

# Curso avanzado de BEAMER

## Universitat Jaume I, Castelló

Facundo Muñoz



Grup d'Estadística espacial i temporal  
en Epidemiologia i Medi Ambient



Departament d'Estadística  
i Investigació Operativa

Febrero, 2011





- 1 BEAMER básico
- 2 Control avanzado de overlays
- 3 Handouts y modos
- 4 Botones y enlaces
- 5 Efectos y animaciones
- 6 Temas y personalización
- 7 Taller de fin de curso





# Sección 1 | BEAMER básico





# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- **Estilos** predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de **overlays** y efectos dinámicos
- Software **libre** y gratuito, con una amplísima comunidad de soporte



# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- Estilos predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de **overlays** y efectos dinámicos
- Software **libre** y gratuito, con una amplísima comunidad de soporte



# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- Estilos predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de overlays y efectos dinámicos
- Software libre y gratuito, con una amplísima comunidad de soporte



# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- **Estilos** predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de **overlays** y efectos dinámicos
- Software **libre** y gratuito, con una amplísima comunidad de soporte



# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- **Estilos** predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de **overlays** y efectos dinámicos
- Software **libre** y gratuito, con una amplísima comunidad de soporte



# Presentaciones con BEAMER

Qué es BEAMER

## BEAMER

BEAMER es una **clase** de L<sup>A</sup>T<sub>E</sub>X para la creación de presentaciones.

- Todas las ventajas heredadas de L<sup>A</sup>T<sub>E</sub>X  
separación de contenido y estilo, programable, estándar, ligero, excelente calidad tipográfica, gestión automática de referencias, etc.
- Presentación en PDF  
estándar, portable, etc.
- **Estilos** predefinidos elegantes y con herramientas útiles  
cabeceras y pies de página informativos, botones de navegación, tablas de contenidos, etc.
- Fácil generación de **overlays** y efectos dinámicos
- Software **libre** y gratuito, con una amplísima comunidad de soporte



# Instalación

- Depende de la **plataforma** (  ) , y de la **distribución** de L<sup>A</sup>T<sub>E</sub>X.
- Instalar según el procedimiento habitual para cualquier paquete de L<sup>A</sup>T<sub>E</sub>X.
- Última versión (estable)  
<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/>
- Versión en desarrollo  
<https://bitbucket.org/rivanvx/beamer/wiki/Home>





# Instalación

- Depende de la **plataforma** (  ) , y de la **distribución** de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .
- Instalar según el procedimiento habitual para cualquier paquete de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .
- Última versión (estable)  
<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/>
- Versión en desarrollo  
<https://bitbucket.org/rivanvx/beamer/wiki/Home>





# Instalación

- Depende de la **plataforma** (  ) , y de la **distribución** de L<sup>A</sup>T<sub>E</sub>X.
- Instalar según el procedimiento habitual para cualquier paquete de L<sup>A</sup>T<sub>E</sub>X.
- Última versión (estable)  
<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/>
- Versión en desarrollo  
<https://bitbucket.org/rivanvx/beamer/wiki/Home>



# Instalación

- Depende de la **plataforma** (  ) , y de la **distribución** de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .
- Instalar según el procedimiento habitual para cualquier paquete de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .
- Última versión (estable)  
<http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/>
- Versión en desarrollo  
<https://bitbucket.org/rivanvx/beamer/wiki/Home>



# Estructura básica

## Ejemplo 1

```

\documentclass{beamer}

\title{Ejemplo b\'asico}
\author{Facundo Mu~noz}
\date{\today}

\begin{document}

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

\begin{frame}
  \frametitle{Primera transp.}
  \framesubtitle{subt\'itulo}
  Contenido primero
\end{frame}

\end{document}

```

The screenshot shows two slides from a Beamer presentation. The top slide is the title page, displaying the title "Ejemplo básico", the author "Facundo Muñoz", and the date "February 8, 2011". The bottom slide is a transition slide, displaying the title "Primera transp." and the subtitle "subtitulo", with the main content "Contenido primero". Both slides include navigation icons at the bottom right.



## pdflatex prese.tex

- pdflatex entiende **formatos de imágenes**  
.tif, .gif, .jpg, .png, .pdf
- Se puede utilizar **.eps** con `\usepackage{epstopdf}`
- Compilar (al menos) **2 veces** (referencias, tabla contenidos, etc.)
- O usar rubber, o un Makefile (**Linux**)





```
pdflatex prese.tex
```

- pdflatex entiende **formatos de imágenes**  
.tif, .gif, .jpg, .png, .pdf
- Se puede utilizar **.eps** con `\usepackage{epstopdf}`
- Compilar (al menos) **2 veces** (referencias, tabla contenidos, etc.)
- O usar rubber, o un Makefile (**Linux**)





```
pdflatex prese.tex
```

- pdflatex entiende **formatos de imágenes**  
.tif, .gif, .jpg, .png, .pdf
- Se puede utilizar **.eps** con `\usepackage{epstopdf}`
- Compilar (al menos) **2 veces** (referencias, tabla contenidos, etc.)
- O usar rubber, o un Makefile (**Linux**)





```
pdflatex prese.tex
```

- pdflatex entiende **formatos de imágenes**  
.tif, .gif, .jpg, .png, .pdf
- Se puede utilizar **.eps** con `\usepackage{epstopdf}`
- Compilar (al menos) **2 veces** (referencias, tabla contenidos, etc.)
- O usar rubber, o un Makefile (**Linux**)



```
pdflatex prese.tex
```

- pdflatex entiende **formatos de imágenes**  
.tif, .gif, .jpg, .png, .pdf
- Se puede utilizar **.eps** con `\usepackage{epstopdf}`
- Compilar (al menos) **2 veces** (referencias, tabla contenidos, etc.)
- O usar rubber, o un Makefile (**Linux**)



- Una forma práctica de empezar con BEAMER es utilizar un **template**.
- El paquete proporciona tres templates comentados y con ejemplos:

.../beamer/solutions/<sup>1</sup>

...short-talks/

Charla de 2 min. para presentar a otro conferenciante

...generic-talks/

Charla de entre 15 y 45 min.

...conference-talks/

Charla en una conferencia o coloquio de unos 20 min.

- Basta copiar un template y modificar el contenido.



---

<sup>1</sup>En mi ordenador: /usr/share/doc/latex-beamer/solutions/



- Una forma práctica de empezar con BEAMER es utilizar un **template**.
- El paquete proporciona tres templates comentados y con ejemplos:

`.../beamer/solutions/1`

`...short-talks/`

Charla de 2 min. para presentar a otro conferenciante

`...generic-talks/`

Charla de entre 15 y 45 min.

