Intermediate Latex
Introduction to Beamer

Jean Hare

Sorbonne Université
Laboratoire Kastler Brossel
ED Physique en Île-de-France
jean.hare@lkb.ens.fr

Support documents at https://www.edpif.org/documents/latex/intermediate

November 2023

Summary

Summary

Introduction to Beamer

- Beamer is a LaTeX document class dedicated to the creation of presentations or slideshows. It was created in 2003 by Till TANTAU, the same author as for PGF/TikZ, and shares many features with them.
- It supersedes the older classes like SliTeX, seminar, prosper , powerdot, etc
- It is mostly intended to be used with pdfLateX to produce PDF presentations, but also works with traditional LaTeX route
- Beside Portability of PDF format, the most prominent and interesting features are:
  - Handling of layers for progressive display of the same slide
  - Ability to embed many kinds of multimedia content (with companion packages).
  - Automatic creation of an handout (flatten version of the presentation)
  - All (almost) the LaTeX formatting tools (namely math)
- Drawbacks: requires compilation, type code instead to click.

The basic presentation structure

- Uses class `beamer`
- Slides defined by environment `frame`
- Sectioning commands: only
  \section and \subsection

  Minimal code:

```
1  \begin{document}
2  \begin{frame}{Outline}
3  \tableofcontents
4  \end{frame}
5  \section{Introduction}
6  \begin{frame}{First~slide}
7  Some content
8  \[ a^2= b^2 + c^2 \]
9  \end{frame}
10 \end{document}
```

Outline

Introduction

First slide

Some content

$$a^2 = b^2 + c^2$$

Controlling appearance

- **General defaults** The page size is typically 128mm cm×96 mm, ensuring that the usual sized fonts produce good results.
  The default `aspectratio=43`, can be changed to `aspectratio=169` (§8.3).
  The body text font is by default a sans-serif font (keep it);

- General layout of slides is defined with a (Presentation) "theme", named after a city around the world. The "theme" defaults settings can altered by "color-themes", "font-themes", "outer-themes" and "inner-themes" .

- From manual § 15.1 :
  - **Outer Themes**: what the "outside" or "border" of the presentation slides should look like: head- and footlines, logo position, sidebar/navigation symbols/bars or not. It also specifies where the frametitle is put and how it is typeset. Globally classified according to the way to display the ToC on each frame.
  - **Inner Themes**: how certain elements of a presentation are typeset. This includes all elements that are at the "inside" of the frame. This includes all enumerations, itemize,block and theorem environments, table of contents.

- This presentation uses: `\usetheme{Berlin}` `\useoutertheme{infolines}`
  `\useinnertheme{rounded}` `\usefonttheme[onlymath]{serif}`

List of standard (outer) themes
(Classified according to table of contents navigation, from beameruserguide.pdf)

without Navig. Bars
default, boxes
Bergen
Boadilla
Madrid
AnnArbor
CambridgeUS
EastLansing
Pittsburgh
Rochester

Tree-Like Navigation Bar
Antibes
JuanLesPins
Montpellier

Table of Contents Sidebar
Berkeley
PaloAlto
Goettingen
Marburg
Hannover

Mini Frame Navigation
Berlin
Ilmenau
Dresden
Darmstadt
Frankfurt
Singapore
Szeged

Gallery of styles

Some nice examples on Internet :

- https://deic-web.uab.cat/~iblanes/beamer_gallery
- http://mcclinews.free.fr/latex/beamergalerie/completsgalerie.html
- https://fr.overleaf.com/gallery/tagged/presentation

Note: If you aim to customize them, many settings must be adjusted with the generic command \setbeamertemplate (§16.3) instead of the standard LaTeX commands

