

Creating PDF_AT_EX Presentations*

Reinhard Furrer

University of Zurich, Switzerland

Reinhard.Furrer@math.uzh.ch

2012/06/01

Abstract

The `pfuef` package provides a set of macros for slides generated with PDF_AT_EX. The package handles background images and has different print options.

Contents

1	Introduction	2
2	The <code>pfuef</code> package	2
2.1	Package Installation and Contents	2
2.2	Package Options	3
2.3	Package Commands	4
3	Examples	7
4	Tips and Tricks for You and Me	8
4.1	Placing objects	8
4.2	Transparent Images	8
4.3	Print or Post Presentations	10
4.4	Animated Graphics	10
4.5	R and Sweave	11
5	Tips and Tricks for T_EXnicians	12
6	Frequently asked questions	14
7	Disclaimer	15
8	Release and Revision Details	15

*This file (`pfuefmanual.pdf`) is for version number v2.3 (or for v.2.4.uzh, last revised 2012/06/01).

1 Introduction

The `pfuef` package is a simple, rather customized set of macros for PDF \LaTeX to generate slides. The background motivation to write another slide package was rather simple: I could not find any satisfying macros which could be used with PDF \LaTeX . `Prosper` was rather close to a satisfactory layout, however it was not compatible with PDF \LaTeX . Some friends brought in a few PowerPoint[®] ideas that I tried to implement. Thus `pfuef` (“p-five”, as ‘p-füf’ - pronounced ‘pe.fyf’ - is ‘five’ in my native language) stands for

Prosper and **P**ower**P**oint[®] ideas for **P**DF \LaTeX to create nice **P**resentations

I have tried to be as independent as possible from other packages, since I wanted to have absolute control of all the layout elements. Therefore, what to some might seem bugs or missing features is intentional. But naturally, there are definitely “true” bugs in the code (please report).

The raw structure seems quite simple and can be extended or modified straightforwardly. The font definitions have been taken from `slides.cls`.

Of course, everything is written under the GNU General Public License as stated in <http://www.gnu.org/licenses/gpl.html>.

2 The `pfuef` package

There exist two versions of `pfuef`, a full version (odd minor numbers) and a lite version, tailored to be used at my home university (even minor number). The lite version has no examples and only a limited set of possible background images and slightly different default values.

2.1 Package Installation and Contents

The most recent version of `pfuef` and its supporting material can be downloaded from <http://user.math.uzh.ch/furrer/research/software.shtml>. Copy the file `pfuef.v2.3.tar` into a convenient place and untar it with the command `tar xvf pfuef.v2.3.tar`. Depending on your gusto, you may want `pfuef` to be part of the \LaTeX distribution. In such a case, a few additional steps are required nicely summarized in http://en.wikibooks.org/wiki/LaTeX/Packages/Installing_Extra_Packages.

`pfuef.cls`
`instructions.txt`
`pfuefmanual.pdf`
`history.txt`

The directory `pfuef` contains several files. The file `pfuef.cls` is the class file of this package. The ASCII file `instructions.txt` contains instructions how to use the package, a condensed version of this document — *i.e.* `pfuefmanual.pdf` — which is also part of the distribution. The ASCII file `history.txt` contains a complete history list, known bugs and a to-do list for future releases.

`./p5figures/`

Under certain circumstances (no fancy slide layouts) the class file `pfuef.cls` is sufficient to create a slide document. However, some of the predefined layouts require various images, available in directory `./p5figures/`.

`./p5figures/rgb.tex`

The file `rgb.tex` contains 550 different predefined colors. It is not required, but if the file exists, it is automatically read and you can access all its colors with `\textcolor{MediumOrchid3}{this is my favorite color}`.

- `./p5figures/standardmacros.sty` The file `standardmacros.sty` contains about 250 useful L^AT_EX commands to simplify mathematical typesetting. For example, `\a`, `\bSigma`, `\itGamma`, `\bepsilon`, `\IR`, `\cN` produce \mathbf{a} , Σ , Γ , ϵ , \mathbb{R} , \mathcal{N} . For most other letters, the construct is similar.
- `./example/Makefile` The directory `./example/` contains a few simple example files on how to use `pfuef`. The `Makefile` can be used to compile all files. There are also two shell scripts for animated graphics (see also Section 4.4).
- `./compile.sh` A shell script that facilitates the creation of slides and print (see also Section 4.3).

2.2 Package Options

There are six principal package options.