`...conference-talks/`

Charla en una conferencia o coloquio de unos 20 min.

- Basta copiar un template y modificar el contenido.

---

<sup>1</sup>En mi ordenador: `/usr/share/doc/latex-beamer/solutions/`



- Una forma práctica de empezar con BEAMER es utilizar un **template**.
- El paquete proporciona tres templates comentados y con ejemplos:

.../beamer/solutions/<sup>1</sup>

...short-talks/

Charla de 2 min. para presentar a otro conferenciante

...generic-talks/

Charla de entre 15 y 45 min.

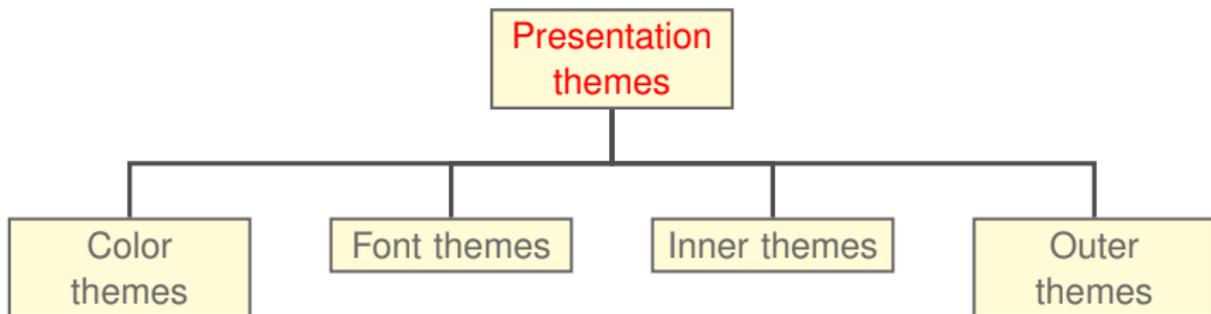
...conference-talks/

Charla en una conferencia o coloquio de unos 20 min.

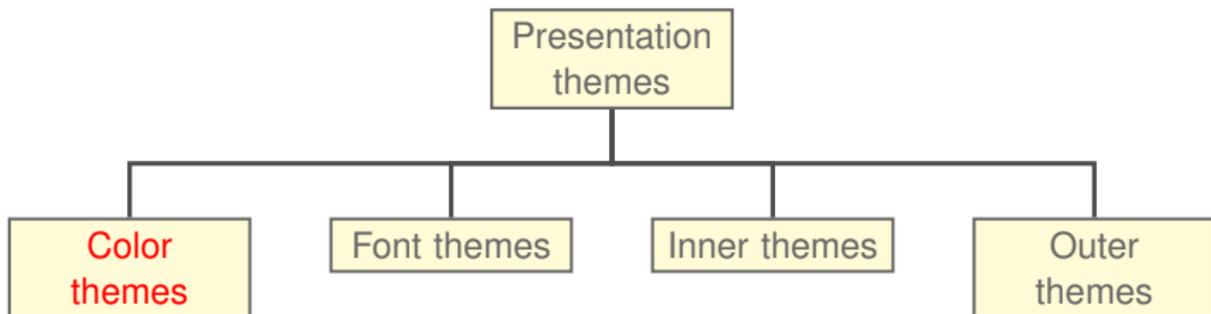
- Basta copiar un template y modificar el contenido.

---

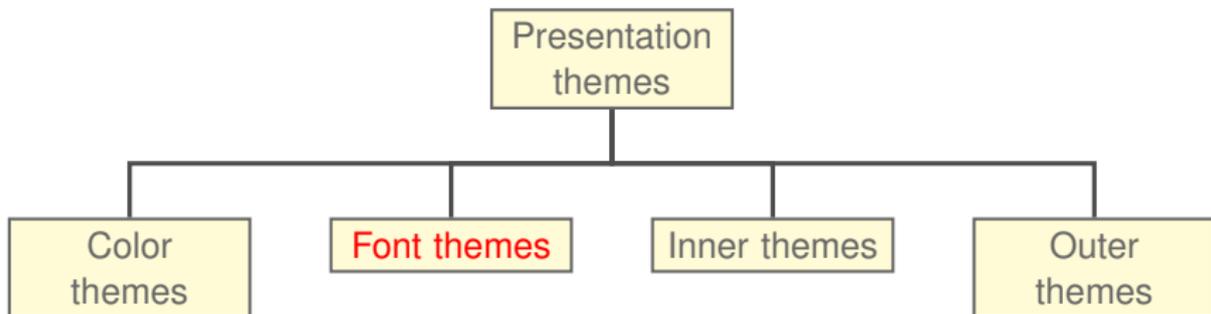
<sup>1</sup>En mi ordenador: /usr/share/doc/latex-beamer/solutions/



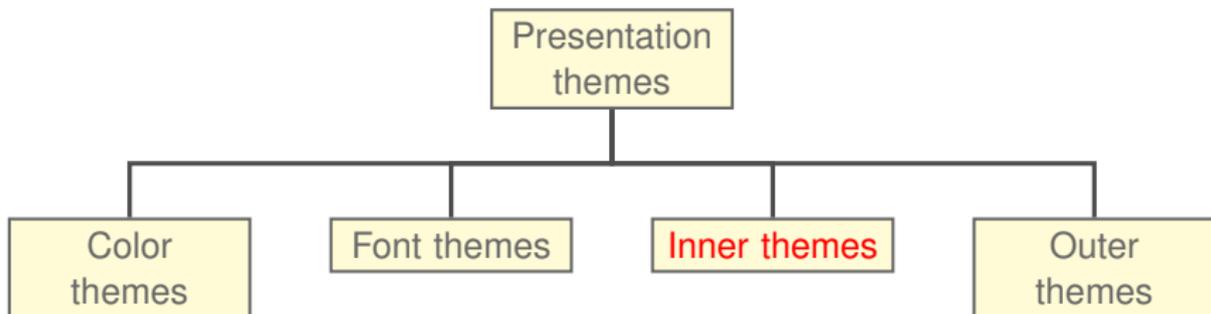
- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



