Presentations in LaTeX

Introduction to the beamer class

Gonzalo Rivero

New York University

January 23, 2012

Introduction

Not the only class for presentations but the current standard in LaTeX

Pros and cons of `beamer`: those of LaTeX.

- Easy to create nice-looking documents (adapted to technical presentations)
- Difficult to customize

`beamer` is a class and everything works the same way as in any other LaTeX document. This presentation will focus only on the distinctive features.

Basic structure of a document

1. The easiest way to start is to use the default template (we will see how to change it later).
2. Note that you may have to run the code twice (TOC, LOF, . . .).

Basic structure of a document

1. The easiest way to start is to use the default template (we will see how to change it later).
2. Note that you may have to run the code twice (TOC, LOF, ...).
3. The basic structure is the standard in LaTeX. But:
   - Indicate that this document is of the `beamer` class.

Example of a document

```
\documentclass{beamer}
% your options here

\begin{document}

\end{document}
```

Basic structure of a document

1. The easiest way to start is to use the default template (we will see how to change it later).
2. Note that you may have to run the code twice (TOC, LOF, ...).
3. The basic structure is the standard in LaTeX. But:
   - Indicate that this document is of the beamer class.
   - Declare each slide (*frame*) you want to create.

Example of a document

```
\documentclass{beamer}
% your options here

\begin{document}

\begin{frame}

% One slide
\end{frame}

\end{document}
```

Basic structure of a document

1. The easiest way to start is to use the default template (we will see how to change it later).
2. Note that you may have to run the code twice (TOC, LOF, ...).
3. The basic structure is the standard in LaTeX. But:
   - Indicate that this document is of the beamer class.
   - Declare each slide (*frame*) you want to create.

Example of a document

```
\documentclass{beamer}
% your options here

\begin{document}

\begin{frame}
  \frametitle{Title of your slide}
% One slide
\end{frame}

\end{document}
```

Title page

- Introduce information for
  - `title`
  - `subtitle`
  - `author`
  - `institute`
  - `date`

Example of a document

```
\documentclass{beamer}
% your options here
\title{Presentations in \LaTeX{}}
\subtitle{Introduction to beamer}
\author{Gonzalo Rivero}
\date{April, 14, 2009}

\begin{document}




\begin{frame}
  \frametitle{Title of your slide}
% One slide
\end{frame}

\end{document}
```

Title page

- Introduce information for
  - `title`
  - `subtitle`
  - `author`
  - `institute`
  - `date`
- Explicitly create one slide for the `titlepage`

Example of a document

```
\documentclass{beamer}
% your options here
\title{Presentations in \LaTeX{}}
\subtitle{Introduction to beamer}
\author{Gonzalo Rivero}
\date{April, 14, 2009}

\begin{document}

\begin{frame}
  \titlepage
\end{frame}

\begin{frame}
  \frametitle{Title of your slide}
% One slide
\end{frame}

\end{document}
```

Sections of the document

Same for a table of contents. Just create a new slide with the command `\tableofcontents` and split the document using the commands `\section{`*name*`}` and `\subsection{`*name*`}`. The dynamic table of contents in the upper bar will be shown anyway.

Code

```
% ... The preamble here

\begin{document}
\begin{frame}
  \titlepage
\end{frame}

\section{Title of the section}



\begin{frame}
  \frametitle{Title of your slide}
% One slide
\end{frame}



\begin{frame}
  \frametitle{Title of your slide}
% Another slide
\end{frame}

\end{document}
```

Sections of the document

Same for a table of contents. Just create a new slide with the command `\tableofcontents` and split the document using the commands `\section{`*name*`}` and `\subsection{`*name*`}`. The dynamic table of contents in the upper bar will be shown anyway.

Code

```
% ... The preamble here

\begin{document}
\begin{frame}
  \titlepage
\end{frame}

\section{Title of the section}
\subsection{Title of the subsection}


\begin{frame}
  \frametitle{Title of your slide}
% One slide
\end{frame}

\subsection{Title of the subsection}

\begin{frame}
  \frametitle{Title of your slide}
% Another slide
\end{frame}

\end{document}
```

Environments: Columns

We can type our slides using the typical LaTeX structure. To organize the information we have two specific environments that are specific to `beamer`.

- *Columns*. Breaks the frame horizontally. Declare the environment and specify the width of the column.
- *Blocks*. Encloses the text in a colored framework with a title. A title is required (may be blank)

Code

```
\begin{frame}
  \frametitle{Frame title}

\end{frame}
```

Environments: Columns

We can type our slides using the typical LaTeX structure. To organize the information we have two specific environments that are specific to `beamer`.

- *Columns*. Breaks the frame horizontally. Declare the environment and specify the width of the column.
- *Blocks*. Encloses the text in a colored framework with a title. A title is required (may be blank)

Code

```
\begin{frame}
  \frametitle{Frame title}
    \begin{columns}




    \end{columns}
\end{frame}
```

Environments: Columns

We can type our slides using the typical LaTeX structure. To organize the information we have two specific environments that are specific to `beamer`.

- *Columns*. Breaks the frame horizontally. Declare the environment and specify the width of the column.
- *Blocks*. Encloses the text in a colored framework with a title. A title is required (may be blank)

Code

```
\begin{frame}
  \frametitle{Frame title}
    \begin{columns}
      \column{.5\textwidth}



      \column{.5\textwidth}



    \end{columns}
\end{frame}
```

Environments: Columns

We can type our slides using the typical LaTeX structure. To organize the information we have two specific environments that are specific to beamer.

- *Columns*. Breaks the frame horizontally. Declare the environment and specify the width of the column.
- *Blocks*. Encloses the text in a colored framework with a title. A title is required (may be blank)

Code

```
\begin{frame}
  \frametitle{Frame title}
    \begin{columns}
      \column{.5\textwidth}

Text for your first column

      \column{.5\textwidth}

Text for your second column

    \end{columns}
\end{frame}
```

Environment: Blocks

Block title

This is a block in blue

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{block}{Block title}
  This is a block in blue
\end{block}

\end{frame}
```

Environment: Blocks

Block title

This is a block in blue

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{block}{Block title}
  This is a block in blue
\end{block}

\end{frame}
```

Environment: Blocks

Block title

This is a block in blue

Alert-block title

This is a block in red

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{block}{Block title}
  This is a block in blue
\end{block}

\begin{alertblock}{Alert-block title}
  This is a block in red
\end{alertblock}




\end{frame}
```

Environment: Blocks

Block title

This is a block in blue

Alert-block title

This is a block in red

Example-block title

This is a block in green

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{block}{Block title}
  This is a block in blue
\end{block}

\begin{alertblock}{Alert-block title}
  This is a block in red
\end{alertblock}

\begin{exampleblock}{Example-block title}

  This is a block in green
\end{exampleblock}

\end{frame}
```

Environment: Theorem/Proof

The environments for mathematical statements are blocks:

Theorem (APS, 1989)

$\mathcal{E}^p \subset \mathbb{R}^n$ is compact.