- `bgcolor` The default format. A slide consists of a predefined background color, a header and a footer, as well as the specific slide text.
- `bgimage` With this package option the user can specify a background image (which is in `./p5figures`) for the slides using the command `\background{backgroundimage}` (accepting all supported `graphicx` formats). This option is not active with `print`.
- `print` The recommended option for printing the slides. It switches to white background color and dark text (ink favorable).
- `ppremarks` Only usable together with the option `print`, the even numbered pages are reserved for the possibility of adding remarks or lines on every even page for printing in handout style (as with PowerPoint[®]). This option is not active with `bgcolor` or `bgimage`.
- `43ratio` At present, the only implemented orientation is landscape. The option `43ratio` defines a page format follows a 4×3 ratio, optimized for projectors or screens with resolutions of 640×480 , 1024×768 , *etc.* The dimension of a `43ratio` page are $210\text{mm} \times 280\text{mm}$, *i.e.* the width of an A4 page and the height of a letter page. The standard paper formats `a4paper`, `a5paper`, `b5paper`, `letterpaper`, `legalpaper` and `executivepaper` are defined as well.
- `gridin` Overlays a grid of thin lines spaced in inches or centimeters. This option can be used to better scale and place figures.
- `gridcm`
- `fields` A specific set of default values that define the layout for presentations of the R-packages 'fields' and 'spam'.
- `uzh` A specific set of default values that define the layout for presentations at the University of Zurich. Note that the slides are not exactly "UniCMS" compatible. The lite version contains essentially only relevant files for this option.
- `cfg` Creates the possibility to specify a set of personalized default values in files `./p5figures/p5style.cfg` and `./p5style.cfg`. The files are – if they exist – read in that order and thus can be seen as global or local options. The content of the files will override the default values. It is possible to include any other relevant L^AT_EX source code. The option `cfg` can be used with options `uzh` or `fields`.

Figure 1 gives the default setup of the layout options as well as an example with a configuration file. The file `p5style.cfg` has the following content:

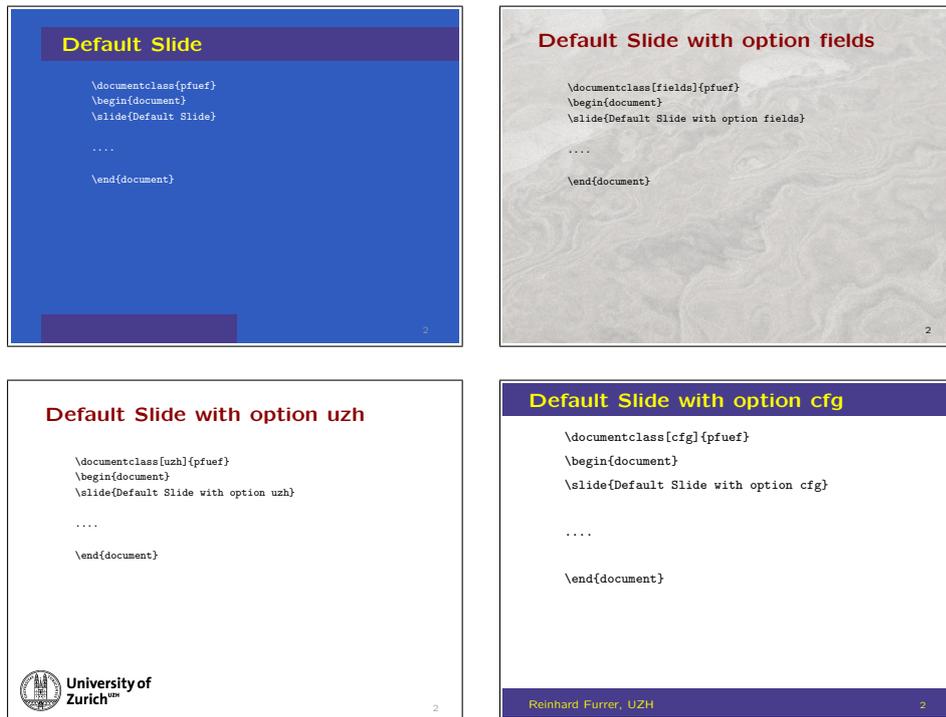


Figure 1: Default settings for slides and a particular configuration setup with a configuration file.

```

% file p5style.cfg
\settextmargins{20mm}{16mm}{16mm}{6mm}{2mm}
\sethfmargins{0mm}{0mm}{21mm}{18mm}{280mm}
\textfontsize{\large}
\footertext{Reinhard Furrer, UZH}
\footertextcolor{yellow}
\slidebackgroundcolor{white}
\setcolortc[black]{black}
\setcolorbc[black]{black}
\setcolorec[DarkBlue]{DarkBlue}
% eof p5style.cfg

```

2.3 Package Commands

- `\slide` The main command is `\slide` and takes the following forms:
- `\slide{}` defines a new slide with no title text. The following text until the next `\slide` command is processed on the specific slide.
 - `\slide{Titletext}` defines a new slide with the title `Titletext`. The title may contain several lines (although this may require a different header definition, see below).
 - `\slide<*>{Titletext}` the slide is only active within the options `bgcolor` and `bimage`, *i.e.* do not consider this page in print option. This option is useful in pseudo overlay mode.