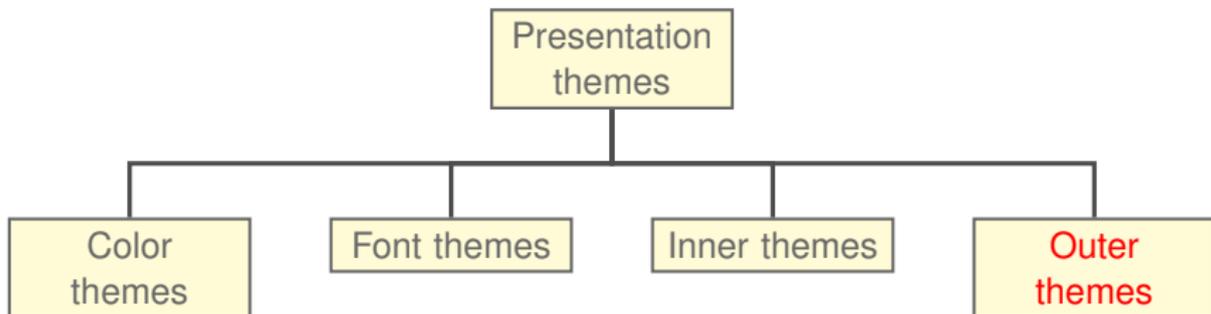
- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



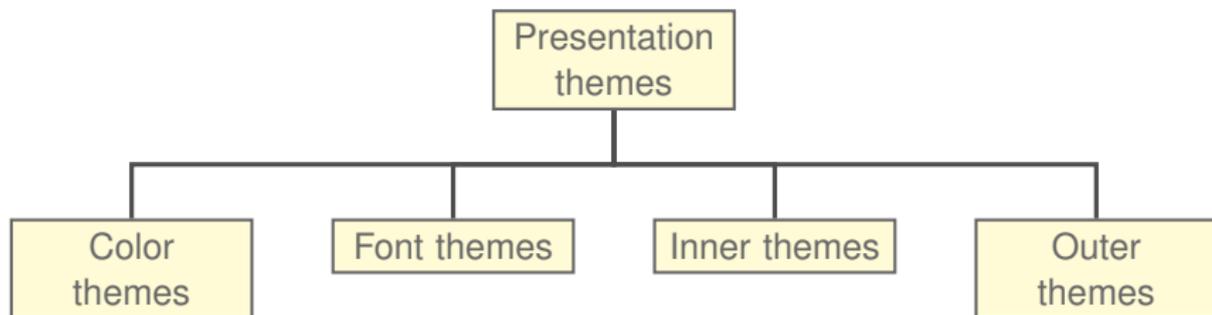
- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



- BEAMER proporciona casi 30 estilos (themes) predefinidos que especifican:
  - Los colores utilizados en cada uno de los elementos de la presentación
  - El estilo de letra
  - El aspecto de las listas y bloques
  - El aspecto del cabezal, pie de página, barras de navegación
- Preámbulo: `\usetheme{nombre tema}`



# Presentation themes

## ■ Sin barra de navegación

default, boxes, Boadilla, **Madrid**, AnnArbor, CambridgeUS, Pittsburg, Rochester.

The image shows two Beamer slide thumbnails side-by-side. The left thumbnail displays a slide with a blue header bar containing the text "Beamer Theme Matrix". Below the header, the text "Sebastian@Pipping.org" is centered, followed by "Institute" and "November 15, 2010". The right thumbnail displays a slide with a blue header bar containing the text "Catchy Slide Title". Below the header, the text "Nested stuff for demonstration purposes:" is followed by a bulleted list of nested items: "Hey one two" (with sub-items "One.. one.." and "One two"), "Like Gentoo" (with sub-items "Come on", "please", and "say now:"), and "One two". Both thumbnails have a footer bar with navigation icons and the text "Sebastian@Pipping.org (Institute) Short Title November 15, 2010 1 / 9" and "Sebastian@Pipping.org (Institute) Short Title November 15, 2010 2 / 9" respectively.

- Con árbol de navegación
- Con tabla de contenidos lateral
- Con esquema de navegación
- Con títulos de sección y subsección



# Presentation themes

- Sin barra de navegación
- **Con árbol de navegación**  
Antibes, JuanLesPins, **Montpellier**.

<p>Short Title</p> <hr/> <p style="text-align: center;">Beamer Theme Matrix</p> <p style="text-align: center;">Sebastian@Pipping.org</p> <p style="text-align: center;">Institute</p> <p style="text-align: center;">November 15, 2010</p> <p style="text-align: right;">◀ ▶ ↻ 🔍</p>	<p>Short Title</p> <ul style="list-style-type: none"> <li>└ Section 1             <ul style="list-style-type: none"> <li>└ Subsection 1.1</li> </ul> </li> </ul> <p>Catchy Slide Title</p> <p>Nested stuff for demonstration purposes:</p> <ul style="list-style-type: none"> <li>▶ Hey one two             <ul style="list-style-type: none"> <li>▶ One.. one..                 <ul style="list-style-type: none"> <li>▶ One two</li> </ul> </li> <li>▶ Say say                 <ul style="list-style-type: none"> <li>▶ say one two</li> </ul> </li> </ul> </li> <li>▶ One two</li> <li>▶ Like Gentoo             <ol style="list-style-type: none"> <li>1. Come on</li> <li>2. please</li> <li>3. say now:</li> </ol> </li> <li>▶ One two</li> </ul> <p style="text-align: right;">◀ ▶ ↻ 🔍</p>
--	---

- Con tabla de contenidos lateral
- Con esquema de navegación
- Con títulos de sección y subsección



# Presentation themes

- Sin barra de navegación
- Con árbol de navegación
- **Con tabla de contenidos lateral**  
Berkeley, PaloAlto; Goettingen, **Marburg**, Hannover.

Beamer Theme Matrix

Sebastian@Pipping.org

Institute

November 15, 2010

Short Title  
Sebastian@Pipping.org

Section 1  
Subsection 1.1  
Subsection 1.2  
Subsection 1.3

Section 2  
Subsection 2.1

Catchy Slide Title

Nested stuff for demonstration purposes:

- ▶ Hey one two
  - ▶ One.. one..
    - ▶ One two
  - ▶ Say say
    - ▶ say one two
- ▶ One two
- ▶ Like Gentoo
  1. Come on
  2. please
  3. say now:
- ▶ One two

Short Title  
Sebastian@Pipping.org

Section 1  
Subsection 1.1  
Subsection 1.2  
Subsection 1.3

Section 2  
Subsection 2.1

- Con esquema de navegación
- Con títulos de sección y subsección



# Presentation themes

- Sin barra de navegación
- Con árbol de navegación
- Con tabla de contenidos lateral
- **Con esquema de navegación**

Berlin, Ilmenau, Dresden, Darmstadt, **Frankfurt**, Singapore, Szeged.