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{theorem}[APS, 1989]
  \mathcal{E}^p\subset \R^n  is compact.
\end{theorem}

\end{frame}
```

Environment: Theorem/Proof

The environments for mathematical statements are blocks:

Theorem (APS, 1989)

$\mathcal{E}^p \subset \mathbb{R}^n$ is compact.

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{theorem}[APS, 1989]
  \mathcal{E}^p\subset \R^n  is compact.
\end{theorem}

\end{frame}
```

Environment: Theorem/Proof

The environments for mathematical statements are blocks:

Theorem (APS, 1989)

$\mathcal{E}^p \subset \mathbb{R}^n$ *is compact.*

Proof.

$\mathcal{E}^p \subset co(\pi(s)) \Rightarrow \mathcal{E}^p$ is bounded $\Rightarrow \mathcal{E}^p \subset cl(\mathcal{E}^p)$. Then $\mathcal{E}^p = B(\mathcal{E}^p) \subset B(cl(\mathcal{E}^p)) \Rightarrow cl(\mathcal{E}^p) \subset B(\mathcal{E}^p) \subset \mathcal{E}^p \Rightarrow \mathcal{E}^p$ is closed. □

Code

```
\begin{frame}
  \frametitle{Frame title}

\begin{theorem}[APS, 1989]
  \mathcal{E}^p\subset \R^n  is compact.
\end{theorem}

\begin{proof}
  $\mathcal{E}^p \subset co(\pi(s))
\Rightarrow \mathcal{E}^p$ is
bounded $\Rightarrow \mathcal{E}^p
\subset cl(\mathcal{E}^p)$. Then
$\mathcal{E}^p = B(\mathcal{E}^p)
\subset B(cl(\mathcal{E}^p))
\Rightarrow cl(\mathcal{E}^p) \subset
B(\mathcal{E}^p) \subset \mathcal{E}^p
\Rightarrow \mathcal{E}^p$ is closed.
\end{proof}

\end{frame}
```

Navigating with hyperlinks

You can create links between pages: a button that can take you to a prespecified point in the presentation.

▸ The text you want

Code

```
\begin{frame}
  \frametitle{The origin frame}
  \hyperlink{jumptable}{
  \beamergotobutton{The text you want}}
\end{frame}

\begin{frame}
  \frametitle{The target frame}
  \hypertarget{jumptable}{}
\end{frame}
```

Navigating with hyperlinks

You can create links between pages: a button that can take you to a prespecified point in the presentation.

▸ The text you want

Code

```
\begin{frame}
  \frametitle{The origin frame}
  \hyperlink{jumptable}{
  \beamergotobutton{The text you want}}
\end{frame}

\begin{frame}
  \frametitle{The target frame}
  \hypertarget{jumptable}{}
\end{frame}
```

Navigating with hyperlinks

You can create links between pages: a button that can take you to a prespecified point in the presentation.

▸ The text you want

Other styles:

- `beamerbutton`: here
- `beamerskipbutton`: ▸▸ here
- `beamerreturnbutton`: ◂ here

Code

```
\begin{frame}
  \frametitle{The origin frame}
  \hyperlink{jumptable}{
  \beamergotobutton{The text you want}}
\end{frame}

\begin{frame}
  \frametitle{The target frame}
  \hypertarget{jumptable}{}
\end{frame}
```

Itemize/Enumerate

`itemize` and `enumerate` work as expected:

- First element
- Second element
- Third element

Code

```
\begin{itemize}
  \item First element
  \item Second element
  \item Third element
\end{itemize}
```

Nevertheless, it might be useful to uncover lines in a given order: *overlays* in `beamer` jargon. For instance, ...

Items and overlays

- This item first

Code

```
\begin{itemize}
  \item<1-> This item first
  \item<3-> This item third
  \item<2-> This item second
\end{itemize}
```

Items and overlays

- This item first
- This item second

Code

```
\begin{itemize}
  \item<1-> This item first
  \item<3-> This item third
  \item<2-> This item second
\end{itemize}
```

Items and overlays

- This item first
- This item third
- This item second

Code

```
\begin{itemize}
  \item<1-> This item first
  \item<3-> This item third
  \item<2-> This item second
\end{itemize}
```

Items and overlays

- This item first
- This item third
- This item second

Code

```
\begin{itemize}
  \item<1-> This item first
  \item<3-> This item third
  \item<2-> This item second
\end{itemize}
```

Note that the order is given by

- <1> Show *only* on slide 1
- <1-> Show on slide 1 *onwards*
- <1-4,6-8> Show on every slide except 5
- \pause Creates stopping points (useful for tables)

Emphasis in overlays

We can emphasize portions of our slide using alerts. Alerts can use overlays.

- This item first

Code

```
\begin{itemize}
  \item<1-> \alert<1>{This item first}
  \item<3-> \textsl<3>{This item third}
  \item<2-> \textbf<2>{This item second}
  \item<4-> \color<4>{blue}{Finally...}
\end{itemize}
```

Emphasis in overlays

We can emphasize portions of our slide using alerts. Alerts can use overlays.

- This item first
- **This item second**

Code

```
\begin{itemize}
  \item<1-> \alert<1>{This item first}
  \item<3-> \textsl<3>{This item third}
  \item<2-> \textbf<2>{This item second}
  \item<4-> \color<4>{blue}{Finally...}
\end{itemize}
```

Emphasis in overlays

We can emphasize portions of our slide using alerts. Alerts can use overlays.

- This item first
- *This item third*
- This item second

Code

```
\begin{itemize}
  \item<1-> \alert<1>{This item first}
  \item<3-> \textsl<3>{This item third}
  \item<2-> \textbf<2>{This item second}
  \item<4-> \color<4>{blue}{Finally...}
\end{itemize}
```

Emphasis in overlays

We can emphasize portions of our slide using alerts. Alerts can use overlays.

- This item first
- This item third
- This item second
- Finally...