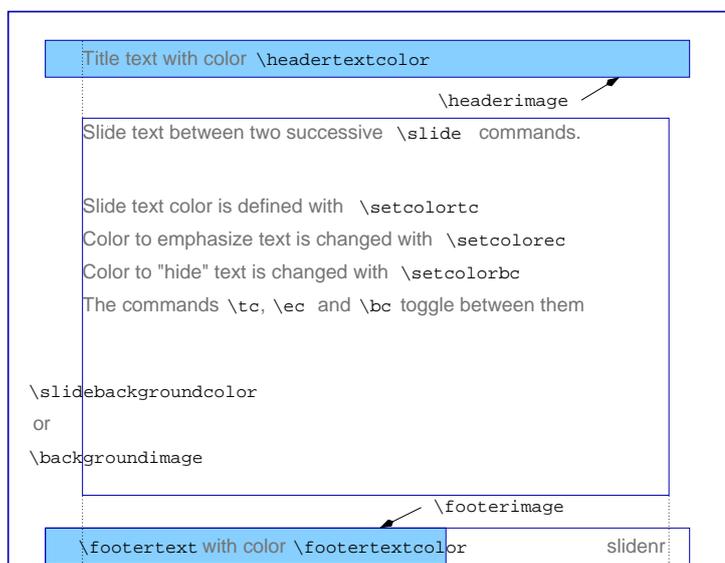


Figure 2: Slide layout, color and images.

`\slide<remarktext><title>` within the option `ppremarks`, print `remarktext` on the next (even numbered) page. Note that `remarktext` cannot contain the character `>`. Otherwise use `<{remark}>`, also necessary if the first character of `remarktext` is `*`.

There exists also a starred version of the `\slide` command, which does not increase the page number, *cf.* `./examples/`.

`\titleslide` The `\titleslide` command is essentially an alias for `\slide{}`.

The following layout commands can be used throughout the text to change the slide layout and take effect from the following new slide onward. Figure 2 shows the global structure of a slide created with `pfuef`. Below are detailed descriptions of how to use the various layout possibilities shown in Figure 2

`\header` Defines the type of the header box. At present, there are three different predefined header boxes: with `\header0`, no header box is drawn; `\header1` and `\header2` place small images on the top of the slide underneath the title text, by default, these are dodger blue bars, available in the directory defined by the command `\figuredir`. See also Section 3 and especially Figure 4.

`\footer` Defines the type of footer box. Works similarly to the `\header` command (`\footer0`, `\footer1`, `\footer2`).

`\footertext` `\footertext{footertext}` puts the text `footertext` on top of the footer image.

Images used in the background, headers, or footers can be specified with the following commands, see Figure 3 for some examples.

`\figuredir` Sets the directory containing the layout figures. By default this is set to `\figuredir{./p5figures/}`.

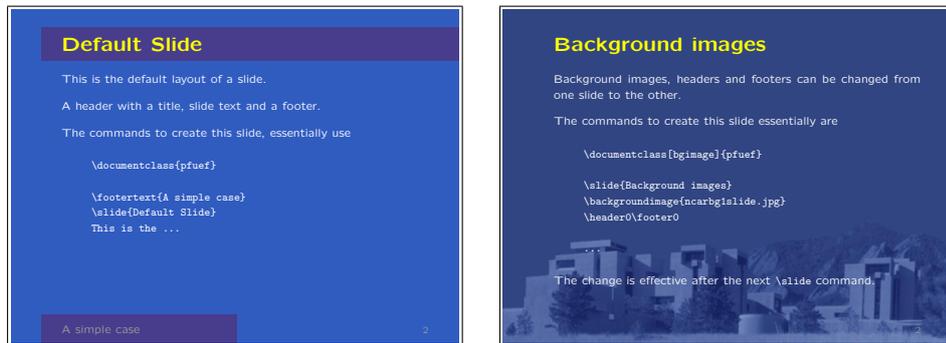


Figure 3: Illustration of default slide outline without and without background image. The corresponding L^AT_EX commands are given.

<code>\backgroundimage</code>	Defines the image used for the background. By default this is a light blue, such as defined in <code>p5figures/bgblue.png</code> . If you use other images, put them in the <code>\figuredir</code> directory. The command takes into effect with the next <code>\slide</code> command and can be used at any time.
<code>\titlebackgroundimage</code>	Defines the image used for the title page. By default this is a blue, such as defined in <code>p5figures/dodgerblue.png</code> .
<code>\titlefontsize</code> <code>\textfontsize</code>	Sets the default font size for the title page and ordinary slides. Default is <code>\titlefontsize{\LARGE}</code> and <code>\textfontsize{\normalsize}</code> . This command can be used to execute any macros or text at the beginning of each slide.
<code>\headerimage</code>	Defines the image used in some of the header layouts. By default it is plain dodger blue, <i>i.e.</i> <code>\headerimage{dodgerblue.png}</code> .
<code>\footerimage</code>	Defines the image used in some of the footer layouts. By default it is plain dodger blue, <i>i.e.</i> <code>\footerimage{dodgerblue.png}</code> .
	The use of personalized background, header and footer images may necessitate the change of page layout parameters such as left and right margins <i>etc.</i> , see Section 5 for more information on how to do that.
	There are several commands, which allow to change various colors. The arguments are defined colors for example from <code>rgb.tex</code> . Personalized colors can be specified by their rgb values with <code>\definecolor{myblue}{rgb}{0,0.53,0.83}</code> . In all following commands, the optional argument in brackets refers to the color used in the print modus.
<code>\slidebackgroundcolor</code>	Specifies the background color; default is <code>\pagecolor{bgblue}</code> (defined in <code>pfuef.cls</code> as rgb 0.188, 0.361, 0.753) in <code>bgcolor</code> or <code>bgimage</code> mode and <code>\pagecolor{white}</code> in print mode, <i>i.e.</i> <code>\slidebackgroundcolor[white]{bgblue}</code> .
<code>\headertextcolor</code>	Specifies the color of the header text; default is <code>\headertextcolor[gray40]{yellow}</code> .
<code>\footertextcolor</code>	Specifies the color of the footer text and the page number; default is <code>\footertextcolor[gray40]{gray60}</code> .
<code>\helpinggridin</code> <code>\helpinggridcm</code>	Places a grid spaced either in a distance of 1in or 1cm over the slide. This is

helpful when placing figures.