The image displays two Beamer presentation slides side-by-side. The left slide is a title slide with a blue header bar containing the text "Beamer Theme Matrix". Below the header, the text "Sebastian@Pipping.org", "Institute", and "November 15, 2010" is centered. The right slide is a content slide with a blue header bar containing the text "Catchy Slide Title". Below the header, the text "Nested stuff for demonstration purposes:" is followed by a bulleted list of nested items: "Hey one two" (with sub-items "One.. one.." and "One two"), "Say say" (with sub-item "say one two"), "One two", "Like Gentoo" (with sub-items "Come on", "please", and "say now:"), and "One two".

- Con títulos de sección y subsección



# Presentation themes

- Sin barra de navegación
- Con árbol de navegación
- Con tabla de contenidos lateral
- Con esquema de navegación
- **Con títulos de sección y subsección**  
Copenhagen, Luebeck, **Malmö**, Warsaw.

Section 1  
Section 2

Beamer Theme Matrix

Sebastian@Pipping.org

Institute

November 15, 2010

Sebastian@Pipping.org Short Title

Section 1  
Section 2

Subsection 2.1  
Subsection 2.2  
Subsection 2.3

Catchy Slide Title

Nested stuff for demonstration purposes:

- ▶ Hey one two
  - ▶ One.. one..
    - ▶ One two
  - ▶ Say say
    - ▶ say one two
- ▶ One two
- ▶ Like Gentoo
  1. Come on
  2. please
  3. say now:
- ▶ One two

Sebastian@Pipping.org Short Title



Una vez decidido el tipo de **estructura** que queremos para la presentación, podemos afinar el **estilo** y los **colores** con las siguientes **galerías**.

- BEAMER theme matrix

<http://www.hartwork.org/beamer-theme-matrix/>

- BEAMER gallery

[http://deic.uab.es/~iblanes/beamer\\_gallery/](http://deic.uab.es/~iblanes/beamer_gallery/)





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - Inserta la correspondiente entrada en la Tabla de Contenidos
  - Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - 1 Inserta la correspondiente entrada en la Tabla de Contenidos
  - 2 Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - 1 Inserta la correspondiente entrada en la Tabla de Contenidos
  - 2 Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - 1 Inserta la correspondiente entrada en la Tabla de Contenidos
  - 2 Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - 1 Inserta la correspondiente entrada en la Tabla de Contenidos
  - 2 Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Organización del contenido

## Estructura global

- Las presentaciones se dividen en **Secciones**, **Subsecciones** y **Sub-subsecciones**
- Cada comando `\section{nombre}`, `\subsection{nombre}` o `\subsubsection{nombre}`, de forma automática:
  - 1 Inserta la correspondiente entrada en la Tabla de Contenidos
  - 2 Inserta la correspondiente entrada en las barras de navegación
- La especificación de secciones se hace **fuera de los frames**
- No producen ningún contenido





# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:

- `\title[abrev.]{título}`
- `\subtitle[abrev.]{subtítulo}`
- `\author[abrev.]{autor(es)}`  
`\author[Benaspandra et al.]{L. Benaspandra\inst{1} \and T. Tartau\inst{2}}`
- `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]`  
`\inst{1}Department of Informatics\University of Rijeka \and`  
`\inst{2}Fakultät für Elektrotechnik und Informatik\TU Berlin`
- `\date[abrev.]{fecha}`
- `\titlegraphic[abrev.]{figura}`

- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:
  - `\title[abrev.]{título}`
  - `\subtitle[abrev.]{subtítulo}`
  - `\author[abrev.]{autor(es)}`  
`\author[Benaspandra et al.]{L. Benaspandra\inst{1} \and T. Tartau\inst{2}}`
  - `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]`  
`\inst{1}Department of Informatics\University of Rijeka` and  
`\inst{2}Fakultät für Elektrotechnik und Informatik`
  - `\date[abrev.]{fecha}`
  - `\titlegraphic[abrev.]{figura}`
- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:
  - `\title[abrev.]{título}`
  - `\subtitle[abrev.]{subtítulo}`
  - `\author[abrev.]{autor(es)}`  
`\author[Nemaspaandra et al.]{L. Nemaspaandra\inst{1} \and T. Tartau\inst{2}}`
  - `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]`  
`\inst{1}Department of Informatics\University of Rijeka` and  
`\inst{2}Fakultät für Elektrotechnik und Informatik`
  - `\date[abrev.]{fecha}`
  - `\titlegraphic[abrev.]{figura}`
- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:

- `\title[abrev.]{título}`
- `\subtitle[abrev.]{subtítulo}`
- `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
- `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]`  
`\inst{1}Department of Informatics\University of Rijeka` and  
`\inst{2}Fakultät für Elektrotechnik und Informatik`
- `\date[abrev.]{fecha}`
- `\titlegraphic[abrev.]{figura}`

- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:

- `\title[abrev.]{título}`
- `\subtitle[abrev.]{subtítulo}`
- `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
- `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]{`  
`\inst{1}Department of Informatics\\University of Rijeka \and`  
`\inst{2}Fakultät für Elektrotechnik und Informatik\\TUB}`
- `\date[abrev.]{fecha}`
- `\titlegraphic[abrev.]{figura}`

- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:

- `\title[abrev.]{título}`
- `\subtitle[abrev.]{subtítulo}`
- `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
- `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]{\inst{1}Department of Informatics\University of Rijeka \and \inst{2}Fakultät für Elektrotechnik und Informatik\TUE}`
- `\date[abrev.]{fecha}`
- `\titlegraphic[abrev.]{figura}`

- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:

- `\title[abrev.]{título}`
- `\subtitle[abrev.]{subtítulo}`
- `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
- `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]{\inst{1}Department of Informatics\University of Rijeka \and \inst{2}Fakultät für Elektrotechnik und Informatik\TUE}`
- `\date[abrev.]{fecha}`
- `\titlegraphic[abrev.]{figura}`

- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:
  - `\title[abrev.]{título}`
  - `\subtitle[abrev.]{subtítulo}`
  - `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
  - `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]{\inst{1}Department of Informatics\University of Rijeka \and \inst{2}Fakultät für Elektrotechnik und Informatik\TUE}`
  - `\date[abrev.]{fecha}`
  - `\titlegraphic[abrev.]{figura}`
- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Página de título

