Introduction to LaTeX

Lecture V: Beamer Slides

Liu Yihao

SJTU-UMJI Technology Department

June 22, 2021

1. Introduction
   - Beamer Document
   - Beamer Structure
2. Overlay and Animation
3. Special Structures

Why `beamer`?

For LaTeX users, `beamer` has a number of advantages over PowerPoint or other presentation software: [1]

- If you are creating slides from a larger document, you can simply re-use your LaTeX source material from that document.
- If you need mathematical content in your slides, you have the wealth of mathematical constructs in LaTeX to draw upon.
- The slides you create are multi-platform.

`beamer` allows you to create slides featuring overlays, animation and so on in LaTeX. You simply insert some calls to `beamer` macros in your LaTeX source file, and compile it into a pdf file. You can then use a pdf viewer to present your slides.

[1] `http://heather.cs.ucdavis.edu/~matloff/beamer.html`

The beamer Class

In order to use beamer, you should use the following command as the first line of your tex document:

Command

```
\documentclass[options]{beamer}
```

Then you can create frames with the frame environment or the \frame command in the document body.

Example

```
\documentclass[options]{beamer}
\begin{document}
\begin{frame}
  some content
\end{frame}
\frame{some content}
\end{document}
```

The Title Page

You can add title, author, date and some other information in the preamble of the document, similar to the document class article.

Example

```
\title{Introduction to \LaTeX}
\author{Liu Yihao}
\date{\today}
\institute{SJTU-UMJI Technology Department}
```

Then you can use the \titlepage command to generate a title page.

Command

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

This is how the ▸ first page of this document is generated.

There are some more options for the title page than the ones presented. The next example is a complete one, most of the commands are optional. [1]

Example

```
\documentclass{beamer}
\usetheme{Boadilla}
\usecolortheme{seahorse}

\title[About Beamer] %optional
{About the Beamer class in presentation making}
\subtitle{A short story}
\author[Arthur, Doe] % (optional, for multiple authors)
{A.~B.~Arthur\inst{1} \and J.~Doe\inst{2}}
\institute[VFU] % (optional)
{
  \inst{1} Faculty of Physics \\ Very Famous University
  \and
  \inst{2} Faculty of Chemistry\\ Very Famous University
}
\date[VLC 2013] % (optional)
{Very Large Conference, April 2013}
\logo{\includegraphics[height=1.5cm]{example-image}}
```

[1] Some of this part is ported from the tutorial of Overleaf: ▸ Link

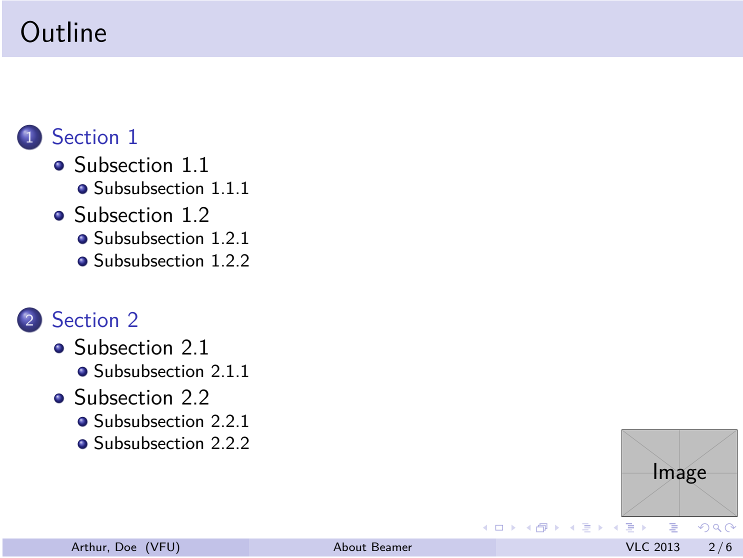
The distribution of each element in the title page depends on the theme, which will be introduced later. Here is a description of each command:

- `\title[short title]{title}` - The title of your presentation must be inside braces. You can set an optional shorter title in the square brackets.
- `\subtitle{subtitle}` - Subtitle can be omitted if unnecessary.
- `\author[short author]{author}` and `\institute[short institute]{institute}` - The usages can be referred to the example code. Use the `\inst` command to state the institute of each author if needed.
- `\date[short date]{date}` - You can also use `\today` as a date.
- `\logo{logo}` - You can use text or image, it will be shown on every slide.

The short versions of title, author, institute and date are often used in the headline or footline in the presentation. If omitted, the long versions will be used there.

The complete example of title page is demonstrated on the next page.

```
\frame{\titlepage}
```



Outline

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2
2. Section 2
   - Subsection 2.1
     - Subsubsection 2.1.1
   - Subsection 2.2
     - Subsubsection 2.2.1
     - Subsubsection 2.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 2 / 6

Frame Title

You may notice that some of the slides have a title (eg., "Frame Title" in this slide). You can use this command to add one:

Command

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

Alternatively, you can add the title as an argument of the `frame` environment:

Command

```
\begin{frame}{frame title}

\end{frame}
```

It is worth noting that in `beamer` the basic container is a `frame`. A `frame` is not exactly equivalent to a slide, one `frame` may contain more than one slides.