Code

```
\begin{itemize}
  \item<1-> \alert<1>{This item first}
  \item<3-> \textsl<3>{This item third}
  \item<2-> \textbf<2>{This item second}
  \item<4-> \color<4>{blue}{Finally...}
\end{itemize}
```

More on overlays

1. First argument

Code

```
\begin{enumerate}[<+-| alert@+>]
  \item First argument
  \item Second argument
  \item Third argument
  \item Fourth argument
\end{enumerate}
```

More on overlays

1. First argument
2. Second argument

Code

```
\begin{enumerate}[<+-| alert@+>]
  \item First argument
  \item Second argument
  \item Third argument
  \item Fourth argument
\end{enumerate}
```

More on overlays

1. First argument
2. Second argument
3. Third argument

Code

```
\begin{enumerate}[<+-| alert@+>]
  \item First argument
  \item Second argument
  \item Third argument
  \item Fourth argument
\end{enumerate}
```

More on overlays

1. First argument
2. Second argument
3. Third argument
4. Fourth argument

Code

```
\begin{enumerate}[<+-| alert@+>]
  \item First argument
  \item Second argument
  \item Third argument
  \item Fourth argument
\end{enumerate}
```

Dynamic displays of tables: rowwise

We can use overlays with tables to show them row- or columnwise

	Mean	Sd. Dev.	95% HPD
μ_1	1.220	0.303	[0.567, 1.821]

Dynamic displays of tables: rowwise

We can use overlays with tables to show them row- or columnwise

	Mean	Sd. Dev.	95% HPD	
μ_1	1.220	0.303	[0.567,	1.821]
μ_2	2.676	0.409	[1.863,	3.498]

Dynamic displays of tables: rowwise

We can use overlays with tables to show them row- or columnwise

	Mean	Sd. Dev.	95% HPD	
μ_1	1.220	0.303	[0.567,	1.821]
μ_2	2.676	0.409	[1.863,	3.498]
ρ	0.313	0.264	[-0.295,	0.749]

Dynamic displays of tables: rowwise

We can use overlays with tables to show them row- or columnwise

	Mean	Sd. Dev.	95% HPD	
μ_1	1.220	0.303	[0.567,	1.821]
μ_2	2.676	0.409	[1.863,	3.498]
ρ	0.313	0.264	[-0.295,	0.749]

Code (Approximate)

```
\begin{table}[!h]
  \centering
    \begin{tabular}{l|cccc}
                & Mean     & Sd. Dev. & 95\% HPD       \\ \hline
      $\mu_1$   & 1.220    & 0.303    & [0.567, 1.821]  \pause\\
      $\mu_2$   & 2.676    & 0.409    & [1.863, 3.498]  \pause\\
      $\rho$    & 0.313    & 0.264    & [-0.295, 0.749]
    \end{tabular}
\end{table}
```

Dynamic displays of tables: columnwise

	M1
β	1.11
σ	4.44
θ	7.77

Code

```
\begin{table}[!h]
  \centering
    \begin{tabular}
      {lc<{\onslide<2->}c<{\onslide<3->}c<{\onslide}}
                    & M1    & M2    & M3    \\ \hline
          $\beta$   & 1.11  & 2.22  & 3.33  \\
          $\sigma$  & 4.44  & 5.55  & 6.66  \\
          $\theta$  & 7.77  & 8.88  & 9.99
\end{tabular}
\end{table}
```

Dynamic displays of tables: columnwise

	M1	M2
β	1.11	2.22
σ	4.44	5.55
θ	7.77	8.88

Code

```
\begin{table}[!h]
  \centering
    \begin{tabular}
      {lc<{\onslide<2->}c<{\onslide<3->}c<{\onslide}}
                    & M1    & M2    & M3    \\ \hline
          $\beta$   & 1.11  & 2.22  & 3.33  \\
          $\sigma$  & 4.44  & 5.55  & 6.66  \\
          $\theta$  & 7.77  & 8.88  & 9.99
\end{tabular}
\end{table}
```

Dynamic displays of tables: columnwise

	M1	M2	M3
β	1.11	2.22	3.33
σ	4.44	5.55	6.66
θ	7.77	8.88	9.99

Code

```
\begin{table}[!h]
  \centering
    \begin{tabular}
      {lc<{\onslide<2->}c<{\onslide<3->}c<{\onslide}}
                    & M1    & M2    & M3    \\ \hline
          $\beta$   & 1.11  & 2.22  & 3.33  \\
          $\sigma$  & 4.44  & 5.55  & 6.66  \\
          $\theta$  & 7.77  & 8.88  & 9.99
\end{tabular}
\end{table}
```

Including code

There are different ways to insert code. beamer incorporates the semiverbatim environment. It needs the option fragile in the definition of the frame.

```
foo <- function(x, ...) {
  if (x < 0) print('Pos.')
  if (x > 0) print('Neg.')
}
```

Code

```
\begin{frame}[fragile]
\begin{semiverbatim}
  foo <- function(x, ...) \{
    if (x < 0) print('Neg.')
    if (x > 0) print('Pos.')
  \}
\end{semiverbatim}
\end{frame}
```

Note that semiverbatim can run LaTeX commands.
If we want to print them in the screen we have to kill them (and the brackets) using \

Including code

In the previous cases, text appeared from invisibility. Here we allowing it to be "transparent".

```
foo <- function(x, ...) {
  if (x < 0) print('Neg.')
  if (x > 0) print('Pos.')
}
```

Code

```
\begin{frame}[fragile]
\begin{semiverbatim}
\setbeamercovered{transparent}
  \uncover<1->{foo <- function(x, ...) \{}
    \uncover<2->{if (x < 0) print('Neg.')}
    \uncover<3->{if (x > 0) print('Pos.')}
  \}
\end{semiverbatim}
\end{frame}
```

Including code

In the previous cases, text appeared from invisibility. Here we allowing it to be "transparent".

```
foo <- function(x, ...) {
  if (x < 0) print('Neg.')
  if (x > 0) print('Pos.')
}
```

Code

```
\begin{frame}[fragile]
\begin{semiverbatim}
\setbeamercovered{transparent}
  \uncover<1->{foo <- function(x, ...) \{}
    \uncover<2->{if (x < 0) print('Neg.')}
    \uncover<3->{if (x > 0) print('Pos.')}
  \}
\end{semiverbatim}
\end{frame}
```

Including code

In the previous cases, text appeared from invisibility. Here we allowing it to be "transparent".

```
foo <- function(x, ...) {
  if (x < 0) print('Neg.')
  if (x > 0) print('Pos.')
}
```