- El comando `\titlepage`<sup>2</sup> genera una portada que incluye la siguiente información especificada en el preámbulo:
  - `\title[abrev.]{título}`
  - `\subtitle[abrev.]{subtítulo}`
  - `\author[abrev.]{autor(es)}`  
`\author[Hemaspaandra et al.]{L. Hemaspaandra\inst{1} \and T. Tantau\inst{2}}`
  - `\institute[abrev.]{afiliación(es)}`  
`\institute[Universities of Rijeka and Berlin]{\inst{1}Department of Informatics\University of Rijeka \and \inst{2}Fakultät für Elektrotechnik und Informatik\TUE}`
  - `\date[abrev.]{fecha}`
  - `\titlegraphic[abrev.]{figura}`
- Las versiones abreviadas se utilizan para las barras de navegación y los pies de página
- La información omitida en el preámbulo no aparece en la portada

---

<sup>2</sup>utilizado dentro de un entorno `frame`



# Tabla de contenidos

- Se genera automáticamente con el comando `\tableofcontents`<sup>3</sup>, basada en las secciones y subsecciones definidas en el documento
- Algunas opciones interesantes:
  - `\tableofcontents[pausesections]`  
pausa las subsecciones
  - `\tableofcontents[hideallsubsections]`  
oculta las subsecciones

---

<sup>3</sup>utilizado dentro de un entorno `frame`



# Tabla de contenidos

- Se genera automáticamente con el comando `\tableofcontents`<sup>3</sup>, basada en las secciones y subsecciones definidas en el documento
- Algunas opciones interesantes:
  - `\tableofcontents[pausesections]`  
enseña las secciones paso a paso
  - `\tableofcontents[hideallsubsections]`  
oculta las subsecciones

---

<sup>3</sup>utilizado dentro de un entorno `frame`



# Tabla de contenidos

- Se genera automáticamente con el comando `\tableofcontents`<sup>3</sup>, basada en las secciones y subsecciones definidas en el documento
- Algunas opciones interesantes:
  - `\tableofcontents[pausesections]`  
enseña las secciones paso a paso
  - `\tableofcontents[hideallsubsections]`  
oculta las subsecciones

---

<sup>3</sup>utilizado dentro de un entorno `frame`



# Tabla de contenidos

- Se genera automáticamente con el comando `\tableofcontents`<sup>3</sup>, basada en las secciones y subsecciones definidas en el documento
- Algunas opciones interesantes:
  - `\tableofcontents[pausesections]`  
enseña las secciones paso a paso
  - `\tableofcontents[hideallsubsections]`  
oculta las subsecciones

---

<sup>3</sup>utilizado dentro de un entorno `frame`



- Se puede incluir un comando `\logo` en el preámbulo, que típicamente se utiliza para especificar una imagen (logo) de la Universidad o institución de afiliación
- Pero en general puede ser cualquier contenido.
- Cómo se procesa esta imagen depende del **Theme**, pero usualmente se presenta el logo en todas las diapositivas, en alguna esquina.





- Se puede incluir un comando `\logo` en el preámbulo, que típicamente se utiliza para especificar una imagen (logo) de la Universidad o institución de afiliación
- Pero en general puede ser cualquier contenido
- Cómo se procesa esta imagen depende del **Theme**, pero usualmente se presenta el logo en todas las diapositivas, en alguna esquina.





- Se puede incluir un comando `\logo` en el preámbulo, que típicamente se utiliza para especificar una imagen (logo) de la Universidad o institución de afiliación
- Pero en general puede ser cualquier contenido
- Cómo se procesa esta imagen depende del **Theme**, pero usualmente se presenta el logo en todas las diapositivas, en alguna esquina.





- Se puede incluir un comando `\logo` en el preámbulo, que típicamente se utiliza para especificar una imagen (logo) de la Universidad o institución de afiliación
- Pero en general puede ser cualquier contenido
- Cómo se procesa esta imagen depende del **Theme**, pero usualmente se presenta el logo en todas las diapositivas, en alguna esquina.

```
\logo{\includegraphics{logoUJI_top}}
```



Además de texto organizado en párrafos, el contenido se suele presentar en forma de lista utilizando los entornos de L<sup>A</sup>T<sub>E</sub>X `itemize`, `enumerate` o `description`

```
\begin{itemize}
  \item El primer punto a destacar
  \item el segundo...
  \item ... y así sucesivamente
\end{itemize}
```

- El primer punto a destacar
- el segundo...
- ... y así sucesivamente



- Para destacar un elemento se utiliza el entorno **block**

```
\begin{block}{Importante}
El título es opcional, pero siempre
hay que incluir las llaves:
  \verb+\begin{block}{}+
\end{block}
```

### Importante

El título es opcional, pero siempre hay que incluir las llaves: `\begin{block}{}`

- ...o alguno de sus primos

### alertblock

```
\begin{alertblock}{\tt alertblock}
...
\end{alertblock}
```

### exampleblock

```
\begin{exampleblock}{\tt exampleblock}
...
\end{exampleblock}
```



- Para destacar un elemento se utiliza el entorno **block**

```
\begin{block}{Importante}
El título es opcional, pero siempre
hay que incluir las llaves:
  \verb+\begin{block}{}+
\end{block}
```

### Importante

El título es opcional, pero siempre hay que incluir las llaves: `\begin{block}{}{}`

- ...o alguno de sus primos

### alertblock

```
\begin{alertblock}{\tt alertblock}
...
\end{alertblock}
```

### exampleblock

```
\begin{exampleblock}{\tt exampleblock}
...
\end{exampleblock}
```



# Entornos tipo teorema

- BEAMER define los entornos `theorem`, `corollary`, `proof`, `definition`, `definitions`, `fact`, `example`, y `examples`; y utiliza bloques con diferentes estilos para representarlos
- Los encabezados salen en inglés
- Excepto por `proof`, que le hace caso al paquete `babel`

## Theorem (Un resultado)

```
\begin{theorem}[Un resultado]  
...  
\end{theorem}
```

## Demostración.

```
\begin{proof}  
...  
\end{proof}
```





# Entornos tipo teorema

- BEAMER define los entornos `theorem`, `corollary`, `proof`, `definition`, `definitions`, `fact`, `example`, y `examples`; y utiliza bloques con diferentes estilos para representarlos
- Los encabezados salen en inglés
- Excepto por `proof`, que le hace caso al paquete `babel`

## Theorem (Un resultado)

```
\begin{theorem}[Un resultado]  
...  
\end{theorem}
```

## Demostración.

```
\begin{proof}  
...  
\end{proof}
```





# Entornos tipo teorema

- BEAMER define los entornos `theorem`, `corollary`, `proof`, `definition`, `definitions`, `fact`, `example`, y `examples`; y utiliza bloques con diferentes estilos para representarlos
- Los encabezados salen en inglés
- Excepto por `proof`, que le hace caso al paquete `babel`