1. Introduction
   - Beamer Document
   - Beamer Structure
2. Overlay and Animation
3. Special Structures

Sections and Parts

You can also structure a beamer document into sections, subsections and subsubsections. Usually subsubsections are not very useful in small presentations.

Command

```latex
\section{section}
\subsection{subsection}
\subsubsection{subsubsection}
```

For large presentations or lectures (such as this one), another structure called \part can be used.

Command

```latex
\part{part}
```

The contents of different parts are often split from each other completely, eg., the counter of figures and tables, the table of contents, etc.

Table of Contents

After dividing your presentation into sections and subsections, you can add a table of contents at the beginning of the document, or before each section, or anywhere.

Command

```latex
\begin{frame}{Outline}
  \tableofcontents[options]
\end{frame}
```

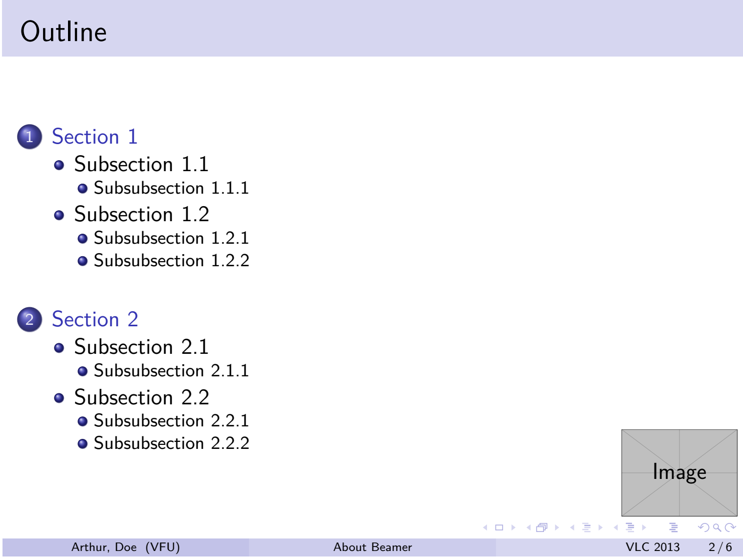
For example, if the document structure is

Example

```latex
\section{Section 1}
\subsection{Subsection 1.1}
\subsubsection{Subsubsection 1.1.1}
\subsection{Subsection 1.2}
\subsubsection{Subsubsection 1.2.1}
\subsubsection{Subsubsection 1.2.2}
\section{Section 2}
\subsection{Subsection 2.1}
\subsubsection{Subsubsection 2.1.1}
\subsection{Subsection 2.2}
\subsubsection{Subsubsection 2.2.1}
\subsubsection{Subsubsection 2.2.2}
```

An example of the default table of contents is shown on the next page.

`\frame{\tableofcontents}`



Outline

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2
2. Section 2
   - Subsection 2.1
     - Subsubsection 2.1.1
   - Subsection 2.2
     - Subsubsection 2.2.1
     - Subsubsection 2.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 2 / 6

By default, all sections, subsections and subsubsections in the current part will be shown in the table and contents. You can also use options to set whether some of the sections or subsections should be shaded, or be hided. The allowed styles are show, shaded and hide. The available options are

- `sectionstyle=<style for current section>/<style for other sections>`
- `subsectionstyle=<style for current subsection>/<style for other subsections in current section>/<style for subsections in other sections>`
- `subsubsectionstyle=<style for current subsubsection>/<style for other subsubsections in current subsection>/<style for subsubsections in other subsections in current section>/<style for subsubsections in other subsections in other sections>`

The later styles can be omitted in each options, in this case, the omitted styles will be set to the last explicit style. For example, these two lines are equivalent:

- `subsectionstyle=show/shaded`
- `subsectionstyle=show/shaded/shaded`

They both cause all subsections except the current subsection in the current section to be shown in a semi-transparent way.

There are also some shorthands of the options above, you can use them alone, or mix them with any other options.

- `currentsection` - `sectionstyle=show/shaded,subsectionstyle=show/show/shaded`
- `currentsubsection` - `subsectionstyle=show/shaded`
- `hideallsubsections` - `subsectionstyle=hide`
- `hideothersubsections` - `subsectionstyle=show/show/hide`

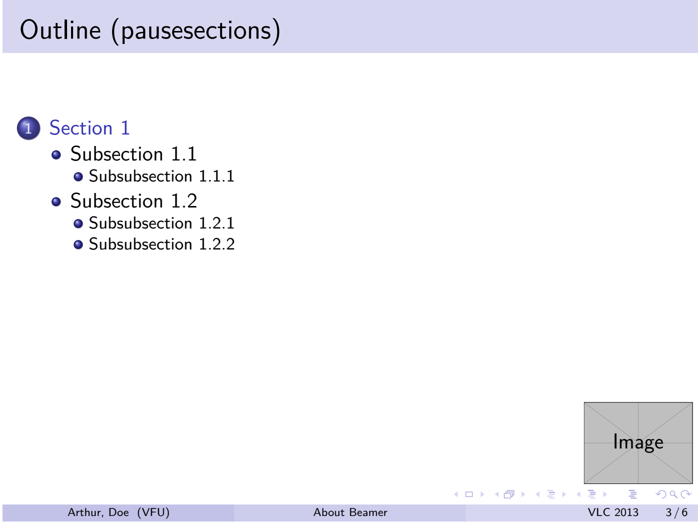
Some other options include

- `part=<part number>` - shows the table of contents of a specific part.
- `pausesections` - causes a `\pause` command to be issued before each section. This is useful if you wish to show the table of contents in an incremental way.
- `pausesubsections` - causes a `\pause` command to be issued before each subsection.