Code

```
\begin{frame}[fragile]
\begin{semiverbatim}
\setbeamercovered{transparent}
  \uncover<1->{foo <- function(x, ...) \{}
    \uncover<2->{if (x < 0) print('Neg.')}
    \uncover<3->{if (x > 0) print('Pos.')}
  \}
\end{semiverbatim}
\end{frame}
```

Including code

In the previous cases, text appeared from invisibility. Here we allowing it to be "transparent".

```
foo <- function(x, ...) {
  if (x < 0) print('Neg.')
  if (x > 0) print('Pos.')
}
```

Code

```
\begin{frame}[fragile]
\begin{semiverbatim}
\setbeamercovered{transparent}
  \uncover<1->{foo <- function(x, ...) \{}
    \uncover<2->{if (x < 0) print('Neg.')}
    \uncover<3->{if (x > 0) print('Pos.')}
  \}
\end{semiverbatim}
\end{frame}
```

We could use `uncover` in combination with `alert` for emphasis

Size of the slide

Remember that the size of a `beamer` slide is 128mm $\times$ 96mm ($\sim$ 5.03in $\times$ 3.77in). These dimensions are fixed and *should not be changed*.

To squeeze extra material into a Beamer slide, you may specify a shrink factor:

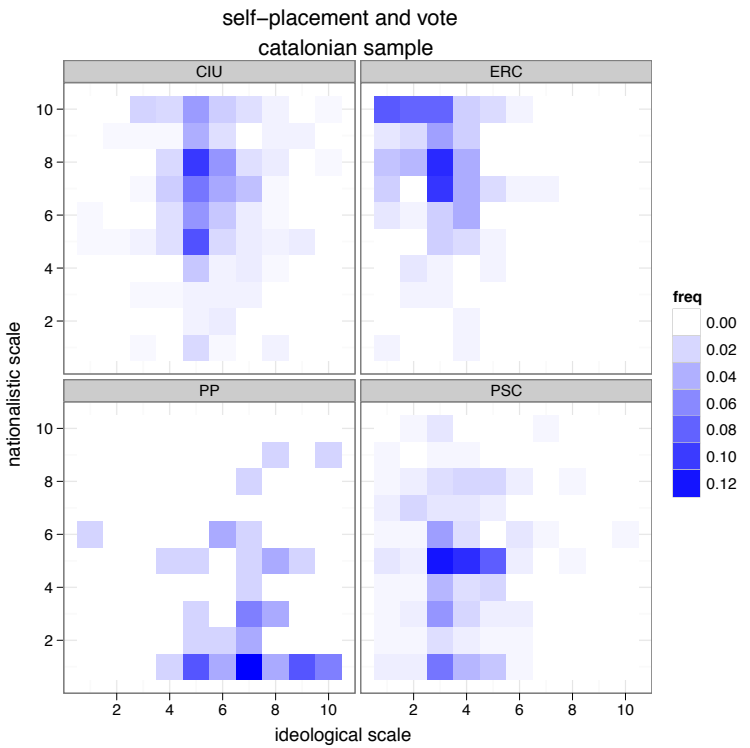
Code

```
\begin{frame}[shrink = 5]
  % Text of the slide
\end{frame}
```

but it is usually better to rewrite the slide.

Figures

beamer supports .png, .jpeg, .jpg, and .pdf files using the package graphicx that is launched by default with beamer. For .pdf files (strongly recommended) use the option pdftex.



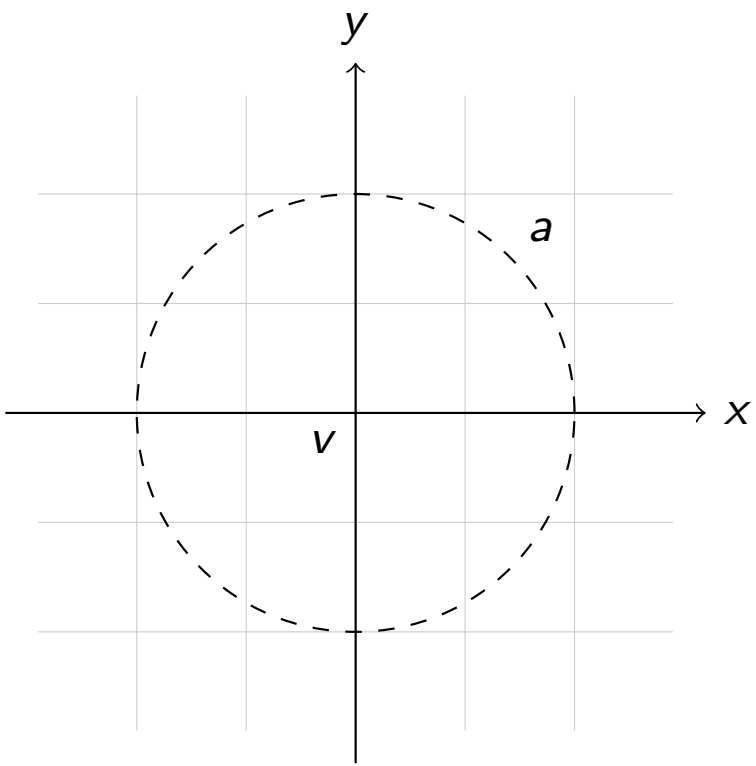
Code

```
\usepackage[pdftex]{graphicx}

% ... Document ...

\begin{frame}
  \frametitle{Figures}

  \begin{figure}[!t]
    \includegraphics[height = 5cm]{p.pdf}
  \end{figure}

\end{frame}
```

Remember the size of the slide: this figure is just 5cm×5cm

Diagrams

It is "easy" to draw diagrams using PGF/TikZ (similar to PSTricks).

