Creating PDFLaTeX Presentations[*]

Reinhard Furrer
National Center for Atmospheric Research
Furrer@ucar.edu

2004/09/23

Abstract

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

Contents

1 Introduction 2

2 The pfuef package 2

2.1 Package Installation and Contents 2

2.2 Package Options 2

2.3 Package Commands 3

3 Examples 6

4 Tips and Tricks for You and Me 6

4.1 Transparent Images 6

4.2 Print or Post Presentations 7

4.3 Animated Graphics 8

5 Tips and Tricks for TeXnichans 8

6 Disclaimer 10

7 Release and Revision Details 10

[*] This file (`pfuefmanual.pdf`) is for version number v1.0, last revised 2004/09/23.

1 Introduction

The `pfuef` package is a simple, rather customized set of macros for PDFLaTeX 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 PDFLaTeX. Prosper was rather close to a satisfactory layout, however it was not compatible with PDFLaTeX. 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**DFLaTeX 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. 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

2.1 Package Installation and Contents

The most recent version of `pfuef` and supporting material can be downloaded from http://www.cgd.ucar.edu/~furrer/research/programs.html. Copy the file `pfuef.v1.0.tar` into a convenient place and untar it with the command `tar xvf pfuef.v1.0.tar`.

`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/`.

`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}`.

`./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 4.3).

2.2 Package Options

There are five 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 210mm × 280mm, *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.

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 bgimage, *i.e.* do not consider this page in print option. This option is useful in pseudo overlay mode.

\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 1 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 1

\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 3.

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

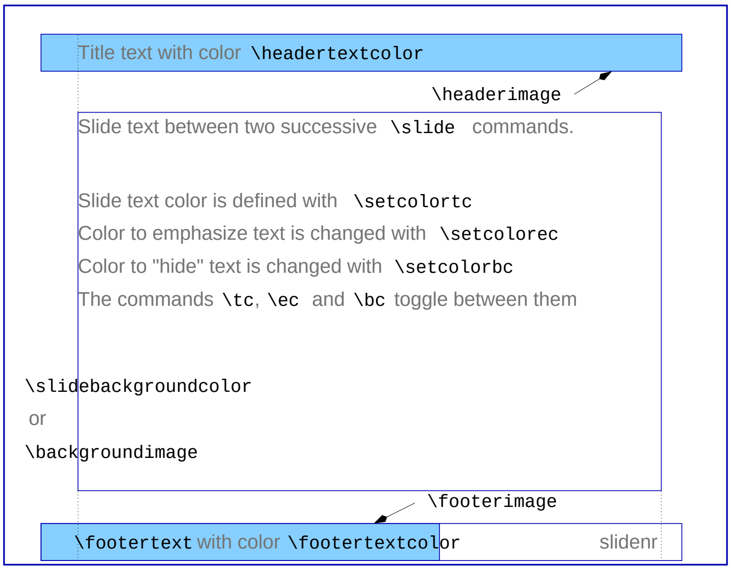

Title text with color \headertextcolor
\headerimage
Slide text between two successive \slide commands.
Slide text color is defined with \setcolortc
Color to emphasize text is changed with \setcolorec
Color to "hide" text is changed with \setcolorbc
The commands \tc, \ec and \bc toggle between them
\slidebackgroundcolor
or
\backgroundimage
\footerimage
\footertext with color \footertextcolor
slidenr

Figure 1: Slide layout, color and images.

`\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 2 for some examples.

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

`\backgroundimage` Defines the image used for the background. By default this is a light blue, such as defined in p5figures/bgblue.png. If you use other images, put them in the \figuredir directory.

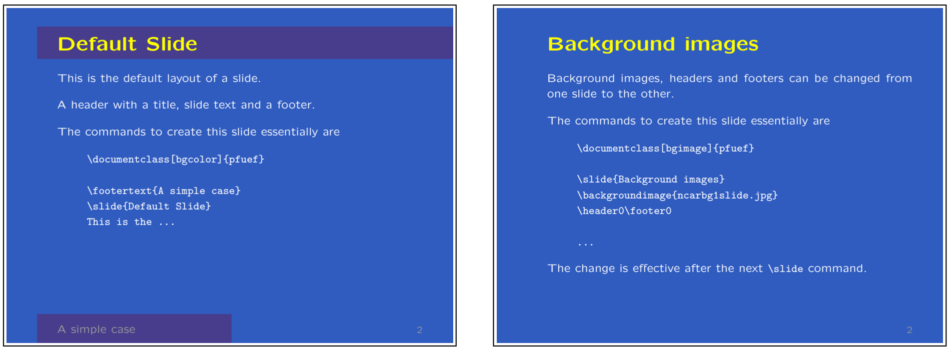
`\headerimage` Defines the image used in some of the header layouts. By default it is plain dodger blue, *i.e.* \headerimage{dodgerblue.png}.