The `\pause` command can split a frame into two slides, which will be introduced in the next section.

Some examples are shown on the next pages.

`\frame{\tableofcontents[pausesection]}`

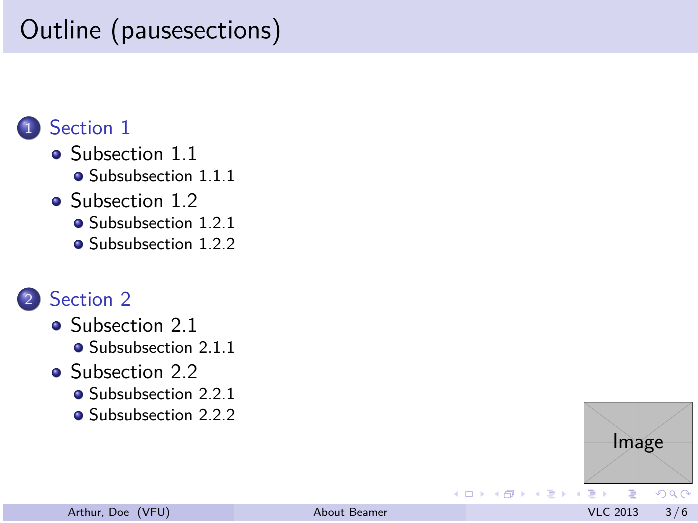


Outline (pausesections)

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 3 / 6

```
\frame{\tableofcontents[pausesection]}
```

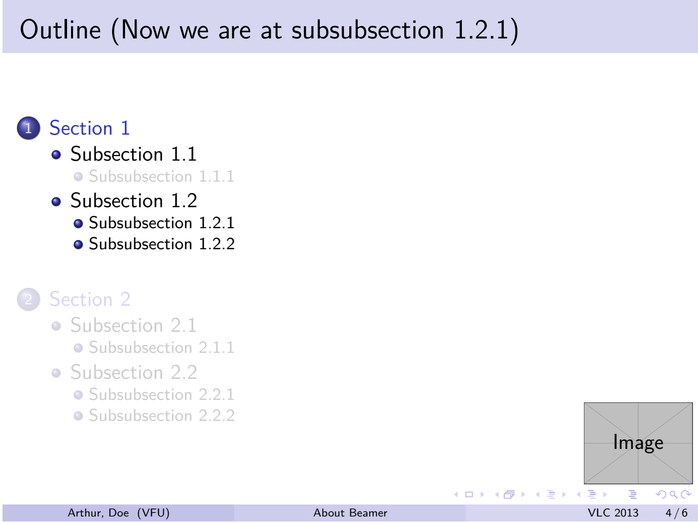


Outline (pausesections)

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2
2. Section 2
   - Subsection 2.1
     - Subsubsection 2.1.1
   - Subsection 2.2
     - Subsubsection 2.2.1
     - Subsubsection 2.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 3 / 6

```
\frame{\tableofcontents[currentsection]}
```



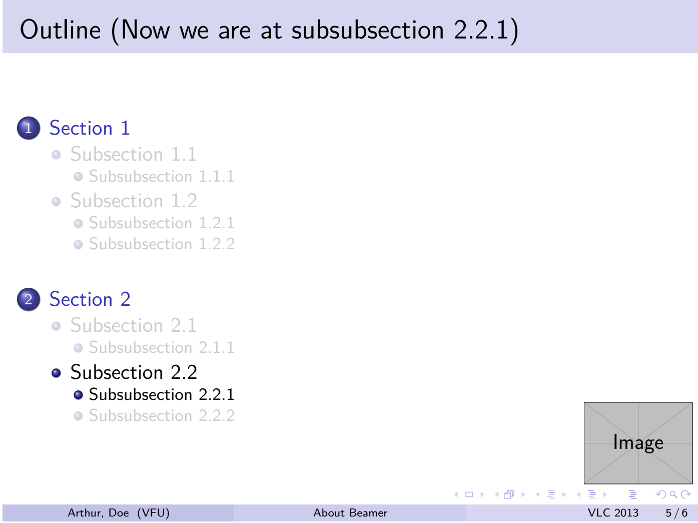
Outline (Now we are at subsubsection 1.2.1)

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2
2. Section 2
   - Subsection 2.1
     - Subsubsection 2.1.1
   - Subsection 2.2
     - Subsubsection 2.2.1
     - Subsubsection 2.2.2

Image

Arthur, Doe (VFU) About Beamer VLC 2013 4 / 6

```
\frame{\tableofcontents[currentsubsection]}
```