Experiment with two themes and colors sets

Omitted `\documentclass{beamer}` and `\end{document}`

```
\usefonttheme[onlymath]{serif}
\usecolortheme{seahorse}
\begin{document}
\section{Section Introduction}
\subsection{First subsection}
\begin{frame}{My first slide}
Some content in theme "Berlin"
(color "seahorse")\\
\& math reset to serif
\end{frame}
\begin{frame}{My second slide}
\alert{Some other content:}
\[ a^2= b^2 + c^2 \]
\end{frame}
\subsection{Second subsection}
\begin{frame}{My third slide}
Some alternate content
\end{frame}
%further content
\section{The very nice following sectio
```

```
 1  \usecolortheme{beetle}
 2  \begin{document}
 3  \section{Section Introduction}
 4  \subsection{First subsection}
 5  \begin{frame}{My first slide}
 6  Some content in theme "Marburg"
 7  (color "beetle")\\
 8  \& math reset to serif
 9  \end{frame}
10  \begin{frame}{My second slide}
11  \alert{Some other content:}
12  \[ a^2= b^2 + c^2 \]
13  \end{frame}
14  \subsection{Second subsection}
15  \begin{frame}{My third slide}
16  Some alternate content
17  \end{frame}
18  %further content
19  \section{The very nice following secti
20  \subsection{Third subsection}
```

My first slide

Some content in theme "Berlin" (color "seahorse")
& math reset to serif

My second slide

Some other content:

$$a^2 = b^2 + c^2$$

My third slide

Some alternate content

My fourth slide

Some final content   (highlighted with \alert{})
For more about themes, see :

- The manual § 15.   (highlighted with \structure{})
- The page http://mcclinews.free.fr/latex/
  beamergalerie/completsgalerie.html.
  This page is part of a web site with a good tutoral, in French :
  http://mcclinews.free.fr/latex/introbeamer.php.
- The page
  http://deic.uab.es/~iblanes/beamer_gallery/

My fifth slide

Some ultimate content

My first slide

Some content in theme "Marburg" (color "beetle")
& math reset to serif

My second slide

Some other content:

$$a^2 = b^2 + c^2$$

My third slide

Some alternate content

My fourth slide

Some final content   ((highlighted with \alert{}))
For more about themes, see :

▶ The manual § 15.   (highlighted with (highlighted with
  \structure{})

▶ The page http://mcclinews.free.fr/latex/
  beamergalerie/completsgalerie.html.
  This page is part of a web site with a good tutoral, in
  French : http:
  //mcclinews.free.fr/latex/introbeamer.php.

▶ The page
  http://deic.uab.es/~iblanes/beamer_gallery/

My fifth slide

Some ultimate content

Metadata, Title page, Table of contents

Some important elements can be added in the following way:

Metadata (in the preamble)

```
\title[A-LaTeX]{Intermediate LaTeX for non-beginners}
\subtitle[A-LaTeX]{Hannover theme}
\institute[SU]{Sorbonne Université\\Laboratoire Kastler Brossel}
\date{\today}
\author[Jean Hare]{\href{mailto:jean.hare@lkb.ens.fr}{Jean Hare}}
```

Titlepage (using metadata):

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

Table of contents

```
\begin{frame}{Outline}
   \tableofcontents
\end{frame}
```

Intermediate LaTeX for non-beginners
Hannover theme

Jean Hare

Sorbonne Université
Laboratoire Kastler Brossel

November 11, 2023

Outline

My first slide

Some content in theme "Hannover" (defaultcolor)

Configuring Table of contents

- Table of contents can be adapted to context with options :
  `currentsection`, `currentsubsection`, `firstsection`=<num>,
  `hideallsubsections`, `hideothersubsections`, `pausesections`, etc.
  (see manual § 10.5)
- Table of content can be repeated at each section or subsection with
  the code (in preamble)

```
\AtBeginSection[]{%
\begin{frame}{Sommaire}
 \tableofcontents[currentsection]
\end{frame}
}
\AtBeginSubsection[]{%
\begin{frame}{Dans la section}
\tableofcontents[currentsection,currentsubsection]
\end{frame}
}
```

Structuring elements in frames

1. Columns (like on this frame)

```
\begin{columns}
\begin{column}{0.5\linewidth}
some content
\end{column}
\begin{column}{0.5\linewidth}
other content
\end{column}
\end{columns}
```

2. Itemize & enumerate, like here
   - Same syntax as standard LaTeX
   - Round labels are selected with
   ```
   \useinnertheme{rounded}
   ```
   - Doesn't work with `enumitem`,
     tune the lengths just before, like
   ```
   \setlength{\leftmarginii}{1.5ex}
   ```

3. Block environments:

   This is an ordinary block

   Really ordinary !