## Theorem (Un resultado)

```
\begin{theorem}[Un resultado]  
...  
\end{theorem}
```

## Demostración.

```
\begin{proof}  
...  
\end{proof}
```





# Entornos tipo teorema

En español

- Para cambiar los encabezados es necesario definir nuevos entornos tipo teorema utilizando los estilos proporcionados por el paquete `amsthm` (cargado automáticamente por `BEAMER`)

## Preámbulo

```
\theoremstyle{plain} % default
  \newtheorem{thm}{Teorema}

\theoremstyle{definition}
  \newtheorem{defn}{Definición}
  \newtheorem{exmp}{Ejemplo}

\theoremstyle{remark}
  \newtheorem{rem}{Nota}
```

## Teorema (Un resultado)

```
\begin{thm}[Un resultado]
...
\end{thm}
```

## Prueba.

```
\begin{proof}[Prueba]
...
\end{proof}
```

- El entorno `proof` sustituye el encabezado por el texto opcional



# Entornos tipo teorema

En español

- Para cambiar los encabezados es necesario definir nuevos entornos tipo teorema utilizando los estilos proporcionados por el paquete `amsthm` (cargado automáticamente por `BEAMER`)

## Preámbulo

```
\theoremstyle{plain} % default
  \newtheorem{thm}{Teorema}

\theoremstyle{definition}
  \newtheorem{defn}{Definición}
  \newtheorem{exmp}{Ejemplo}

\theoremstyle{remark}
  \newtheorem{rem}{Nota}
```

## Teorema (Un resultado)

```
\begin{thm}[Un resultado]
...
\end{thm}
```

## Prueba.

```
\begin{proof}[Prueba]
...
\end{proof}
```

- El entorno `proof` sustituye el encabezado por el texto opcional



# Texto recuadrado

Comando `beamercolorbox`

- Podemos recuadrar texto (o ecuaciones, o figuras) con los comandos estándar de L<sup>A</sup>T<sub>E</sub>X `\fbox` y `\frame`

```
\fbox{Texto}\  
\frame{recuadrado}
```

Texto  
recuadrado

- BEAMER proporciona además el comando `\beamercolorbox` para crear recuadros con colores
- Hay que definir previamente el `beamercolor`, que es una estructura que define al menos los colores de fondo y frente

```
\setbeamercolor{postit}{fg=black,bg=yellow}  
\begin{beamercolorbox}[sep=1em,wd=2cm]{postit}  
Postit!  
\end{beamercolorbox}
```

Postit!



# Texto recuadrado

Comando `beamercolorbox`

- Podemos recuadrar texto (o ecuaciones, o figuras) con los comandos estándar de L<sup>A</sup>T<sub>E</sub>X `\fbox` y `\frame`

```
\fbox{Texto}\  
\frame{recuadrado}
```

Texto  
recuadrado

- BEAMER proporciona además el comando `\beamercolorbox` para crear recuadros con colores
- Hay que definir previamente el `beamercolor`, que es una estructura que define al menos los colores de fondo y frente

```
\setbeamercolor{postit}{fg=black,bg=yellow}  
\begin{beamercolorbox}[sep=1em,wd=2cm]{postit}  
Postit!  
\end{beamercolorbox}
```

Postit!



# Texto recuadrado

Comando `beamercolorbox`

- Podemos recuadrar texto (o ecuaciones, o figuras) con los comandos estándar de L<sup>A</sup>T<sub>E</sub>X `\fbox` y `\frame`

```
\fbox{Texto}\  
\frame{recuadrado}
```

Texto  
recuadrado

- BEAMER proporciona además el comando `\beamercolorbox` para crear recuadros con colores
- Hay que definir previamente el `beamercolor`, que es una estructura que define al menos los colores de fondo y frente

```
\setbeamercolor{postit}{fg=black,bg=yellow}  
\begin{beamercolorbox}[sep=1em,wd=2cm]{postit}  
Postit!  
\end{beamercolorbox}
```

Postit!



# Texto recuadrado II

## Opciones de beamercolorbox

- Algo útil de los **beamercolor**, es que se puede jugar con los colores propios del template

```
\begin{beamercolorbox}[sep=1em]{block title}  
  Postit!  
\end{beamercolorbox}
```

Postit!

- Con las opciones, se puede controlar el tamaño y la alineación

```
[sep=1em,wd=5cm,center]
```

```
[sep=1em,wd=6cm,ht=.5in,right]
```

- ... y también alguna decoración

```
[...,rounded=true,shadow=true]
```



# Texto recuadrado II

## Opciones de beamercolorbox

- Algo útil de los `beamercolor`, es que se puede jugar con los colores propios del template

```
\begin{beamercolorbox}[sep=1em]{block title}  
  Postit!  
\end{beamercolorbox}
```

Postit!

- Con las opciones, se puede controlar el tamaño y la alineación

```
[sep=1em,wd=5cm,center]
```

```
[sep=1em,wd=6cm,ht=.5in,right]
```

- ... y también alguna decoración

```
[...,rounded=true,shadow=true]
```



# Texto recuadrado II

## Opciones de beamercolorbox

- Algo útil de los `beamercolor`, es que se puede jugar con los colores propios del template

```
\begin{beamercolorbox}[sep=1em]{block title}  
  Postit!  
\end{beamercolorbox}
```

Postit!

- Con las opciones, se puede controlar el tamaño y la alineación

```
[sep=1em,wd=5cm,center]
```

```
[sep=1em,wd=6cm,ht=.5in,right]
```

- ... y también alguna decoración

```
[...,rounded=true,shadow=true]
```



# Columnas

## Entorno columns

```
\begin{columns}
  \column{.5\textwidth}
  \framebox[\textwidth]{
    Contenido de la primera columna.
  }
  \column{.5\textwidth}
  \framebox[\textwidth]{
    Contenido de la segunda columna.
  }
\end{columns}
```

Contenido de la primera columna.

Contenido de la segunda columna.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.

```
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}
```

■ Todo

---

<sup>4</sup>El comportamiento por defecto es invisible.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.

```
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}
```

- Todo
- lo que sube

---

<sup>4</sup>El comportamiento por defecto es invisible.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.

```
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}
```

- Todo
- lo que sube
- baja.

---

<sup>4</sup>El comportamiento por defecto es invisible.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.
- El comando `\setbeamercovered{transparent}` hace que lo *cubierto* sea ligeramente visible, con un efecto de transparencia<sup>4</sup>

```

% Preámbulo, o fuera de frames
\setbeamercovered{transparent}
...
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}

```

- Todo
- lo que sube
- baja.