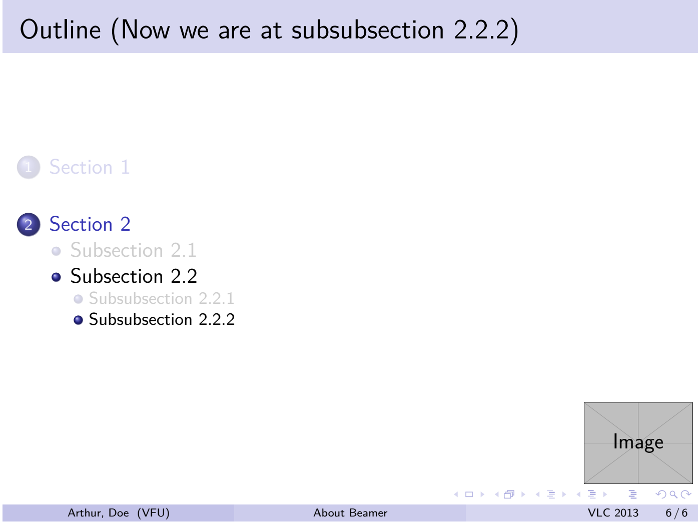


Outline (Now we are at subsection 2.2.1)

1. Section 1
   - Subsection 1.1
     - Subsubsection 1.1.1
   - Subsection 1.2
     - Subsubsection 1.2.1
     - Subsubsection 1.2.2
2. Section 2
   - Subsection 2.1
     - Subsubsection 2.1.1
   - Subsection 2.2
     - Subsubsection 2.2.1
     - Subsubsection 2.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 5 / 6

```
\frame{\tableofcontents[sectionstyle=show/shaded,
subsectionstyle=show/shaded/hide,subsubsectionstyle=show/shaded/hide]}
```



Outline (Now we are at subsubsection 2.2.2)

1 Section 1

2 Section 2
- Subsection 2.1
- Subsection 2.2
  - Subsubsection 2.2.1
  - Subsubsection 2.2.2

Arthur, Doe (VFU) About Beamer VLC 2013 6 / 6

Bibliography

Like the article class, you can use \cite to create citations and add a bibliography at the end of the file.

Unfortunately, bibtex is not perfectly supported in beamer, so usually you need to typeset them by hand.

Example

```
\begin{frame}
  \frametitle{For Further Reading}
  \begin{thebibliography}{Dijkstra, 1982}
    \bibitem[Salomaa, 1973]{Salomaa1973} A.~Salomaa.
    \newblock {\em Formal Languages}. \newblock Academic Press, 1973.
    \bibitem[Dijkstra, 1982]{Dijkstra1982} E.~Dijkstra.
    \newblock Smoothsort, an alternative for sorting in situ.
    \newblock {\em Science of Computer Programming}, 1(3):223--233, 1982.
  \end{thebibliography}
\end{frame}
```

Appendix

You can add an appendix by using the `\appendix` command. The command essentially just starts a new part named `\appendixname`. However, it also sets up certain hyperlinks.

All frames, all `\subsection` commands, and all `\section` commands used after this command will not be shown as part of the normal navigation bars.

Example

```latex
\begin{document}
\frame{\titlepage}
\section*{Outline}
\frame{\tableofcontents}
\section{Main Text}
\frame{Some text}
\section*{Summary}
\frame{Summary text}


\appendix
\section{\appendixname}
\frame{\tableofcontents}
\subsection{Additional material}
\frame{Details}
\frame{Text omitted in main talk.}
\subsection{Even more additional material}
\frame{More details}
\end{document}
```

1. Introduction
2. Overlay and Animation
   - Overlay
   - Animation
3. Special Structures

Simple Overlay

In the introduction, it was mentioned that a frame is not equivalent to a slide.

Simple Overlay

In the introduction, it was mentioned that a frame is not equivalent to a slide.

The simplest way to verify this is to add a simple overlay with the command

Command

```
\pause
```

Simple Overlay

In the introduction, it was mentioned that a frame is not equivalent to a slide.

The simplest way to verify this is to add a simple overlay with the command

Command

```latex
\pause
```

A direct example is this frame itself:

Example

```latex
\begin{frame}{Simple Overlay}
  some contents
  \pause
  some contents
  \pause
  some contents
\end{frame}
```

Note that the page numbers in the bottom right corner of these three slides are the same, the page counter only counts the number of frames.

Overlay Specifications

However, the `\pause` command only provide a basic support of splitting slides. Using commands with an overlay specification will be more flexible, which means you can have different effects with the same command on different slides.

Overlay specifications can only be written behind certain commands, not every command. `\textbf` is one of these commands.

Example

```
\textbf{This line is bold on all slides.}
\textbf<2>{This line is bold only on the second slide.}
```

This line is bold on all slides. This line is bold only on the second slide.

The syntax of (basic) overlay specifications is the following: They are comma-separated lists of slides and ranges. Ranges are specified like this: 2-5, which means slide two through to five.

The start or the end of a range can be omitted. For example, 3- means "slides three, four, five, and so on".

Overlay Specifications

However, the `\pause` command only provide a basic support of splitting slides. Using commands with an overlay specification will be more flexible, which means you can have different effects with the same command on different slides.