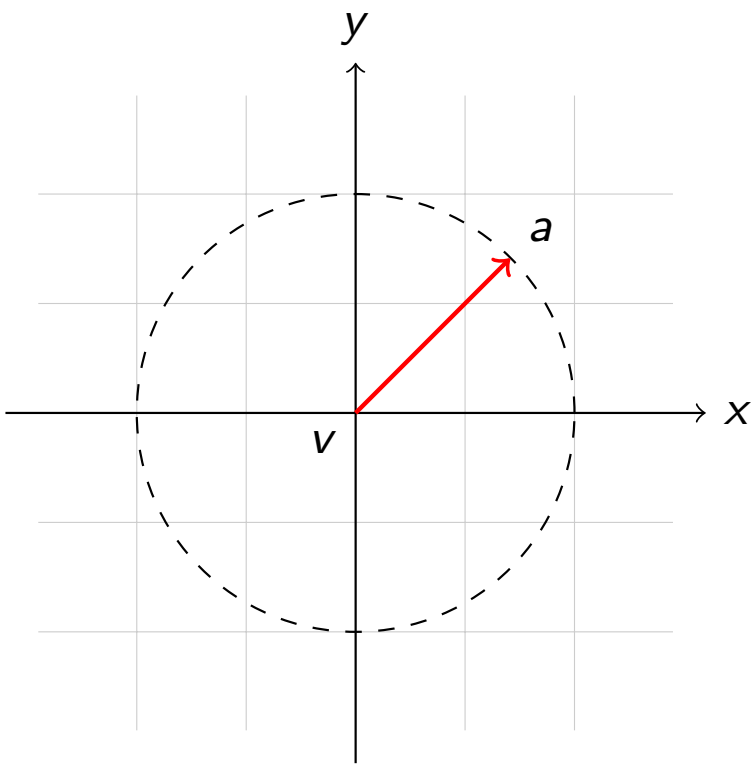


Code

```
\usepackage{tikz}

% ... Document ...

\begin{figure}
\begin{tikzpicture}[domain=-4:4,scale=.6]
\draw[very thin,color=black!20]
    (-2.9,-2.9) grid (2.9,2.9);
\draw[->] (-3.2,0) -- (3.2,0) node[right]
    {$x$};
\draw[->] (0,-3.2) -- (0,3.2) node[above]
    {$y$};

\draw (0,0) node[below left] {$v$};
\draw (1.41,1.41) node[above right]{$a$};

\draw[dashed] (0,0) circle (2cm);
\uncover<2->{\draw [red, ->, thick]
    (0,0) -- (1.41,1.41);}
\end{tikzpicture}
\end{figure}
```

Diagrams

It is "easy" to draw diagrams using PGF/TikZ (similar to PSTricks).
You can use overlays.



Code

```
\usepackage{tikz}

% ... Document ...

\begin{figure}
\begin{tikzpicture}[domain=-4:4,scale=.6]
\draw[very thin,color=black!20]
    (-2.9,-2.9) grid (2.9,2.9);
\draw[->] (-3.2,0) -- (3.2,0) node[right]
    {$x$};
\draw[->] (0,-3.2) -- (0,3.2) node[above]
    {$y$};

\draw (0,0) node[below left] {$v$};
\draw (1.41,1.41) node[above right]{$a$};

\draw[dashed] (0,0) circle (2cm);
\uncover<2->{\draw [red, ->, thick]
    (0,0) -- (1.41,1.41);}
\end{tikzpicture}
\end{figure}
```

Movies

Movies can be included in `.mpg` or `.mov` format. We need package `movie15`. Not all PDF readers support multimedia.

(Loading...)

Code

```
\usepackage{movie15}


% ... Document ...

\begin{frame}
  \frametitle{Figures}

  \begin{figure}
  \frametitle{Figures}
    \includemovie[poster,
      text={\small(Loading...)}]
      {4cm}{6cm}{plot.mov}
  \end{figure}

\end{frame}
```