`\footerimage` Defines the image used in some of the footer layouts. By default it is plain dodger blue, *i.e.* \footerimage{dodgerblue.png}.

The use of personalized background, header and footer images will necessitate. the handling of page layout parameters such as left and right margins *etc.*, 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 rgb.tex. Personalized colors can be specified by their rgb values with \definecolor{myblue}{rgb}{0,0.53,0.83}. In all following commands, the optional argument in brackets refers to the color used in the print modus.

`\slidebackgroundcolor` Specifies the background color; default is \pagecolor{bgblue} (defined in pfuef.cls as rgb 0.188, 0.361, 0.753) in bgcolor or bgimage mode and \pagecolor{white} in print mode, *i.e.* \slidebackgroundcolor[white]{bgblue}.



Default Slide

This is the default layout of a slide.

A header with a title, slide text and a footer.

The commands to create this slide essentially are

```
\documentclass[bgcolor]{pfuef}

\footertext{A simple case}
\slide{Default Slide}
This is the ...
```

A simple case

2

Background images

Background images, headers and footers can be changed from one slide to the other.

The commands to create this slide essentially are

```
\documentclass[bgimage]{pfuef}

\slide{Background images}
\backgroundimage{ncarbg1slide.jpg}
\header0\footer0

...
```

The change is effective after the next `\slide` command.

2

Figure 2: Illustration of default slide outline without and without background image. The corresponding LaTeX commands are given.

`\headertextcolor` Specifies the color of the header text; default is `\headertextcolor[gray40]{%` `yellow}`

`\footertextcolor` Specifies the color of the footer text and the page number; default is `\footertextcolor[gray40]{gray60}`.

`\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`, `\setcolorbc`, and `\setcolorbc` with default values `\setcolortc[blue]{white}`, `\setcolorbc[bgblue]{bgblue}`, and `\setcolorec[DarkBlue]{red}`.

The package provides a few utility commands.

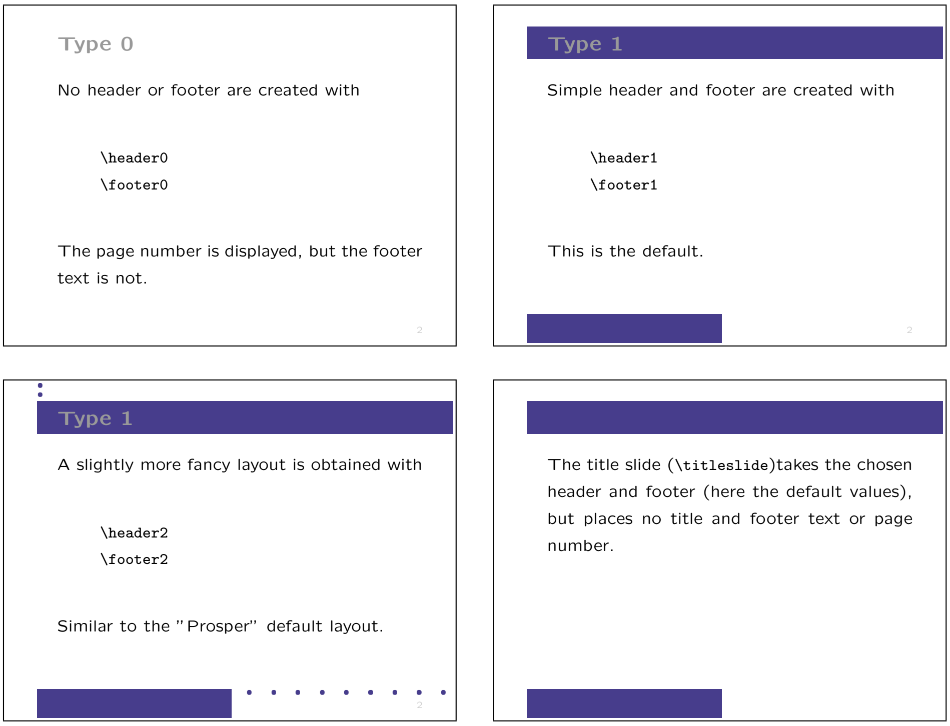
`\pause` At this command, the PDF viewer waits for a “next page” key. It can be used for a sequential build of pages. `\pause` does not work with boxes like `\mbox` or `\fbox` or in some environments such as `\parbox` or `\minipage`. Due to the construction, a new paragraph is created after the `\pause` 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.



Type 0

No header or footer are created with

```
\header0
\footer0
```

The page number is displayed, but the footer text is not.