Overlay specifications can only be written behind certain commands, not every command. `\textbf` is one of these commands.

Example

```
\textbf{This line is bold on all slides.}
\textbf<2>{This line is bold only on the second slide.}
```

This line is bold on all slides. This line is bold only on the second slide.

The syntax of (basic) overlay specifications is the following: They are comma-separated lists of slides and ranges. Ranges are specified like this: 2-5, which means slide two through to five.

The start or the end of a range can be omitted. For example, 3- means "slides three, four, five, and so on".

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification: `\textbf`, `\textit`, `\textmd`, `\textnormal`, `\textrm`, `\textsc`, `\textsf`, `\textsl`, `\texttt`, `\textup`, `\emph`, `\color`, `\textcolor`, `\alert`, `\structure`. If a command takes several arguments, like `\color`, the specification should directly follow the command.

Example

```
\color<2>[rgb]{1,0,0} This text is red on slides 2, otherwise black.
```

This text is red on slides 2, otherwise black.

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification: `\textbf`, `\textit`, `\textmd`, `\textnormal`, `\textrm`, `\textsc`, `\textsf`, `\textsl`, `\texttt`, `\textup`, `\emph`, `\color`, `\textcolor`, `\alert`, `\structure`. If a command takes several arguments, like `\color`, the specification should directly follow the command.

Example

```
\color<2>[rgb]{1,0,0} This text is red on slides 2, otherwise black.
```

This text is red on slides 2, otherwise black.

The `onslide` Command

When you want to display or hide some contents on some slides, you can use

Command

```
\onslide(modifier)<overlay specification>{text}
```

Example

```
\onslide<1>{(1) onslide}
\uncover<1>{(1) uncover}

\onslide+<2>{(2) onslide+}
\visible<2>{(2) visible}

\onslide*<3>{(3) onslide*}
\only<3>{(3) only}
```

(1) onslide (1) uncover

The `onslide` Command

When you want to display or hide some contents on some slides, you can use

Command

```
\onslide(modifier)<overlay specification>{text}
```

Example

```
\onslide<1>{(1) onslide}
\uncover<1>{(1) uncover}

\onslide+<2>{(2) onslide+}
\visible<2>{(2) visible}

\onslide*<3>{(3) onslide*}
\only<3>{(3) only}
```

(2) onslide+ (2) visible

The onslide Command

When you want to display or hide some contents on some slides, you can use

Command

```
\onslide(modifier)<overlay specification>{text}
```

Example

```
\onslide<1>{(1) onslide}
\uncover<1>{(1) uncover}

\onslide+<2>{(2) onslide+}
\visible<2>{(2) visible}

\onslide*<3>{(3) onslide*}
\only<3>{(3) only}
```

(3) onslide* (3) only

Some explanations:

- `\onslide` is equivalent to `\uncover`, the text is only shown ("uncovered") on the specified slides. On other slides, the text still occupies space and it is still typeset, but it is not shown or only shown as if transparent (can be set with the command `\setbeamercovered`).
- `\onslide+` is equivalent to `\visible`, it does almost the same as `\uncover`, but it is never transparent, but rather it is not shown at all.
- `\onslide*` is equivalent to `\only`, the text is inserted only into the specified slides. For other slides, the text is simply thrown away. In particular, it occupies no space.

There are also some similar commands:

- `\invisible<overlay specification>{text}` is opposite to `\visible`.
- `\alt<overlay specification>{text}{alternate text}` will show the text on the specified slides and the alternate text on other slides.
- `\temporal<overlay specification>{before text}{text}{after text}` will show the text on the specified slides, the before text on the slides before the interval, and the after text on the slides after the interval.

1. Introduction
2. Overlay and Animation
   - Overlay
   - Animation
3. Special Structures

Zooming

Zooming is necessary when you want to explain a part of a frame (or a very complicated graphic).

Command

```
\framezoom<button overlay specification><zoomed overlay
specification>[options](x,y)(w,d)
```

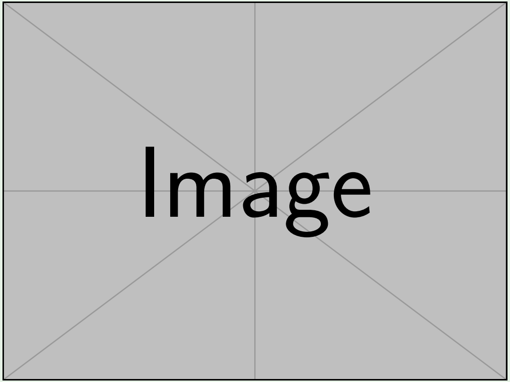
This command should be given somewhere at the beginning of a frame. The button overlay specification is your main slide with the whole graph, and there will be a clickable area which will navigate you to the zoomed slide, specified by the zoomed overlay.

`(x,y)` is the upper left corner of the clickable area. Thus, the location `(0pt,0pt)` is at the beginning of the normal text (which excludes the headline and also the frame title). `(w,d)` is the width and depth (height) of the clickable area.

You can also add `border=width` as `options` so that there will be a border around the clickable area.

