

# Presentations using LaTeX

## An Introduction to the Beamer Class

Marek Seliger

April 18, 2007

- 1 Introduction
  - Aspects of presentations using a computer
  - Why using LaTeX to produce a PDF Presentation
  - First Steps
- 2 Creation of a Presentation
  - Basic steps
  - Structuring slides
- 3 Advanced
  - Timings
  - Alternative Timing
  - Colors
  - Overprint
- 4 Bonus
  - Grafics
  - Modi and Themes

# Why using a computer?

- It's up to date
- Get color, incremental display for free
- Lecture halls have usually a better beamer than projector
- No need to lug slides to conference
- Talk can be easily posted and sent to colleagues
- Copier slides cost 0.10 EUR each, printer slides 1 EUR each
- No drag-and-drop, i.e. everything is reproducible

# Slides look different through a projector than on a screen

As you have limited space go easy on decorations and navigation.

As the resolution is low, smoothed characters look differently.

In particular be careful with thin lines: A

Colors look different than on a Monitor, depending on the projector:

- LCD Projector (Older Technology, large projectors): Light colors will not be visible
- DLP Projector (Newer, usually very small): Tiny mirrors produce very bright picture but colors come out darker

# Projector Trouble

You cannot assume that your laptop will work by default on projector X.

In particular older projectors may not be able to synchronize to new laptops.

If possible at all, try out your laptop in advance.

Also know how to change your display settings.

If all fails, have a Plan B:

- Have your slides on a memory stick (+USB cable).
- PDF is probably the only format any laptop can display
- Think about carrying printed slides

# Why using LaTeX to produce a PDF Presentation

## Why using LaTeX

- Standard mathematical typesetting
- Reuse already written up material

## Why producing a PDF

- PDF is highly portable (Portable Document Format)
- PDF works on virtually any platform
- Full screen display (CTRL-L) with every resolution

# What is LaTeX Beamer?

The beamer class is a LaTeX class that allows you to create a beamer presentation. It can also be used to create slides. It behaves similarly to other packages like Prosper, but has the advantage that it works together directly with pdflatex, but also with dvips. Text describing the class in this presentation is taken from their website: <http://latex-beamer.sourceforge.net/>.

# Useful features

The beamer class has several useful features:

- You don't need any external programs to use it other than `pdflatex`, but it works also with `dvips`.
- You can easily and intuitively create sophisticated overlays.
- Finally, you can easily change the whole slide theme or only parts of it.



# Advantages

The beamer class has several useful advantages:

- You only need a text editor and LaTeX to create a slideshow
- You only need a PDF reader to show it
- Your presentation is in a light text file (+ media if present)
- There are plenty of themes readily available
- Big letters = less text = more easy to read / understand

# Installation

To use beamer, you need to have three packages installed in your TEX path:

*beamer*, *xcolor* and *pgf*.

You can find that all at <http://latex-beamer.sourceforge.net/>

You install the packages by unpacking them either in the global or your private TEX tree.

To update a lookup table you need to call texhash after unpacking. In MiKTEX use the update wizard to install the packages called *latex-beamer*, *xcolor*, and *pgf*.

# Basic steps to use it

Once you have installed the beamer class, the basic steps to create a beamer presentation are the following:

- 1 Specify beamer as document class instead of article.
- 2 Structure your LaTeX text using section and subsection commands.
- 3 Place the text of the individual slides inside frame commands.
- 4 Run `pdflatex` on the text (or `latex` and `dvips`).

# References

Till Tantau  
Users Guide to the Beamer Class  
Version 3.01

<http://latex-beamer.sourceforge.net>

# Creation of a Presentation

## Compile with

- 1 `pdflatex file.tex`

The PDF-File can be viewed with a viewer of your choice.  
Adobe Reader supports transition effects and Full-Screen-modus.

## alternative:

- 1 `latex file.tex`
- 2 `dvips -o file.ps file.dvi`
- 3 `ps2pdf file.ps`

# Minimal Template

```
\documentclass{beamer}
\usetheme{Berkeley}

\title{My Presentation}
\author{Myself}
\institute{Here}
\date{January 2006}
\begin{document}
\frame{ \titlepage}

\frame{ BlaBla }

\end{document}
```

What if you need more room on a slide? You can also do a plain slide (like this one, without header and footer graphics) by using

```
\begin{frame}[plain]
```

# Block and Example

## Fancy Block Title

You can place text (or math, or whatever) in a snazzy colored block like this just by surrounding with

```
\begin{block}{Snazzy Block Title}
```

You can ...

```
\end{block}
```

## Example

Beamer also offers the example environment

```
\begin{example}
```

```
\end{example}
```



# Block and Example

## Theorem

$$2=1$$

## Proof.

$$-2 = -2$$

$$4 - 6 = 1 - 3$$

$$4 - 6 + 9/4 = 1 - 3 + 9/4$$

$$(2 - 3/2)^2 = (1 - 3/2)^2$$

$$2 - 3/2 = 1 - 3/2$$

$$2 = 1$$



# Use of Alert In Beamer

You can emphasize text, like this, using **this** which is colored red by default.