<sup>4</sup>El comportamiento por defecto es invisible.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.
- El comando `\setbeamercovered{transparent}` hace que lo *cubierto* sea ligeramente visible, con un efecto de transparencia<sup>4</sup>

```

% Preámbulo, o fuera de frames
\setbeamercovered{transparent}
...
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}

```

- Todo
- lo que sube
- baja.

<sup>4</sup>El comportamiento por defecto es invisible.



# Overlays (velos) básicos con pause

- El comando `\pause` permite generar overlays para visualización paso a paso.
- El comando `\setbeamercovered{transparent}` hace que lo *cubierto* sea ligeramente visible, con un efecto de transparencia<sup>4</sup>

```
% Preámbulo, o fuera de frames
\setbeamercovered{transparent}
...
\begin{itemize}
\item Todo \pause
\item lo que sube \pause
\item baja.
\end{itemize}
```

- Todo
- lo que sube
- baja.

---

<sup>4</sup>El comportamiento por defecto es invisible.



# Texto verbatim

## Ejemplo 2

- El comando `\verb` y el entorno `verbatim` de `LATEX` no se pueden usar directamente en `BEAMER`.
- Se debe añadir la opción `fragile` al frame:

```
\begin{frame}[fragile]
\frametitle{Texto verbatim}
El texto contenido en un entorno
\verb+verbatim+ se imprime tal
cual, con todos los saltos de
línea y espacios, sin interpretar
ningún comando de \LaTeX.
```

```
\begin{verbatim}
10 PRINT "HELLO WORLD";
20 GOTO 10
\end{verbatim}
\end{frame}
```

Nota: La opción `fragile` exige compilar dos veces

### Texto verbatim

El texto contenido en un entorno `verbatim` se imprime tal cual, con todos los saltos de línea y espacios, sin interpretar ningún comando de `LATEX`.

```
10 PRINT "HELLO WORLD";
20 GOTO 10
```



# Texto verbatim

## Ejemplo 2

- El comando `\verb` y el entorno `verbatim` de `LATEX` no se pueden usar directamente en `BEAMER`.
- Se debe añadir la opción **fragile** al frame:

```
\begin{frame}[fragile]
\frametitle{Texto verbatim}
El texto contenido en un entorno
\verb+verbatim+ se imprime tal
cual, con todos los saltos de
línea y espacios, sin interpretar
ningún comando de \LaTeX.
```

```
\begin{verbatim}
10 PRINT "HELLO WORLD";
20 GOTO 10
\end{verbatim}
\end{frame}
```

Nota: La opción `fragile` exige compilar dos veces





# Texto verbatim

## Ejemplo 2

- El comando `\verb` y el entorno `verbatim` de `LATEX` no se pueden usar directamente en `BEAMER`.
- Se debe añadir la opción **fragile** al frame:

```
\begin{frame}[fragile]
\frametitle{Texto verbatim}
El texto contenido en un entorno
\verb+verbatim+ se imprime tal
cual, con todos los saltos de
línea y espacios, sin interpretar
ningún comando de \LaTeX.
```

```
\begin{verbatim}
10 PRINT "HELLO WORLD";
20 GOTO 10
\end{verbatim}
\end{frame}
```

Nota: La opción `fragile` exige compilar dos veces

### Texto verbatim

El texto contenido en un entorno `verbatim` con todos los saltos de línea y espacios, sin comando de `LATEX`.

```
10 PRINT "HELLO WORLD";
20 GOTO 10
```



# Texto verbatim

## Ejemplo 2

- El comando `\verb` y el entorno `verbatim` de  $\text{\LaTeX}$  no se pueden usar directamente en `BEAMER`.
- Se debe añadir la opción **fragile** al frame:

```
\begin{frame}[fragile]
\frametitle{Texto verbatim}
El texto contenido en un entorno
\verb+verbatim+ se imprime tal
cual, con todos los saltos de
línea y espacios, sin interpretar
ningún comando de  $\text{\LaTeX}$ .
```

```
\begin{verbatim}
10 PRINT "HELLO WORLD";
20 GOTO 10
\end{verbatim}
\end{frame}
```

Nota: La opción `fragile` exige compilar dos veces

### Texto verbatim

El texto contenido en un entorno `verbatim` con todos los saltos de línea y espacios, sin comando de  $\text{\LaTeX}$ .

```
10 PRINT "HELLO WORLD";
20 GOTO 10
```



# Opciones de *clase* y de *frame*

- Todo proyecto de BEAMER se compone de una serie de **frames**.
- Cada frame contiene una o más **slides**
- Algunas **opciones** interesantes

## Algunas **opciones** útiles

Alineación vertical	[c b t]
Frame vacío	[plain]
Contenido verbatim	[fragile]

```
\begin{frame}[<opciones>]  
  \frametitle{<titulo>}  
  \framesubtitle{<subtitulo>}  
  <contenido>  
\end{frame}
```



# Opciones de *clase* y de *frame*

- Todo proyecto de BEAMER se compone de una serie de **frames**.
- Cada frame contiene una o más **slides**
- Algunas **opciones** interesantes

## Algunas **opciones** útiles

Alineación vertical	[c b t]
Frame vacío	[plain]
Contenido verbatim	[fragile]

```
\begin{frame}[<opciones>]
  \frametitle{<titulo>}
  \framesubtitle{<subtitulo>}
  <contenido>
\end{frame}
```



# Opciones de *clase* y de *frame*

- Todo proyecto de BEAMER se compone de una serie de **frames**.
- Cada frame contiene una o más **slides**
- Algunas **opciones** interesantes

## Algunas **opciones** útiles

Alineación vertical	[c b t]
Frame vacío	[plain]
Contenido verbatim	[fragile]

```
\begin{frame}[<opciones>]  
  \frametitle{<titulo>}  
  \framesubtitle{<subtitulo>}  
  <contenido>  
\end{frame}
```



# Ejercicio 1

## Compilación y templates

- 1 Editar y compilar el documento básico de BEAMER  
(../ejercicios/1-basico.tex)
- 2 Añadir datos de autores e instituciones
- 3 Añadir el logo de la UJI, bajo la carpeta `img`
- 4 Añadir unos cuantos frames y dividirlos en secciones
- 5 Crear una tabla de contenidos
- 6 Elegir un theme y un color theme, incorporarlos y volver a compilar
- 7 Compilar los templates que vienen con BEAMER y visualizarlos



## Sección 2 | Control avanzado de overlays





# Overlay specifications

- Ya hemos visto el uso del comando `\pause` para generar **overlays**.
- Más generalmente, podemos tener control sobre qué se visualiza exactamente en cada slide mediante las **overlay specifications**

## Sintaxis

Lista separada por comas de rangos de slides.