- `\arrow` Places an arrow at position #2 (horizontal) and #3 (vertical) of angle #4 and length #5. The arrow is specified by #1. There are many arrows in the figure directory (e.g., `arrow` (black, default), `whitearrow`, `yellowarrow`, `redarrow`, `shortarrow` (black), `shortwhitearrow`, `shortyellowarrow`, `shortredarrow`). The short arrows have a length of roughly 4cm, the long ones of roughly 10 cm. See also Figure 6 and Section 4.1.
- `\placeobject` Places object #3 at position #1 (horizontal) and #2 (vertical) by not changing the current typesetting position. See also Figure 6 and Section 4.1.
- `\placefig`
`\source` Places an image #3 at position #1 (horizontal) and #2 (vertical) by advancing only the horizontal typesetting position. If a fourth argument is given, a text typesetted by `\source` is printed below the figure. `\source` takes one argument and writes in a gray font of size `\tiny`. See also Figure 6 and Section 4.1.
- `\tc`
`\ec`
`\bc` The main text coloring is done with three commands: `\tc`, `\ec`, and `\bc`. The first is the color with which text is printed by default, the second is a color to emphasize parts of text. The third is a color close to or the same as the background color. I use `\bc` often in my pseudo overlay method to put emphasis on everything else; examples can be found on my [homepage](#). Note that `{\tc emphasized text}` is equivalent to `\ec emphasized text \tc`.
- `\setcolortc`
`\setcolorec`
`\setcolorbc` In order to change the colors `\tc`, `\ec`, and `\bc` use the command `\setcolortc`, `\setcolorec`, and `\setcolorbc` with default values `\setcolortc[blue]{white}`, `\setcolorec[bgblue]{bgblue}`, and `\setcolorbc[DarkBlue]{red}`.
- The package provides a few utility commands.
- `\pause` At this command, the PDF viewer waits for a “next page” key. It can be used for a sequential build of pages. `\pause` does not work with boxes like `\mbox` or `\fbox` or in some environments such as `\parbox` or `\minipage`. Due to the construction, a new paragraph is created after the `\pause` command. If the space between consecutively shown blocks is too large, you need to use a constructions like `\[-7mm]`.
- `\ppline` a simple line stretching across the entire slide, can be used in the `<>` construction for creating handouts in PowerPoint® style.
- `\pplines` `\pplines{n}` draws n lines as in `\ppline`.
- `\myitem` Gray bullet, similar to the dots of the default footer. It can, for example, be used with the environments `ul` or with `itemize` (e.g. `\item[\myitem]`).
- `\ul` The environment `\begin{ul} \end{ul}` constructs an unnumbered/undotted list. Instead of `\item`, use an empty line. The list for the `\slide` command is composed with this environment.

3 Examples

This section illustrates some of the functionalities of `pfuef`. More extensive examples are given in the directory `./example/`.

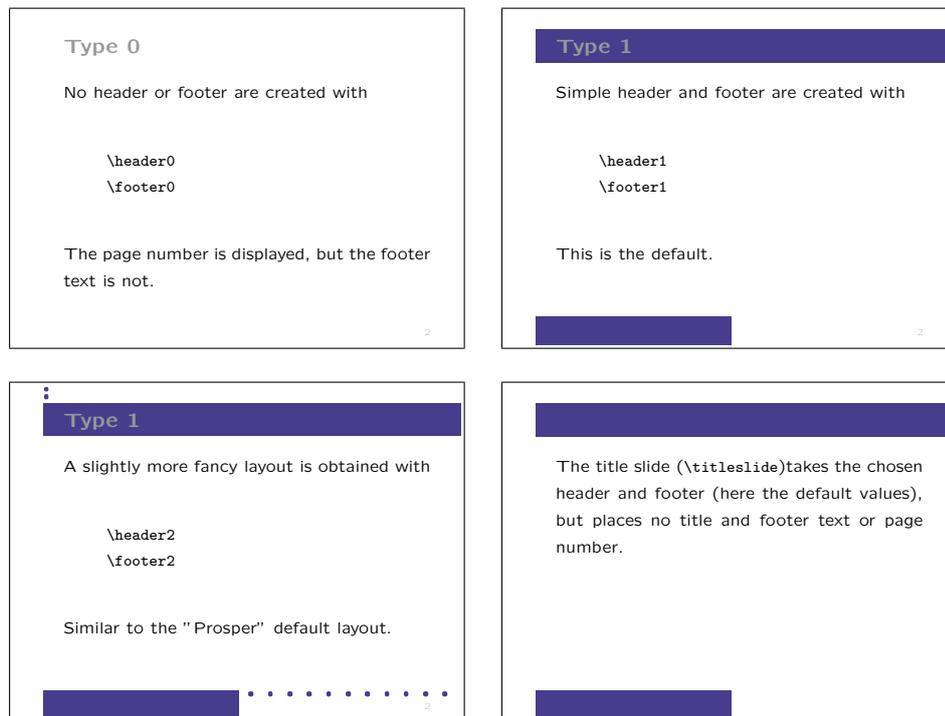


Figure 4: Different predefined types of headers and footers (displayed with print option of `pdfx`).