**Heres An alertblock**

Get an emphasized block using

```
\begin{alertblock}{Alert Block Title}
\end{alertblock}
```

# Columns

Smaller Column 1

...

Bigger Column 2

...

# Columns

Smaller Column 1

...

Bigger Column 2

...

```

\begin{columns}
\begin{column}{3cm}
\begin{block}{Smaller Column 1} ... \end{block}
\end{column}
\begin{column}{5cm}
\begin{block}{Bigger Column 2} ... \end{block}
\end{column}
\end{columns}

```

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay
- Item Bee

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay
- Item Bee
- Item See



# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay
- Item Bee
- Item See
- ① Item Won

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay
- Item Bee
- Item See
- ① Item Won
- ② Item Too

# Pause, Timings, Etc.

Within practically any LATEX environment, use

`\pause`

before any command to delay its appearance:

- Item Aaay
- Item Bee
- Item See
- ① Item Won
- ② Item Too
- ③ Item Tree

# Pause Source Code

```
\begin{itemize}
\pause \item Item Aaay
\pause \item Item Bee
\pause \item Item See
\end{itemize}

\begin{enumerate}
\pause \item Item Won
\pause \item Item Too
\pause \item Item Tree
\end{enumerate}
```

# Alternative Timing Specifications With Alert

To highlight the current line

## Alternative

- Item Aaay

```
\begin{itemize}[<+--| alert@+>]
\item Item Aaay
\item Item Bee
\item Item See
\item Item Dee
\end{itemize}
```

# Alternative Timing Specifications With Alert

To highlight the current line

## Alternative

- Item Aaay
- **Item Bee**

```
\begin{itemize}[<+--| alert@+>]
\item Item Aaay
\item Item Bee
\item Item See
\item Item Dee
\end{itemize}
```

# Alternative Timing Specifications With Alert

To highlight the current line

## Alternative

- Item Aaay
- Item Bee
- **Item See**

```
\begin{itemize}[<+--| alert@+>]
\item Item Aaay
\item Item Bee
\item Item See
\item Item Dee
\end{itemize}
```

# Alternative Timing Specifications With Alert

To highlight the current line

## Alternative

- Item Aaay
- Item Bee
- Item See
- **Item Dee**

```
\begin{itemize}[<+--| alert@+>]
\item Item Aaay
\item Item Bee
\item Item See
\item Item Dee
\end{itemize}
```



# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

### First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\  
\alert<2,5,8>{Loop instruction $a$} \\  
\alert<3,6,9>{Loop instruction $b$} \\  
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\
\alert<2,5,8>{Loop instruction $a$} \\
\alert<3,6,9>{Loop instruction $b$} \\
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

**Loop instruction *b***

Loop instruction *c*

```
\alert<1>{First instruction} \\
\alert<2,5,8>{Loop instruction $a$} \\
\alert<3,6,9>{Loop instruction $b$} \\
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\
\alert<2,5,8>{Loop instruction $a$} \\
\alert<3,6,9>{Loop instruction $b$} \\
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\  
\alert<2,5,8>{Loop instruction $a$} \\  
\alert<3,6,9>{Loop instruction $b$} \\  
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\  
\alert<2,5,8>{Loop instruction $a$} \\  
\alert<3,6,9>{Loop instruction $b$} \\  
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\
\alert<2,5,8>{Loop instruction $a$} \\
\alert<3,6,9>{Loop instruction $b$} \\
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\  
\alert<2,5,8>{Loop instruction $a$} \\  
\alert<3,6,9>{Loop instruction $b$} \\  
\alert<4,7,10>{Loop instruction $c$} \\
```



# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

**Loop instruction *b***

Loop instruction *c*

```
\alert<1>{First instruction} \\
\alert<2,5,8>{Loop instruction $a$} \\
\alert<3,6,9>{Loop instruction $b$} \\
\alert<4,7,10>{Loop instruction $c$} \\
```

# Loops With Timings

Notice how we can cycle through items.