Themes: Frankfurt

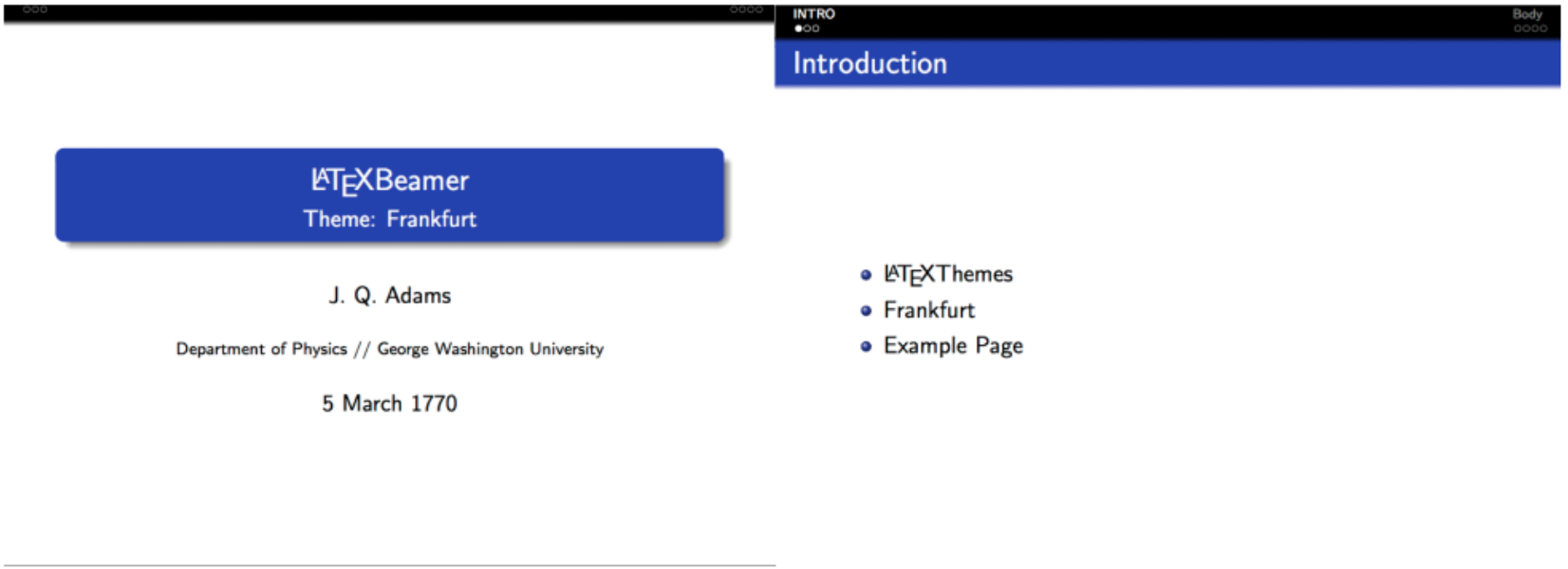


Figure: Frankfurt Theme

Themes: Boadilla

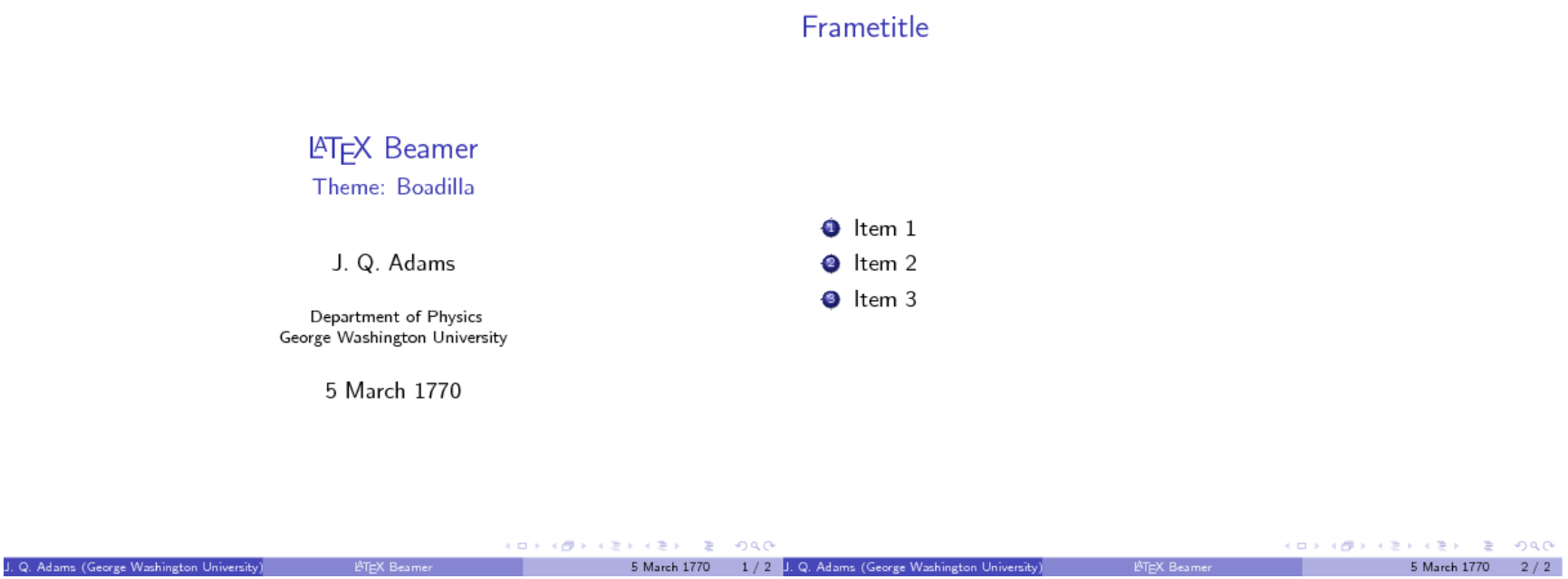


Figure: Boadilla Theme

Themes: Montpellier

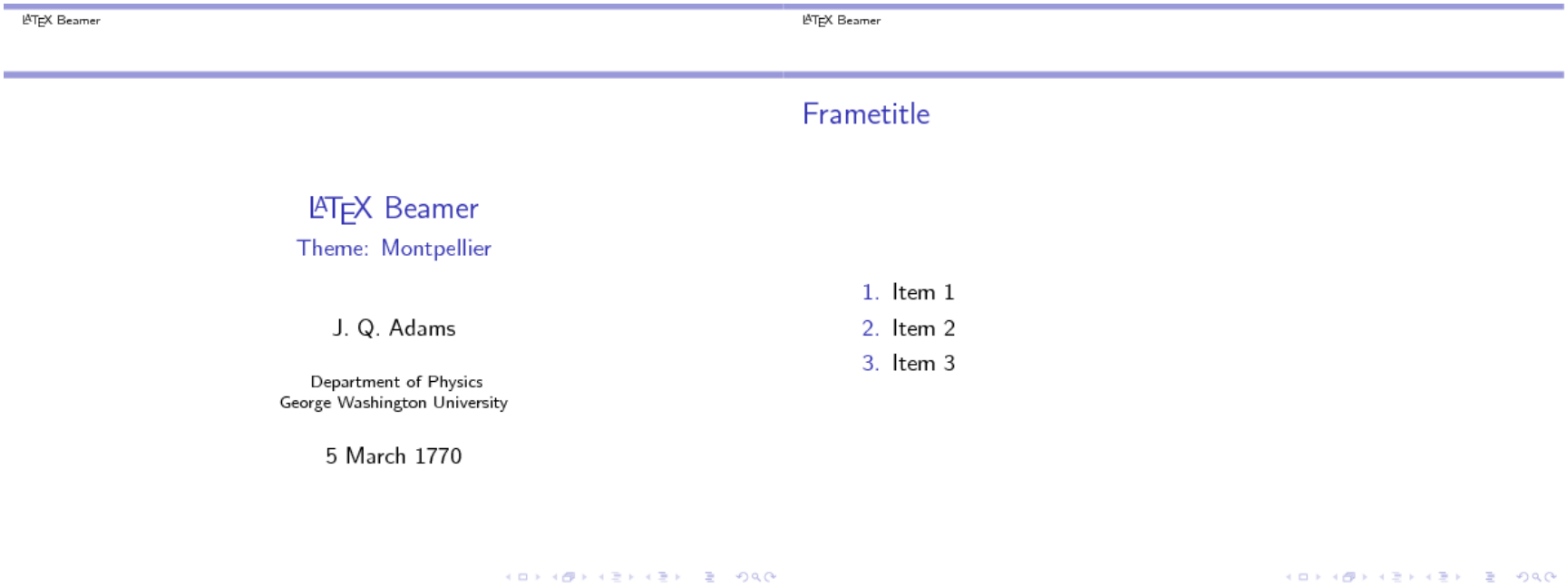