   This is an `alertblock`

   Really alert !

   This is an

   It is always useful...

   produced with:
   ```
   \begin{<blocktype>}{<blockltitle>}
   block content
   \end{<blocktype>}
   ```

Options for `\documentclass` and `frame`

\documentclass[...]{beamer}

- `handout`, `article`   (select docpument target)
- `aspectratio`
- `hyperref`={*list of hyperref's options*}
- `xcolor`={*list of xcolor's options*}

\begin{frame}[...]{*frametitle*}

- `fragile`   (enable verbatim content)
- `label`=*name*
- `plain`   (suppress outer bars)
- `noframenumbering`   (don't increment `framenumber`)
- `allowframebreaks`=*fraction*   (evil)
- `shrink`=*minimal percentage*   (evil)
- `squeeze`   (kills vertical white space, evil)

Handling frames

- To restrict the frame set, use `\includeonlyframes{label-list}` where labels were defined by `label` option of frames
- To repeat any previous frame, use `\againframe{label}`
- Page's geometry can be changed with moderation with `\setbeamersize{...}`. This command (instead of `\setlength`) can be used to alter various lengths.
- Elements or frame can be included only in different versions with : `\mode<beamer>{...}` and `\mode<handout>{...}`, like :

  ```
  \mode<handout>{\renewcommand{\alert}[1]{\textbf{#1}}}
  ```

- If a frame must not have any border elements, use option `plain`
- If a frame must have verbatim contents, use option `fragile`
- Frame transitions can be managed, with moderation (manual § 14.3)

Summary

Managing the layers

The handling of layers (called "Overlays" in the manual) is a great interest of Beamer (with pdfLaTeX or LuaLaTeX).

- Basically, if you want that a whole frame is not displayed all at once, but its points once-by-once, you will ask Beamer to splits the frame into successive slides.

- The most simple command for this purpose is the command \pause that you can insert between any paragraphs or structures.
  Notice: Doesn't work inside $\mathcal{AMS}$-math align-like blocks

- As a (dummy) example, you could write :

```
\begin{itemize}
\item 2 is prime (two divisors: 1 and 2).
\pause
\item 3 is prime (two divisors: 1 and 3).
\pause
\item 4 is not prime (\alert{three} divisors: 1, 2, and 4).
\end{itemize}
```

Example (slide 1)

in itemize (inside block)

- 2 is prime (two divisors: 1 and 2).

Example (slide 2)

in itemize (inside block)

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).

Example (slide 3)

in itemize (inside block)

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (three divisors: 1, 2, and 4).

The <···> syntax

(slide 1)

- Typing endless `\pause` is rather tedious and does not provide the whole power of overlays
- For this purpose, Beamer introduces a new syntax of optional argument, using so called "pointed brackets" `<...>`
- The content of `<...>` is a range of slides on the which the command must be executed. E.g. `<1-3,5-8>` selects 1 to 3 and 5 to 8.
- **Most commands specific to Beamer** accept this parameter, and many commands of LaTeX are redefined to handle it, namely `\textbf`, `\textit`, `\textsl`, `\textrm`, `\textsf`, `\color`, etc. (§9.3)
- For example this frame contains in the fourth `\item`:

```
\item \textbf<4>{Most command specific to Beamer}
  \alert<5>{accept this parameter} ...
```

Another (funny) example   (slide 1)

This syntax enable some funny (but not really useful) things like:

Reverse order

1. 2 is prime (two divisors: 1 and 2).
2. 3 is prime (two divisors: 1 and 3).
3. 4 is not prime (three divisors: 1, 2, and 4).