## Some Algorithm

First instruction

Loop instruction *a*

Loop instruction *b*

Loop instruction *c*

```
\alert<1>{First instruction} \\  
\alert<2,5,8>{Loop instruction $a$} \\  
\alert<3,6,9>{Loop instruction $b$} \\  
\alert<4,7,10>{Loop instruction $c$} \\
```

# Timings

## A Block With Timings

- Item Aaay
- Item Eee

```
\item<1-3> Item Aaay
```

```
\item<2-4> Item Bee
```

```
\item<3-5> Item See
```

```
\item<4-> Item Dee
```

```
\item<-4> Item Eee
```

# Timings

## A Block With Timings

- Item Aaay
- Item Bee
  
- Item Eee

```
\item<1-3> Item Aaay
```

```
\item<2-4> Item Bee
```

```
\item<3-5> Item See
```

```
\item<4-> Item Dee
```

```
\item<-4> Item Eee
```

# Timings

## A Block With Timings

- Item Aaay
- Item Bee
- Item See
  
- Item Eee

```
\item<1-3> Item Aaay
```

```
\item<2-4> Item Bee
```

```
\item<3-5> Item See
```

```
\item<4-> Item Dee
```

```
\item<-4> Item Eee
```

# Timings

## A Block With Timings

- Item Bee
- Item See
- Item Dee
- Item Eee

```
\item<1-3> Item Aaay
```

```
\item<2-4> Item Bee
```

```
\item<3-5> Item See
```

```
\item<4-> Item Dee
```

```
\item<-4> Item Eee
```

# Timings

## A Block With Timings

- Item See
- Item Dee

```
\item<1-3> Item Aaay
```

```
\item<2-4> Item Bee
```

```
\item<3-5> Item See
```

```
\item<4-> Item Dee
```

```
\item<-4> Item Eee
```

# Colors, the Basics

```
\setbeamercolor{mycolor}{fg=red,bg=blue}  
\begin{beamercolorbox}{mycolor}  
some text in red on blue background  
\end{beamercolorbox}
```

```
\setbeamercolor{mycolor2}{fg=blue,bg=green}  
\begin{beamercolorbox}{mycolor2}  
some text in blue on grey background  
\end{beamercolorbox}
```



# Colors, the Basics 2

- `fg=red,bg=white`
- `fg=black,bg=red`
- `fg=yellow,bg=black`
- `fg=red,bg=green`
  - are there even more colors?

# Colors Advanced

You can create your own colors

- `fg=red!90!blue,bg=white`
- `fg=red!60!blue,bg=white`
- `fg=red!40!blue,bg=white`
- `fg=red!10!blue,bg=white`
  - the number in between determines the mixture

# Colors

## Political Nonsense

When in the course of human events ...

# Colors

## Political Nonsense

When in the course of human events ...

# Colors

## Political Nonsense

When in the **course** of **human events** ...

# Overprint

## Theorem

*Here is a theorem*

## Proof.

Here is the first (bad) proof, in red.

# Overprint

## Theorem

*Here is a theorem*

## Proof.

Here is the second (presumably correct) ...



# How to do an Overprint

Use something like the following. Notice the *onslide* timings.

```
\begin{overprint}
  \onslide<1>
  \begin{alertblock}{Proof.}
    Here is the first (bad) proof, in red.
  \end{alertblock}

  \onslide<2>
  \begin{proof}
    Here is the second (presumably correct) ...
  \end{proof}
\end{overprint}
```



# Include Graphics (JPG)

Include a jpg-picture



# Include Graphics (JPG)

Include a jpg-picture



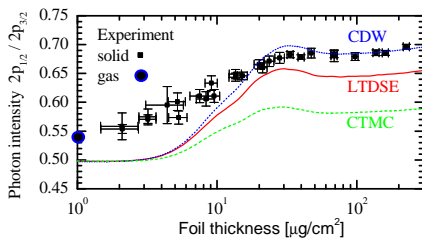
To include an eps-figure, first convert it to pdf using

- Linux: `ps2pdf figure.eps figure.pdf`
- Windows: use AdobeDistiller or print to a pdf-printer

# Include Graphics (PDF)

Include a pdf-picture

```
\includegraphics [width=8cm] {figure.pdf}
```



# Modi for different Media

## Presentation

```
\mode<presentation>  
\mode<presentation>{\usetheme{Berlin}}
```

## Handouts

```
\mode<handout>  
{\beamertemplatesolidbackgroundcolor{black!5}}
```

## Text/Article

```
\mode<article>{\usepackage{fullpage}}\
```

# Different themes

There are different styles for the beamer class:

```
\usetheme[compress]{Ilmenau}
```

You can use *a lot* of themes already included in Beamer (see in `/usr/share/texmf/tex/latex/beamer/beamertheme*.sty`).

Try also: Bergen, Antibes, bars, Berlin, JuanLesPins, Madrid, PaloAlto, Singapore, beamerthemesplit, and many more.

# FIN

Thank you for your attention!