the frame.