2

Type 1

Simple header and footer are created with

```
\header1
\footer1
```

This is the default.

2

Type 1

A slightly more fancy layout is obtained with

```
\header2
\footer2
```

Similar to the "Prosper" default layout.

2

The title slide (`\titleslide`)takes the chosen header and footer (here the default values), but places no title and footer text or page number.

Figure 3: Different predefined types of headers and footers (displayed with print option of pfuef).

3 Examples

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

Figure 2 illustrates some of the basic features concerning background images and background colors.

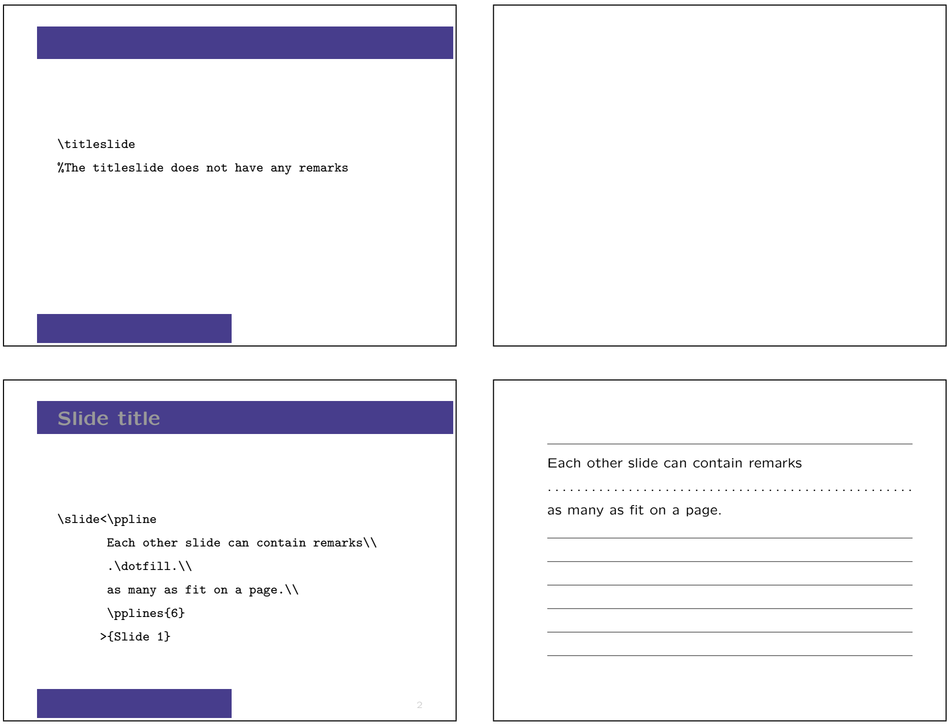
Each slide consists of a header, slide contents and a footer. For the current version v1.0, three different types of headers and footers are predefined and can be selected with `\header` and `\footer`, as illustrated in Figure 3.

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 4 illustrates this powerful option.

4 Tips and Tricks for You and Me

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



```
\titleslide
%The titleslide does not have any remarks
```

Slide title

```
\slide<\ppline
       Each other slide can contain remarks\\
       .\dotfill.\\
       as many as fit on a page.\\
       \pplines{6}
      >{Slide 1}
```

2

Each other slide can contain remarks

. . . .

as many as fit on a page.

Figure 4: Example on how to use ppremarks.

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!).

4.2 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.

```
   pdftops presentation.pdf
```

```
psnup -l -nup 6 presentation.ps 6pages.ps
ps2pdf 6pages.ps 6pages.pdf
rm presentation.ps
```