Example

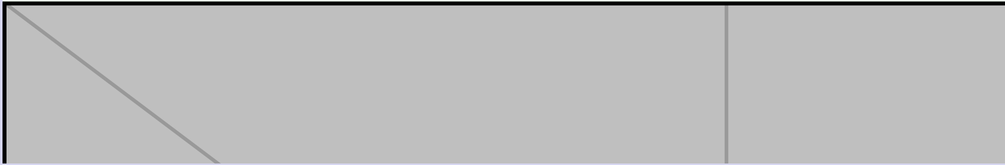
```
\framezoom<1><2>(0cm,0cm)(4cm,3cm)
\framezoom<1><3>(1.5cm,3.5cm)(4cm,3cm)
\pgfimage[height=5cm]{example-image}
```

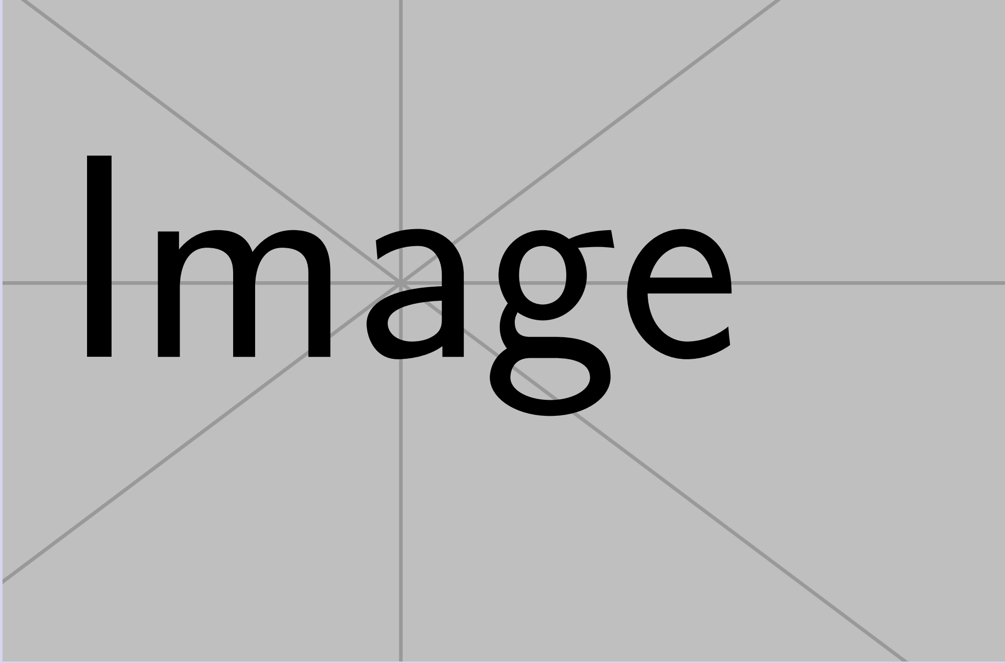


Try to click on the code and the image.

Example

```
\framezoom<1><2>(0cm,0cm)(
\framezoom<1><3>(1.5cm,3.5
\pgfimage[height=5cm]{exam
```





Image

Limitations of Zooming

Though `\framezoom` is very powerful, it still has some limitations.

- You can click on the zoomed slide to jump back to the origin slide, but it only works on Adobe Reader or Acrobat.
- The backend of XƎLATEX will ignore all hyperlinks without text in it, so when compiling with `xelatex`, the area is no longer clickable. You can use `pdflatex` or `lualatex` when you need the zooming feature.

  You can also redefine some macros in the `beamer` package to use `xelatex` with expected behavior, but it will require some knowledge of the inner concept of LATEX. You can check `lecture.sty` for more details.

The againframe Command

You can use the `\againframe` command to “continue” frames that you previously started somewhere, but where certain details have been suppressed. You need to add a label for the frame to be repeated.

Command

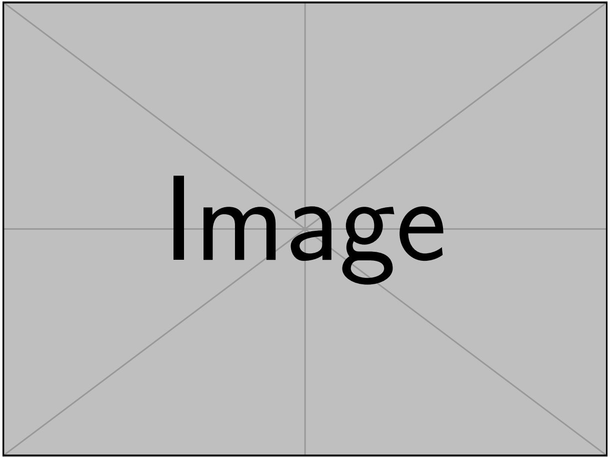
```
\againframe<overlay specification>[options]{label}
```

You can use this command together with the `\framezoom` command to put the zoomed slides at the end of the presentation.