which is coded as:

```
\begin{enumerate}
\item<3-4> 2 is prime (two divisors: 1 and 2).
\item<2-4> 3 is prime (two divisors: 1 and 3).
\item<1-4> 4 is not prime (three divisors: 1, 2, and 4).
\end{enumerate}
```

Incremental specification   (slide 1)

- Manually enter and keep up-to-date slides numbers is cumbersome.
- Fortunately, Beamer provides shortcuts:
  - The range can end with the – alone, meaning "until the end of the frame". E.g. in the previous example `<2-4>` would be written simply `<2->`.
  - In the same way, `<1-4>` could be written `<-4>` to start from the first.
  - The range can be defined as incremental by using `<+->` which means that the current slide number will be used, and incremented at the end.

Hence the previous: becomes:

```
\begin{itemize}
\item<1-> my first item,
\item<2-> my second item.
\end{itemize}
```

```
\begin{itemize}
\item<+-> my first item,
\item<+-> my second item.
\end{itemize}
```

- Incremental overlay specification is inherited, allowing the more simple:

```
\begin{itemize}[<+->]
\item my first item,
...
```

Commands controlling overlay

Beamer defines a bunch of commands intended to control overlays:

- `\only<...>{text}` Throws away `text` content on slides not in `<...>`
- `\onslide<...>{text}` Same, but when hidden `text` still takes space.
- `\visible<...>{text}` Same.
- `\uncover<...>{text}` Same, but also handle transparency.
- `\invisible<...>{text}` Opposite of `\visible`
- `\alt<...>{text1}{text2}` Alternates between `text1` and `text2` for `<...>`.
- `\temporal<...>{before}{inside}{after}` Alternate between three texts depending on slide index `before`, `inside` or after the range of `<...>`.

For the commands `\only` and `\alt` the `<...>` can also be *after* the `text`.
Then `\only` can be used to make commands `<...>`-aware (§9.3) like in:

```
\newcommand{\myblue}{\only{\color{blue}}}
\myblue<2> This text is blue only on slide 2.
```

Finally, `\only` and `\onslide` without `text` argument work as toogles.
Much more options, described in §9.4 to 9.6

Action specifications

- Inside `<...>` it is possible to add some action specifications
- Action are specified after the slide range & a `|` and followed by `@` and the target slide or range. For example one can write:

```
\item<3-|alert@4> Shown from slide 3 on, alerted on slide 4.
```

which set the `\alert` for item 3 only in slide 4.
- Actions can be defined for `\item`, `\action`, `\begin{actionenv}` and the block environments and the possible actions are by default, `alert`, `uncover`, `only`, `visible`, `invisible`, but other can be defined by the user. Se manual § 9.6.3
- Simple example using `uncover` with specified transparency:

```
\setbeamercovered{transparent=30}
\begin{itemize}[<+-|uncover@+>]
\item first
\item second
\item third
\end{itemize}
```

Example with uncover

- first
- second
- third

Example with uncover

- first
- second
- third

Example with uncover

- first
- second
- third

Summary

Animate with `multimedia`'s `\animate` (I/III)

Beamer has a companion package: `multimedia`
Multipage PDF (see `rot-sol.tex`)
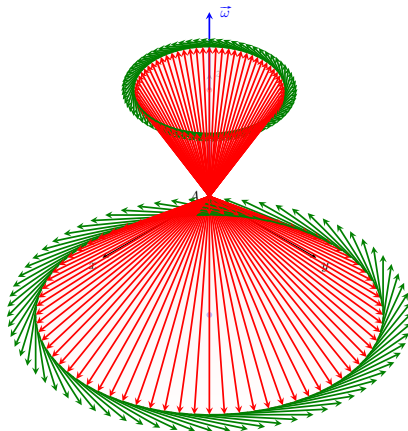Create it e.g. with `standalone` class with option `tikz` & PGF `\foreach` loop.