Figure: Montpellier Theme

Themes: Goettingen

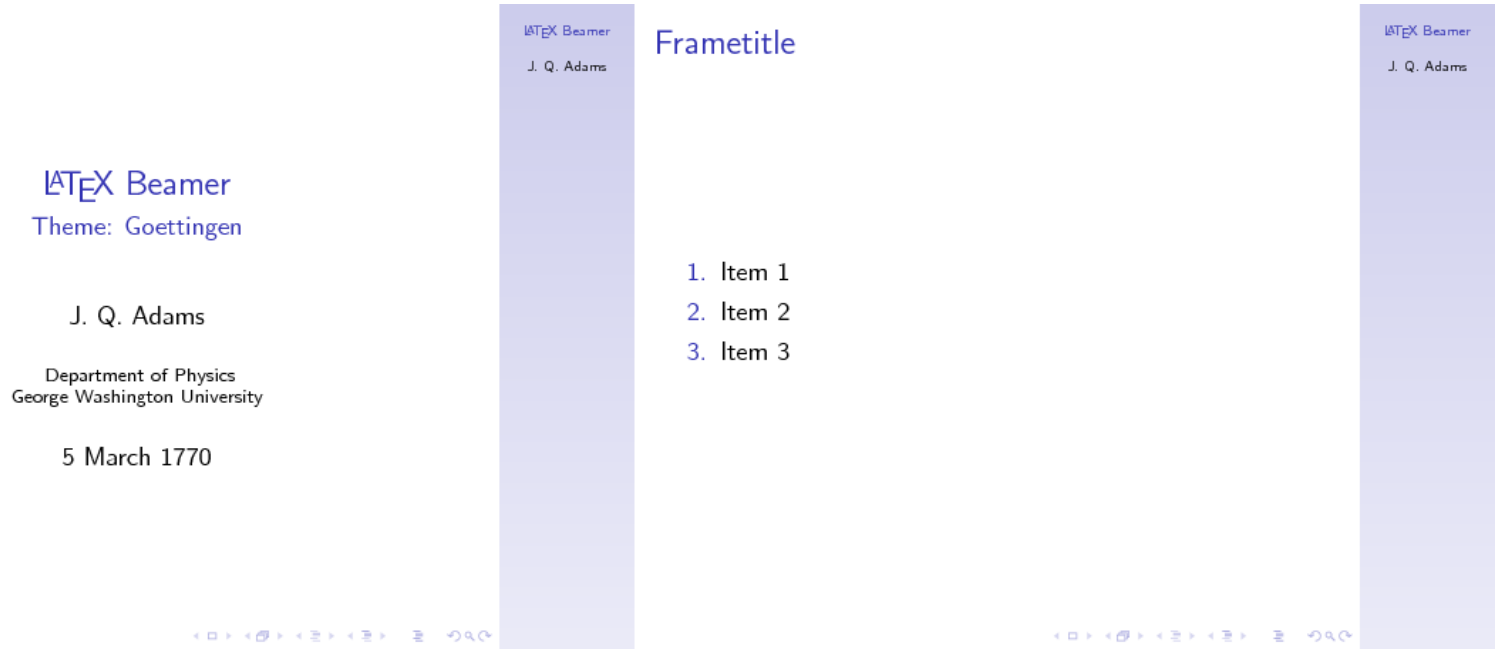


Figure: Goettingen Theme

Themes: PaloAlto

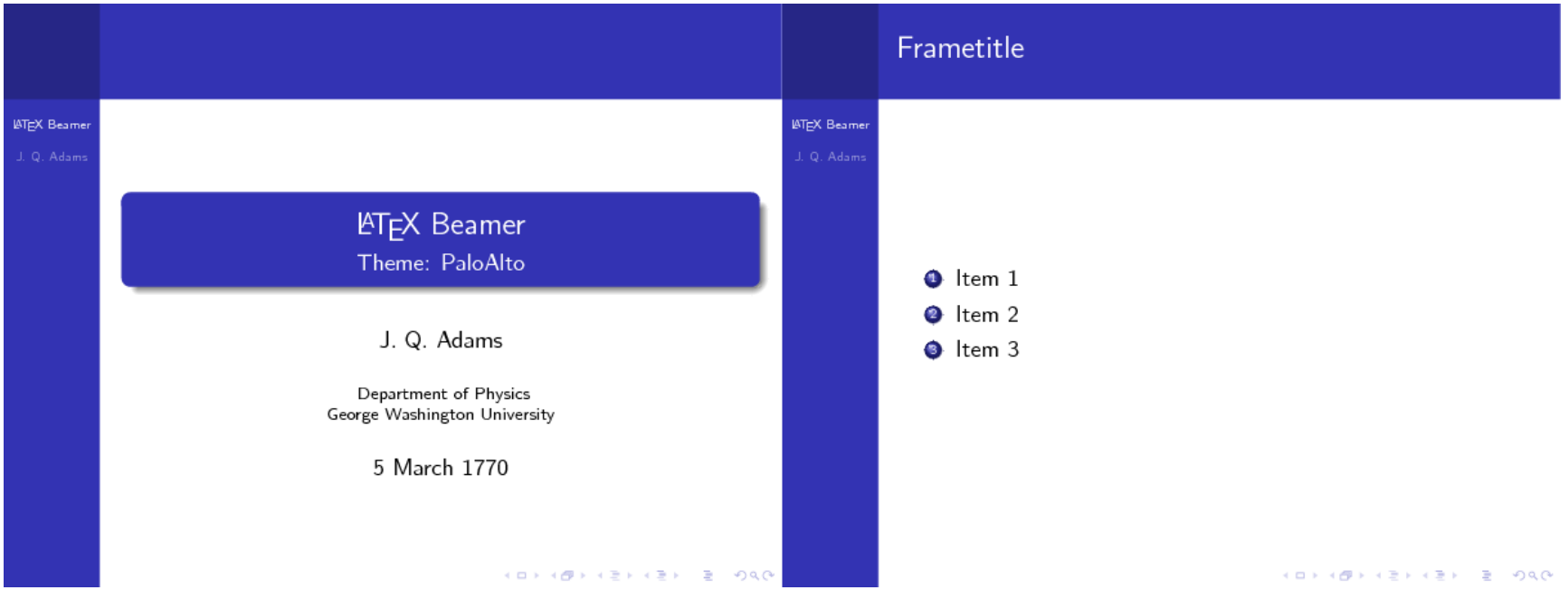


Figure: Palo Alto Theme

Changing templates and colors

\usetheme{Warsaw} or...