Example

```
\begin{frame}<1>[label=zooms]
\frametitle<1>{A Complicated Picture}
\framezoom<1><2>[border](0cm,0cm)(2cm,1.5cm)
\framezoom<1><3>[border](1cm,2cm)(2cm,1.5cm)
\pgfimage[height=6cm]{example-image}
\end{frame}
% other slides
\againframe<2->[noframenumbering]{zooms}
```

A Complicated Picture



1. Introduction
2. Overlay and Animation
3. Special Structures
   - Blocks and Columns
   - Hyperlinks and Buttons
   - Fragile Frame

The block Environment

Blocks in beamer are based on the tcolorbox, with all styles configured.

Command

```
\begin{block}<action specification>{title}
  % block contents
\end{block}
```

If the action specification is present, the given actions are taken on the specified slides.

There are three types of blocks: block, alertblock and exampleblock. The only difference is their color and style.

Example

```latex
\begin{block}{Normal Block}
  This is a normal block.
\end{block}
\begin{alertblock}{Alert Block}
  This is an alert block.
\end{alertblock}
\begin{exampleblock}{Example Block}
  This is an example block.
\end{exampleblock}
```

Normal Block

This is a normal block.

Alert Block

This is an alert block.

Example Block

This is an example block.

Predefined block environments

Some other block environment are predefined for ease to use. They are `theorem`, `corollary`, `definition`, `definitions`, `example`, `examples`. Here only a few of them are shown below.

Example

```
\begin{theorem}[additional text]
  The additional text will be in brackets if the option is provided.
\end{theorem}
\begin{proof}[Proof Name]
  The default title is ``Proof.'', which can be replaced by the option.
\end{proof}
```

Theorem (additional text)

The additional text will be in brackets if the option is provided.

Proof Name.

The default title is “Proof.”, which can be replaced by the option. □

The columns Environment

In an article class, minipage is often used to split contents into multiple columns; In beamer, you can use the columns environment as an alternative.

Command

```
\begin{columns}[options]
  \begin{column}[placement]{width}
    % contents
  \end{column}
  \column[placement]{width}{...}
\end{columns}
```

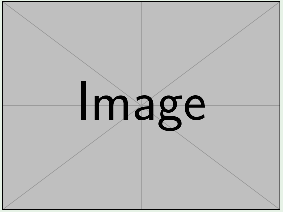
For the options in the columns environment,

- b, c and t - will cause the columns to be vertically aligned bottom, center and top.
- T - similar to t, if strange things happen in t, try this option.
- totalwidth=width - will cause the columns to occupy not the whole page width, but only width.

Example

```latex
\begin{columns}[c]
  \begin{column}{0.5\textwidth}
    \begin{center}
      The first line \\
      The second line
    \end{center}
  \end{column}
  \column{0.5\textwidth}{
    \includegraphics[width=0.6\textwidth]{example-image}
  }
\end{columns}
```

The first line
The second line



You should place only `column` environments or `\column` commands in the `columns` environment.

For the `placement` in the `column` environment or the `\column` command, you can overwrite the `b`, `c`, `t` and `T` in the outer `columns` environment. The default of `placement` is `t` if not specified.

The `width` is the same as other `width` in LaTeX. For example, you can use `5cm`, or `\textwidth`.

Generally speaking, there are few differences between `minipage` and `columns`, but the `columns` environment has a more user-friendly structure, and overlap is supported better in it. Despite which do you prefer, choosing one of them and sticking to it throughout a presentation is suggested.

Footnote in column

Using footnotes is usually not a good idea. They disrupt the flow of reading.

When you really need it, you can use the \footnote command, which is slightly different from common LaTeX.

Command

```
\footnote<overlay specification>[options]{text}
```

As usual, you can give a number as options, which will cause the footnote to use that number.

You can also add a frame as options so that the footnote will be shown at the bottom of the frame. This is normally the default behavior anyway, but in minipage, columns and certain blocks it makes a difference.

In a minipage or column, the footnote is usually shown as part of the minipage rather than as part of the frame.

Example

```latex
\begin{columns}[c,totalwidth=0.9\textwidth]
  \begin{column}{0.3\textwidth}
    The first line \footnote[frame,1]{footnote 1}
  \end{column}
  \begin{column}{0.3\textwidth}
    The second line \footnote[frame,2]{footnote 2}
  \end{column}
  \begin{column}{0.3\textwidth}
    The third line \footnote[3]{footnote 3}
  \end{column}
\end{columns}
```

The first line [1]

The second line [2]

The third line [c]

[c] footnote 3

As you can see, placing footnote at the bottom of a `column` is not a good idea, so using the `frame` option is preferred in most situations.

[1] footnote 1

[2] footnote 2

1. Introduction
2. Overlay and Animation
3. Special Structures
   - Blocks and Columns
   - Hyperlinks and Buttons
   - Fragile Frame

Hyperlinks and Buttons

Here is a simple three-step workflow of how to create hyperlinks and buttons in your slides:

1. You specify a target using the command `\hypertarget` or (easier) the command `\label`. In some cases, see below, this step may be skipped.
2. You render the button using `\beamerbutton` or a similar command. This will render the button, but clicking it will not yet have any effect.
3. You put the button inside a `\hyperlink` command. Now clicking it will jump to the target of the link.