4.3 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}{some text or graphic}
```

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 pfuef, 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 Holeček and Petr Sojka).

5 Tips and Tricks for TeXnichans

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,%
            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.

Figure 5 gives the slide layout parameters. The default values are

`\paperheight`	`210mm`	`\paperwidth`	`280mm`
`\hoffset`	`0.25in`	`\voffset`	`0.50in`
`\srightmargin`	`0.75in`	`\stopmargin`	`0.60in`
`\hfleftmargin`	`0.50in`	`\hfrightmargin`	`0.00in`
`\footerimageheight`	`0.80in`	`\footerimagewidth`	`4.80in`
`\headerimageheight`	`0.70in`		

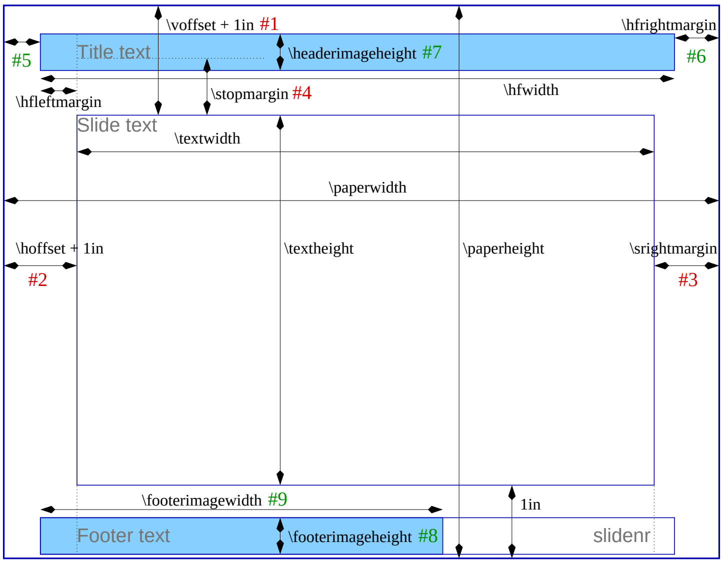


Figure 5: Slide layout parameters.

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

`\oddsidemargin`	`0in`	`\evensidemargin`	`0in`
`\headheight`	`0in`	`\topmargin`	`0in`
`\headsep`	`0in`	`\parindent`	`0in`

`\settextmargins` The command `settextmargins` allows to set four values concerning slide text margins, the parameters are indicated with `#1` to `#4` in Figure 5. By default `\settextmargins{1.5in}{1.25in}{.75in}{.5in}`.

`\sethfmargins` The command `\sethfmargins` allows to set five values concerning the header and footer placement, the parameters are indicated with `#5` to `#9` in Figure 5. By default `\sethfmargins{.75in}{0in}{.8in}{.7in}{4.8in}`.

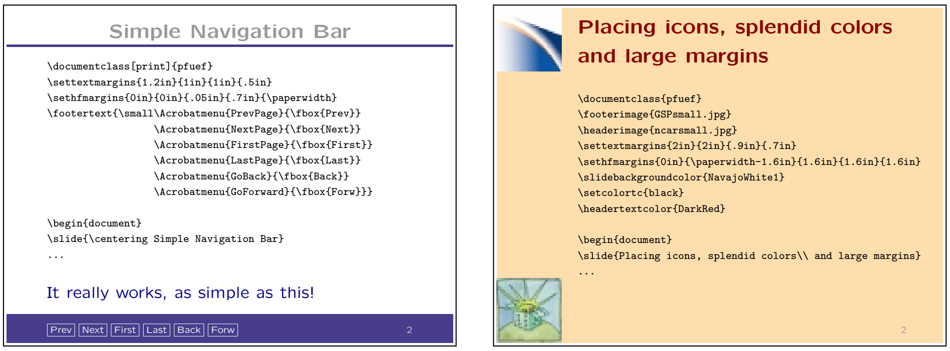
With the previous two commands in addition to the other features of pfuef, it is possible to “design” a personalized slide layout. Figure 6, for example, shows the capacity of pfuef, to create a wide variety of slide layouts with a few simple commands.

`\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).

`\header` The commands `\header` and `\footer` take one token and define accordingly



Simple Navigation Bar

```
\documentclass[print]{pfuef}
\settextmargins{1.2in}{1in}{1in}{.5in}
\sethfmargins{0in}{0in}{.05in}{.7in}{\paperwidth}
\footertext{\small\Acrobatmenu{PrevPage}{\fbox{Prev}}
                   \Acrobatmenu{NextPage}{\fbox{Next}}
                   \Acrobatmenu{FirstPage}{\fbox{First}}
                   \Acrobatmenu{LastPage}{\fbox{Last}}
                   \Acrobatmenu{GoBack}{\fbox{Back}}
                   \Acrobatmenu{GoForward}{\fbox{Forw}}}

\begin{document}
\slide{\centering Simple Navigation Bar}
...
```

It really works, as simple as this!

Placing icons, splendid colors and large margins

```
\documentclass{pfuef}
\footerimage{GSPsmall.jpg}
\headerimage{ncarsmall.jpg}
\settextmargins{2in}{2in}{.9in}{.7in}
\sethfmargins{0in}{\paperwidth-1.6in}{1.6in}{1.6in}{1.6in}
\slidebackgroundcolor{NavajoWhite1}
\setcolortc{black}
\headertextcolor{DarkRed}

\begin{document}
\slide{Placing icons, splendid colors\\ and large margins}
...
```

Figure 6: Illustration of other slide layouts. The corresponding LaTeX commands are given.

\footer
\@header
\@footer

\@header and \@footer. To create personalized headers and footers, it is possible to redefine just the latter commands. Of course, one needs a few trial and errors 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.

6 Disclaimer

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

7 Release and Revision Details

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

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

September 26, 2004
Reinhard Furrer
GSP/NCAR