Figure 3 illustrates some of the basic features concerning background images and background colors.

Each slide consists of a header, slide contents and a footer. For the current version v2.3, three different types of headers and footers are predefined and can be selected with `\header` and `\footer`, as illustrated in Figure 4.

The option `ppremark` together with `print` reserves each even side for remarks on handouts. It is possible to provide additional remarks on the presented topic or life in general in the \LaTeX code within the `<>` delimiters. Figure 5 illustrates this powerful option.

4 Tips and Tricks for You and Me

4.1 Placing objects

Figure 6 some insights on how to place effectively figures. See the actual commands for noticing the differences in typesetting position advancements.

4.2 Transparent Images

To use images with transparent background, which overlay slide background images, we cannot use simple png files (ask News-Groups why). You have

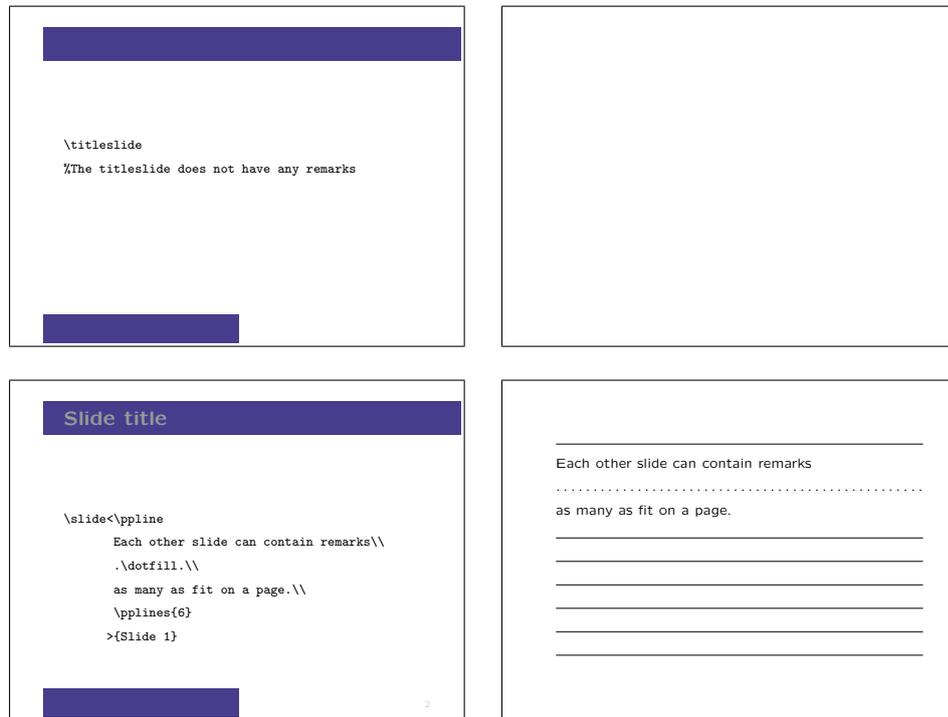


Figure 5: Example on how to use ppremarks.

to transform the png image to PDF format. A nice tool can be found at <http://sourceforge.net/projects/png2pdf>.

The open source program R (<http://www.r-project.org/>) creates PDF graphics with transparent backgrounds by default.

Sometimes it is necessary to have different figures for the presentation and printed version (if transparency is really not possible but colors in the image are adapted to the background of the slides. In that case, I recommend to generate the figures twice and put those for the final versions in a directory called **figures** and those intended for printing in a directory called **printfigures**. With the command

```
\ifbgimage\newcommand{\figdir}{./figures/}\else%
\newcommand{\figdir}{./printfigures/}\fi
```

placed before `\begin{document}` the corresponding directory is automatically chosen. Figures are included with the command

```
\includegraphics[height=12cm]{\figdir fig12.pdf}
```

(note the space!). Of course, a manual change in

```
\figuredir{}
```

would do the same job.