The beamerbutton command

Command

```
\beamerbutton{text}
```

There are four types of buttons, the suggested usages of them are:

- `\beamerbutton` - a normal button
- `\beamergotobutton` - jump to another area of the presentation
- `\beamerskipbutton` - skip a well-defined part
- `\beamerreturnbutton` - return back to a previous part

Example

```
\beamerbutton{normal}   \beamergotobutton{goto}
\beamerskipbutton{skip} \beamerreturnbutton{return}
```

normal ▸ goto ▸▸ skip ◂ return

The color of the buttons will follow the current color of the block.

normal ▸ goto ▸▸ skip ◂ return

The hyperlink command

Command

```
\hyperlink<overlay specification>{target}{text}
```

If the overlay specification is present, the hyperlink (including the text) is completely suppressed on the non-specified slides.

You will jump to the target you defined before with a \hypertarget command.

You can also use a LaTeX command from the hyperref package as a target, e.g., \href.

Example

```
You can find the source of the slides on
\href{https://github.com/SJTU-UMJI-Tech/LaTeX}{\beamerbutton{GitHub}}.
```

You can find the source of the slides on GitHub.

The hypertarget command

Command

```
\hypertarget<overlay specification>{target}{text}
```

If the overlay specification is present, the text is the target for hyper jumps only on the specified slide. On all other slides, the text is shown normally.

Example

```
\begin{itemize}
  \item<1-> First item.
  \item<2-> Second item.
\end{itemize}
\hyperlink{jumptosecond}{\beamergotobutton{Jump to second slide}}
\hypertarget<2>{jumptosecond}{}
```

- First item.

▸ Jump to second slide

The `hypertarget` command

Command

`\hypertarget<overlay specification>{target}{text}`

If the overlay specification is present, the text is the target for hyper jumps only on the specified slide. On all other slides, the text is shown normally.

Example

```
\begin{itemize}
  \item<1-> First item.
  \item<2-> Second item.
\end{itemize}
\hyperlink{jumptosecond}{\beamergotobutton{Jump to second slide}}
\hypertarget<2>{jumptosecond}{}
```

- First item.
- Second item.

▸ Jump to second slide

1. Introduction
2. Overlay and Animation
3. Special Structures
   - Blocks and Columns
   - Hyperlinks and Buttons
   - Fragile Frame

Fragile Frame

When you need to insert any verbatim (such as `minted`) in a frame, you have to add the option `[fragile]`, and the `\end{frame}` must be alone on a single line (except for any leading whitespace).

Example

```latex
\begin{frame}[fragile]
  \begin{minted}{latex}
\documentclass{beamer}
  \end{minted}
\end{frame}
```

Using this option will cause the frame contents to be written to an external file and then read back.

Frame in Fragile Frame

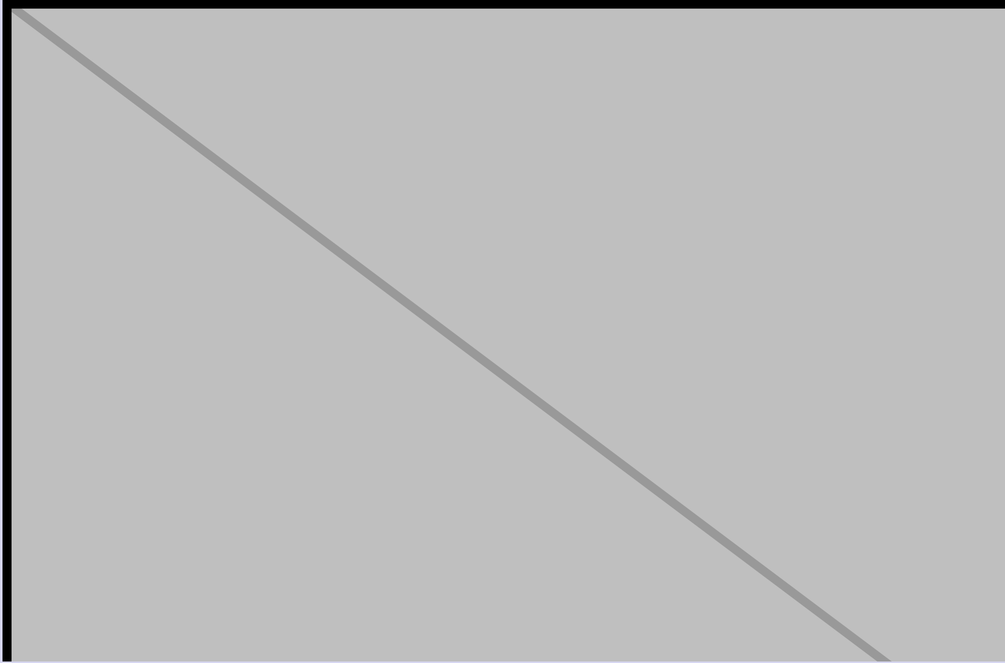
A more tricky situation is that you want to put `\begin{frame}` and `\end{frame}` inside a verbatim environment (like this lecture).

One possible solution is to define a new environment so that the new environment name won't conflict with the content in the verbatim environment.

Example

```
\newenvironment{fragileframe}%
{\begin{frame}[fragile,environment=fragileframe]}%
{\end{frame}}
```

Then you can use `\begin{fragileframe}` to substitute `\begin{frame}[fragile]` for all fragile frames.



lm