Antibes Bergen Berkeley Berlin Boadilla Copenhagen Darmstadt Dresden Frankfurt Goettingen Hannover Ilmenau Juanlespins Madrid Malmoe Marburg Montpellier PaloAlto Pittsburgh Rochester Singapore Szeged Warsaw boxes default

\usecolortheme{default} or...

albatross crane beetle dove fly seagull wolverine beaver

Theme tuning

Inner elements, like blocks:

\usecolortheme{lily} or...

lily orchid rose

\useinnertheme{rectangles} or...

rectangles circles inmargin rounded

Outter elements, like headline and footline:

\usecolortheme{whale} or...

whale seahorse dolphin

\useoutertheme{infoline} or...

infoline miniframes shadow sidebar smoothbars smoothtree split tree

A few common options

Font themes

```
\usepackage{helvet}  % Font families
\usefonttheme{serif}  % For the structural elements
```

Remove navigation bar

```
\setbeamertemplate{navigation symbols}{}
```

Slide numbers

```
\setbeamertemplate{footline}[slide number]  % Typically
\insertframenumber/\inserttotalframenumber  % To insert them in specific places
```

Style of each element (check the documentation for the full list)

```
\setbeamertemplate{itemize items}[triangle]
\setbeamertemplate{blocks}[shadow=false]
```

Color and font of each element (check the documentation for the full list)

```
\setbeamercolor{background canvas}{bg=white}
\setbeamerfont{title}{family=\rm}
```

Transitions: example of transblindshorizontal

We can use dynamic transitions between frames. It only works with Adobe Acrobat!

Code

```
\begin{frame}
\frametitle{Example of transblindshorizontal}
  \transblindshorizontal

% Your slide
\endframe
```

Quousque tandem abutere, Catilina, patientia nostra? Quam diu etiam furor iste tuus nos eludet? Quem ad finem sese effrenata iactabit audacia? Nihilne te nocturnum praesidium Palati, nihil urbis vigiliae, nihil timor populi, nihil concursus bonorum omnium, nihil hic munitissimus habendi senatus locus, nihil horum ora voltusque moverunt? Patere tua consilia non sentis, constrictam iam horum omnium scientia teneri conjurationem tuam non vides? Quid proxima, quid superiore nocte egeris, ubi fueris, quos convocaveris, quid consilii ceperis, quem nostrum ignorare arbitraris?

Transitions: example of transboxin

Slide transitions are overlay specification aware, so \transboxin<2> will cause the second slide of the frame to use the transboxin effect.

Code

```
\begin{frame}
\frametitle{Example of transboxin}
  \transboxin

  \uncover<2>This text will be uncovered with a transition
\endframe
```

Transitions: example of transboxin

Slide transitions are overlay specification aware, so `\transboxin<2>` will cause the second slide of the frame to use the transboxin effect.

Code

```
\begin{frame}
\frametitle{Example of transboxin}
  \transboxin

  \uncover<2>This text will be uncovered with a transition
\endframe
```

Tutti gli Stati, tutti i domini che hanno avuto ed hanno imperio sopra gli uomini sono stati e sono o repubbliche. I principati sono o ereditari, dé quali il sangue del loro signore ne sia stato lungo tempo principe, o é sono nuovi. I nuovi o sono nuovi tutti, come fu Milano a Francesco Sforza, o é sono come membri aggiunti allo Stato ereditario del principe che gli acquista, come è il regno di Napoli al re di Spagna.

Transtitions

Options

`\transblindshorizontal`	Horizontal blinds pulled away
`\transblindsvertical`	Vertical blinds pulled
`\transboxin`	Move to center from all sides
`\transboxout`	Move to all sides from center
`\transdissolve`	Slowly dissolve what was shown before
`\transglitter`	Glitter sweeps in specified direction
`\transslipverticalin`	Sweeps two vertical lines in
`\transslipverticalout`	Sweeps two vertical lines
`\transhorizontalin`	Sweeps two horizontal lines in
`\transhorizontalout`	Sweeps two horizontal lines out
`\transwipe`	Sweeps single line in specified direction

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

om the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

rs from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Code (from Ki-Joo Kim)

```
\begin{frame}
  \animate<2-15, 17-32>
  \frametitle{Animations}
    \begin{itemize}
      \item[]\animatevalue<2-15>{\opaqueness}{0}{100}
        \begin{colormixin}
          {\the\opaqueness!averagebackgroundcolor}This text appears progressively.
        \end{colormixin}

      \item[] \animatevalue<17-32>{\opaqueness}{0}{100}
      \animatevalue<17-32>{\offset}{-5cm}{5cm}
        \begin{colormixin}{\the\opaqueness!averagebackgroundcolor}
          \hspace{\offset}This text appears from the left.
        \end{colormixin}
    \end{itemize}
\end{frame}
```

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Code (from Ki-Joo Kim)

```
\begin{frame}
  \animate<2-15, 17-32>
  \frametitle{Animations}
    \begin{itemize}
      \item[]\animatevalue<2-15>{\opaqueness}{0}{100}
        \begin{colormixin}
          {\the\opaqueness!averagebackgroundcolor}This text appears progressively.
        \end{colormixin}

      \item[] \animatevalue<17-32>{\opaqueness}{0}{100}
      \animatevalue<17-32>{\offset}{-5cm}{5cm}
        \begin{colormixin}{\the\opaqueness!averagebackgroundcolor}
          \hspace{\offset}This text appears from the left.
        \end{colormixin}
    \end{itemize}
\end{frame}
```

Animations

Animations consume a lot of frames, and they are difficult to program. Only with Adobe Acrobat.

This texts appears progressively.

This text enters from the left.

Code (from Ki-Joo Kim)

```
\begin{frame}
  \animate<2-15, 17-32>
  \frametitle{Animations}
    \begin{itemize}
      \item[]\animatevalue<2-15>{\opaqueness}{0}{100}
        \begin{colormixin}
          {\the\opaqueness!averagebackgroundcolor}This text appears progressively.
        \end{colormixin}

      \item[] \animatevalue<17-32>{\opaqueness}{0}{100}
      \animatevalue<17-32>{\offset}{-5cm}{5cm}
        \begin{colormixin}{\the\opaqueness!averagebackgroundcolor}
          \hspace{\offset}This text appears from the left.
        \end{colormixin}
    \end{itemize}
\end{frame}
```

Notes on animation

Two basic commands for making animations:

1. `\animate<n>` to animate *n* slides
2. `\animatevalue<n>{name}{start}{end}` for specifying animation effects
   - name: counter of the dimension
   - start and end values of the value

We specify which variable we want to change (above, *opaqueness* and *offset*) and the sequence we want to use (0 to 100 opaqueness, -5cm to 5cm offset). Changing the color requires the environment `colormixin`

Resources

Where to find more information?

- The excellent documentation of the package
- A previsualization of the `beamer` themes (Mike Depalatis)
- The collection of `beamer` color palettes (JiHO)
- The `beamer` v3.0 Guide (Ki-Joo Kim)
- A `beamer` tutorial in `beamer` (Charles T. Batts)