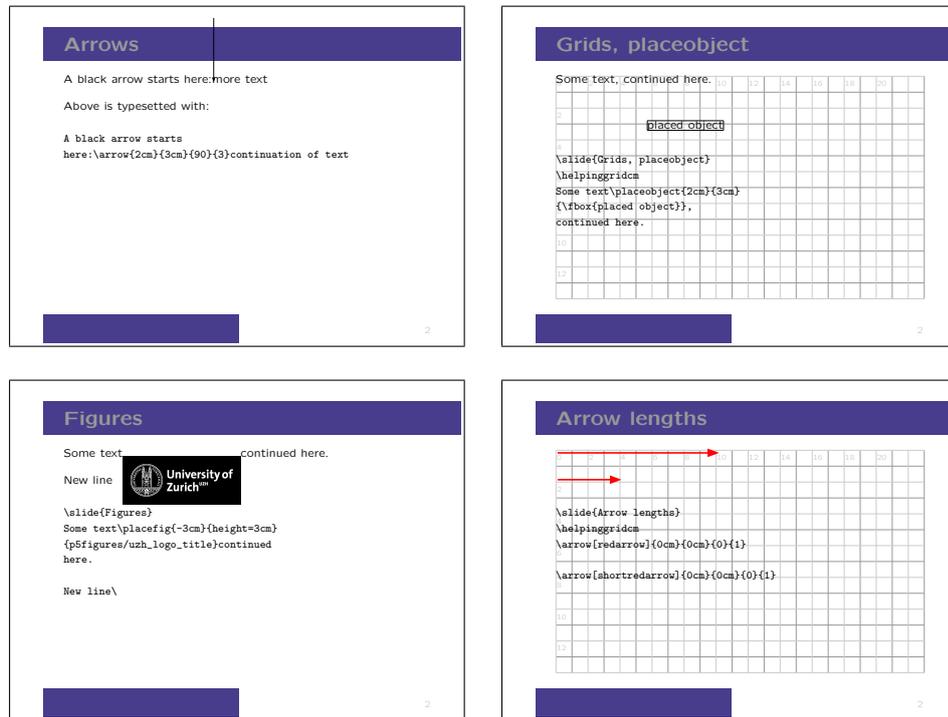


Figure 6: Example on how to use placing commands.

4.3 Print or Post Presentations

Often, several slides can be printed on the same page (also useful when the presentation is posted on the web). The following Unix/Linux commands allow to place six slides on one page.

```
pdfnup --nup 2x3 --orient portrait presentation.pdf
```

Often, the source code for slides or print material differs only in the options passed to `documentclass`. Instead of duplicating the file several time, it is possible to use the script `compile.p5`. It essentially takes the source file and adds an appropriate `\documentclass[...]{pfuef}` line. Therefore the source file cannot contain a `\documentclass` command! For further help, execute

```
compile.p5 -h
```

4.4 Animated Graphics

There are essentially two possibilities to animate graphics within a PDF presentation. The first uses the `hyperref` functionality, the second uses a supplementary package.

The first method is a simple way to call external programs from Acrobat[®] Reader[™]. To run the program `./animatelauncher`, use the \LaTeX command

```
 $\href{run:./animatelauncher}{\text{some text or graphic}}$ 
```

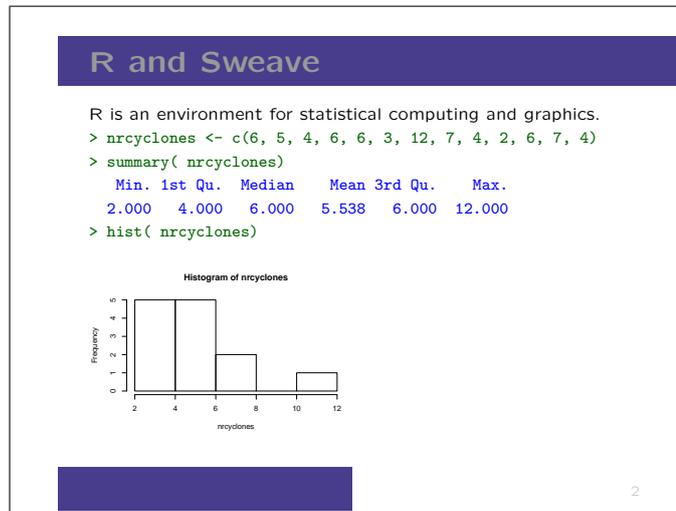


Figure 7: Slide created with R and Sweave.

The program `animatelauncher` could be a shell script such as

```
#!/bin/sh
animate -geometry 432x273+102+180 -delay 40 figures/frame?.png
```

The Acrobat[®] Reader[™] demands whether any program should be executed or not, which can be disturbing during a presentation. Therefore, I execute a dummy script, *i.e.* an empty shell script before the beginning and answer by clicking on All.

The quite new package `pdfanim` written by Jochen Skupin is an efficient and neat way to implement animated PDFs. The version v0.50 is compatible with `pdfuef`, but some programs such as `xpdf` and `pdftops` can not resolve the animations correctly.

The web offers many more possibilities, though most of them should be placed in the next section (*e.g.* [Animations in pdfTeX-generated PDF](#) by Jan Holevcek and Petr Sojka).

4.5 R and Sweave

R is a language and environment for statistical computing and graphics. It produces well-designed publication-quality plots with a careful choice of default values. As R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form, it is *the* choice of software environment for statisticians and many scientists.

R with the Sweave environment allows to create documents containing source code and output (results, figures, ...) from the source code. Figure 7 gives a basic example.