helpermacro.tex   To `\input` somewhere in your document preamble:

```
% helper macro for animation, to be placed in tikzpicture
\NewDocumentCommand{\ShowSlide}{sO{0.9}mm}{%
% #2: optionnal fraction of linewidth #3 :filename #4 page number
\IfBooleanTF{#1}{% starred version : show on all slides
  \node<#4->{\includegraphics[page=#4,width=#2\linewidth]{#3}};
}{% un-starred version : show only on one slide
  \node<#4-#4>{\includegraphics[page=#4,width=#2\linewidth]{#3}};
}}
```

animate1.tex   Input in a frame (& in minipage), to create a bunch of slides
`\animate` asks the viewer (Acrobat reader) to show `<...>` as fast as possible.

```
\animate<2-73>
\begin{tikzpicture}
\ShowSlide*[0.5]{rot-sol.pdf}{1} % put fixed bachground image
\foreach\ii in {2,...,73} {\ShowSlide[0.5]{rot-sol.pdf}{\ii}}
\end{tikzpicture}
```
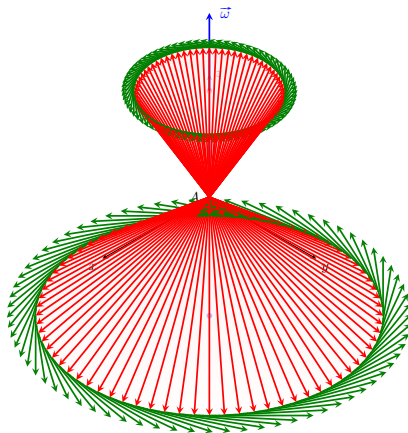
Animate with `multimedia`'s \animate (II/III)

Animation is played too fast!

Animate with `multimedia`'s `\transduration` (III/III)

Replace `\animate<2-73>` on line 4 by `\transduration<2-73>{0.2}`
(0.2 is the frame duration in seconds)

Much better!

With `animate`'s `\animategraphics`

The package `animate` is much more powerful ! Two simple uses:

1. **Multipage PDF**
   - Create it e.g. with `standalone` class with option `tikz` & PGF `\foreach` loop (see `rot-sol.tex`).
   - Put this code in a centered `minipage` (20 is the fps):
     ```
     \animategraphics[controls,loop, autoplay,%
     timeline=rot-sol-tl.txt,width=\linewidth]%
     {20}{rot-sol}{}{}
     ```
   - Warning, use the `<basename>`=`rot-sol` and not `rot-sol.pdf`, as `\animategraphics` starts by looking for `<basename>.pdf`

2. **Separated Images**, named as `<basename><start>.ext` to `<basename><end>.<ext>`, where `<ext>` is the extension (pdf, jpg, png), and `<start>` & `<end>` are the corresponding numbers (e.g. 00 & 73).

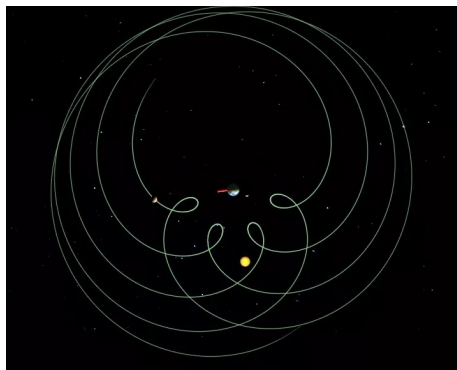   In the code above, provide them as the two last parameters:
   ```
   \animategraphics[<options>]{<fps>}{<basename>}{<start>}{<end>}
   ```

With animate's \animategraphics

Movie with `multimedia`'s `\movie`

Embedding video with `multimedia` does compile, but playing doesn't work.
The solution is to launch an external program with the option `externalviewer`.
(for me, `C:/Progra 1/VideoLAN/VLC/vlc.exe`)

```
1 \movie[externalviewer]%
2 {\includegraphics[width=0.5\linewidth]{RetrogradationMars.jpg}}%
3 {RetrogradationMars.mp4}
```

Movie with `embedvideo` package

Using `embedvideo` package like this :

```
1 \embedvideo%
2 {\includegraphics[width=0.5\linewidth]{RetrogradationMars.jpg}}%
3 {RetrogradationMars.mp4}
```