The source code is

```
% begin file sweave.Rnw
\documentclass[print]{pdfuef}
\usepackage[nogin]{p5Sweave}
```

```

\begin{document}
\slide{R and Sweave}
R is an environment for statistical computing and graphics.
\setkeys{Gin}{height=.5\textheight}
<<fig=T,height=4,width=6>>=
nrcyclones <- c(6, 5, 4, 6, 6, 3, 12, 7, 4, 2, 6, 7, 4)
summary( nrcyclones)
hist( nrcyclones)
@
\end{document}
% eof sweave.Rnw

```

And the file is processed with

```

R CMD Sweave sweave.Rnw
pdflatex sweave.tex

```

For more details see <http://www.statistik.lmu.de/~leisch/Sweave/>.

5 Tips and Tricks for T_EXnicians

pfuef requires the following packages:

```

\RequirePackage[usenames,dvipsnames]{color}
\RequirePackage[pdftex]{graphicx}
\RequirePackage{calc}

```

Although not strictly necessary, it also loads

```

\usepackage{amssymb,amsmath}
\usepackage[colorlinks=true,%
plainpages=false,pageanchor=false,pdfpagelabels=false,%
pdfcreator={pdflatex with pfuef}]{hyperref}

```

If you need to pass different options to the hyperref package, you may use a `hyperref.cfg` file or edit `pfuef.cls` manually.

By default, pfuef sets the following graphics extensions:

```

\DeclareGraphicsExtensions{.pdf,.png,.jpg}

```

Figure 8 gives the slide layout parameters. The default values are

<code>\paperheight</code>	210mm	<code>\paperwidth</code>	280mm
<code>\hoffset</code>	0.25in	<code>\voffset</code>	0.50in
<code>\srighmargin</code>	0.75in	<code>\stopmargin</code>	0.60in
<code>\hleftmargin</code>	0.50in	<code>\hfrightmargin</code>	0.00in
<code>\footerimageheight</code>	0.80in	<code>\footerimagewidth</code>	4.80in
<code>\headerimageheight</code>	0.70in		

As we still use the standard page layout, a few parameter have been defined as follows

<code>\oddsidemargin</code>	0in	<code>\evensidemargin</code>	0in
<code>\headheight</code>	0in	<code>\topmargin</code>	0in
<code>\headsep</code>	0in	<code>\parindent</code>	0in

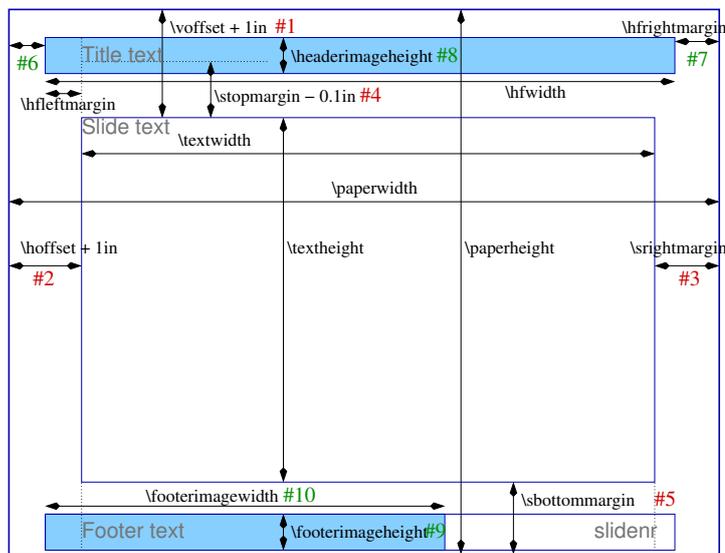


Figure 8: Slide layout parameters.

The default values for `\parskip` are 14pt (tiny), 15pt (small), 20pt (normalsize and large), 24pt (Large, huge) respectively.

With respect to lists, the following values are used

```

\topsep 10pt
\parsep 5pt
\itemsep 10pt

```

`\settextmargins` The command `settextmargins` allows to set five values concerning slide text margins, the parameters are indicated with #1 to #5 in Figure 8. By default `\settextmargins{1.5in}{1.25in}{.75in}{.5in}{.1in}`.

`\sethfmargins` The command `sethfmargins` allows to set five values concerning the header and footer placement, the parameters are indicated with #6 to #10 in Figure 8. By default `\sethfmargins{.75in}{0in}{.8in}{.7in}{4.8in}`.

With the previous two commands in addition to the other features of `pfuef`, it is possible to “design” a personalized slide layout. Figure 9, for example, shows the capacity of `pfuef`, to create a wide variety of slide layouts with a few simple commands.

`\slide` The `\slide` command executes the following steps:

- if `ppremarks` but not the first page, process the remark page
- if `bgimage`, place background image
- place header, title text, footer, footer text and slide number

Then the consecutive text is processed on the active page. Since the `ppremarks` option is quite different to an optional argument, the `<>` construction was chosen. This also leaves future extensions with optional arguments (`[]`) open (and it was fun implementing).

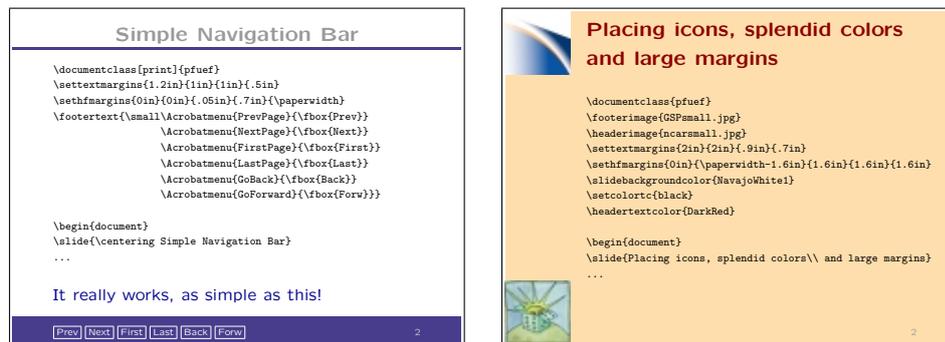


Figure 9: Illustration of other slide layouts. The corresponding L^AT_EX commands are given.

`\header` The commands `\header` and `\footer` take one token and define accordingly
`\footer` `\@header` and `\@footer`. To create personalized headers and footers, it is possible
`\@header` to redefine just the latter commands. Of course, one needs a few trial and errors
`\@footer` in order to obtain the correct placements and dimensions.

`\pause` The `\pause` command essentially saves the composed output in a `vbox`, creates
a new page and restores the page with the previously saved contents. The method
is equivalent to Klaus Guntermann's version 1.1 (25 May 99) of `pause.sty`. This
is also the reason why it only works in LR mode.

To simplify things, I reset the font size to `\normalsize` at the beginning of
each slide. That means, I do not have to mess with variable baseline skips.

`itemize` For lists, the following left margins are defined: `\leftmargini{38pt}`,
`\leftmarginii{30pt}`, `\leftmarginiii{20pt}`, etc. The other list parameters
are set to `\leftmargin\leftmargini`, `\parsep .5\parskip`, `\topsep \parsep`,
`\itemsep\parskip`, `\partopsep\z@`, where `\parsep{5pt}`. For second lists the
values are the same up to `\labelwidth`, which is adjusted according to the left
margin.

6 Frequently asked questions

1. *Do I really need the directory `p5figures`?*

No, the package works without, provided no headers, footers etc are used.
Hence, calls `\footer0` and `\header0` are necessary.

2. *The spacings in the lists are too large. Can I change that?*

The spacings are set to paragraph spacings. Changing (globally) the value
of `\parsep` will do the trick.

3. *How can I set `parskip` globally to `0pt`?*

Each time a font is changed, the `\parskip` (and other parameters) are reset
to the default values. Instead of redefining `\parskip` manually, it is possible
to use

```
\textfontsize{\Large\parskip 30pt}
```

for example.

4. *Can I add a second footer figure instead of a footer text?*

Yes, it is possible to define the footer text as an image. To avoid unnecessary spacings, the image can be placed within a `\raisebox` command. For example:

```
\footertext{\hspace*{40mm}\raisebox{-3.6mm}[0pt][0pt]{%  
  \includegraphics[height=30mm,width=208mm]{banner}}}
```

would work (provide the image exists and is in the figure path).

5. *Can specific packages be loaded by default?*

Yes, put the `\usepackage{...}` commands into the global style file and use the option `cfg`.

7 Disclaimer

The author does not guarantee the correctness of any function or program in this package.

8 Release and Revision Details

The file `history.txt` contains further informations about implementation history.

- 2012-06-07: Releases of version 2.3 and 2.4-uzh:
(releases 2.1, 2.2 were rather development versions)
Many new features like:
- new package options (`gridin`, `gridcm`, `uzh`, `cfg`)
 - new macros (`arrow`, `placefig`, `placeobject`, ...)
 - possibility of setting paragraph and list spacing
 - new examples
 - updated manual
 - more control to personalize the layout
 - default setting is independent of figures provided in `./p5figures`
 - distribution of `p5Sweave.sty`
- Slightly different default spacing/setting may cause a minor loss in backwards compatibility..
- 2009-08-07: Release of version 2.0: the new version has a slightly different default spacing and space arrangement. Hence, we have lost a bit of backwards compatibility. Many new features are available:
- commands to change textsize and titlesize, default is:

```

\titlefontsize{\LARGE}
\textfontsize{\normalsize}

```
 - the figure for the title slide can be specified now with

```

\titlebackgroundimage{ }

```
 - new option `fields`, used to create the slides for the ‘fields tour’.
- The script file `compile.p5` for efficient compilation is included.
- 2004-09-25: Release of version 1.0:
quite a few new commands concerning color and page layout handling;
backward compatibility with lower versions could not be maintained completely.
- 2004-08-25: Release of version 0.3:
this is essentially a beta version for 1.0
- 2004-05-05: Release of version 0.2:
a distributed and stable version
- 2004-02-12: Release of version 0.1

June 14, 2012
Reinhard Furrer
Institute of Mathematics
University of Zurich