La especificación

`-3,6-8,10,12-15`

seleccionará las slides

`1, 2, 3, 6, 7, 8, 10, 12, 13, 14, 15`



# Overlay specifications

- Ya hemos visto el uso del comando `\pause` para generar **overlays**.
- Más generalmente, podemos tener control sobre qué se visualiza exactamente en cada slide mediante las **overlay specifications**

## Sintaxis

Lista separada por comas de rangos de slides.

La especificación

**-3,6-8,10,12-15**

seleccionará las slides

1, 2, 3, 6, 7, 8, 10, 12, 13, 14, 15



# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

- Todo
- lo que sube
- baja





# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

- Todo
- lo que sube
- baja





# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

- Todo
- lo que sube
- baja





# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja





# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja





# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja



# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja



# Visualización paso a paso

```
\begin{itemize}
  \item<1-> Todo
  \item<2-> lo que sube
  \item<3-> baja
\end{itemize}
```

- Todo
- lo que sube
- baja

```
\begin{itemize}
  \item<1-> Todo
  \item<3> lo que sube
  \item<2-4> baja
\end{itemize}
```

- Todo
- lo que sube
- baja



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

negrita sólo en 2

negrita de 3 en adelante

```
\begin{itemize}
 \item<4-> \alert<4>{cada comando o entorno}
 \item<5-> \alert<5>{actúa diferente frente}
 \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

■ cada comando o entorno

■ actúa diferente frente

■ a una especificación



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

**negrita sólo en 2**

negrita de 3 en adelante

```
\begin{itemize}
  \item<4-> \alert<4>{cada comando o entorno}
  \item<5-> \alert<5>{actúa diferente frente}
  \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

■ cada comando o entorno

■ actúa diferente frente

■ a una especificación



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

negrita sólo en 2

**negrita de 3 en adelante**

```
\begin{itemize}
  \item<4-> \alert<4>{cada comando o entorno}
  \item<5-> \alert<5>{actúa diferente frente}
  \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

- cada comando o entorno
- actúa diferente frente
- a una especificación



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

negrita sólo en 2

**negrita de 3 en adelante**

```
\begin{itemize}
  \item<4-> \alert<4>{cada comando o entorno}
  \item<5-> \alert<5>{actúa diferente frente}
  \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

■ cada comando o entorno

■ actúa diferente frente

■ a una especificación



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

negrita sólo en 2

**negrita de 3 en adelante**

```
\begin{itemize}
 \item<4-> \alert<4>{cada comando o entorno}
 \item<5-> \alert<5>{actúa diferente frente}
 \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

- cada comando o entorno
- **actúa diferente frente**
- a una especificación



# Overlay specifications en comandos

- Las especificaciones se pueden utilizar junto con otros comandos.
- El efecto depende de cada caso.

```
\textbf{negrita}
\textbf<2>{negrita sólo en 2}
\textbf<3->{negrita de 3 en adelante}
```

**negrita**

negrita sólo en 2

**negrita de 3 en adelante**

```
\begin{itemize}
 \item<4-> \alert<4>{cada comando o entorno}
 \item<5-> \alert<5>{actúa diferente frente}
 \item<6-> \alert<6>{a una especificación}
\end{itemize}
```

- cada comando o entorno
- actúa diferente frente
- **a una especificación**



# Comandos compatibles

```
\textbf<2>{Ejemplo}  
\textit<2>{Ejemplo}  
\textsl<2>{Ejemplo}  
\alert<2>{Ejemplo}  
\textrm<2>{Ejemplo}  
\textsf<2>{Ejemplo}  
\color<2>{green} Ejemplo  
\structure<2>{Ejemplo}
```

Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo  
Ejemplo



# Comandos compatibles

```
\textbf<2>{Ejemplo}  
\textit<2>{Ejemplo}  
\textsl<2>{Ejemplo}  
\alert<2>{Ejemplo}  
\textrm<2>{Ejemplo}  
\textsf<2>{Ejemplo}  
\color<2>{green} Ejemplo  
\structure<2>{Ejemplo}
```

## Ejemplo

*Ejemplo*

*Ejemplo*

**Ejemplo**

Ejemplo

Ejemplo

Ejemplo

Ejemplo



# Comandos específicos de overlays

- Comandos para mostrar contenido en los overlays especificados.
- Se diferencian en su comportamiento en los overlays **no especificados**.
- En dos aspectos: cubierto/invisible y espacio ocupado.

```
|\only<2>{only}| \\  
|\uncover<2>{uncover}| \\  
|\visible<2>{visible}|
```

Hay más comandos específicos (`\onslide`, `\invisible`, `\alt`, `\temporal`), pero todos se pueden reducir a estos tres. Más info: BEAMER User's guide [Tantau, 2010, s.9.3].



# Comandos específicos de overlays

- Comandos para mostrar contenido en los overlays especificados.
- Se diferencian en su comportamiento en los overlays **no especificados**.
- En dos aspectos: cubierto/invisible y espacio ocupado.

```
|\only<2>{only}| \\  
|\uncover<2>{uncover}| \\  
|\visible<2>{visible}|
```

```
|only|  
|uncover|  
|visible|
```

Hay más comandos específicos (`\onslide`, `\invisible`, `\alt`, `\temporal`), pero todos se pueden reducir a estos tres. Más info: *BEAMER User's guide* [Tantau, 2010, s.9.3].



# Comandos específicos de overlays

- Comandos para mostrar contenido en los overlays especificados.
- Se diferencian en su comportamiento en los overlays **no especificados**.
- En dos aspectos: cubierto/invisible y espacio ocupado.

```
|\only<2>{only}| \\  
|\uncover<2>{uncover}| \\  
|\visible<2>{visible}|
```

Hay más comandos específicos (`\onslide`, `\invisible`, `\alt`, `\temporal`), pero todos se pueden reducir a estos tres. Más info: *BEAMER User's guide* [Tantau, 2010, s.9.3].



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+>`, con el mismo efecto.

```
\begin{itemize}
  \item<1-> Uno
  \item<2-> Dos
  \item<3-> Tres
  \item<4-> Cuatro
\end{itemize}
```

■ Uno

■ Dos

■ Tres

■ Cuatro



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+>`, con el mismo efecto.

```
\begin{itemize}
  \item<1-> Uno
  \item<2-> Dos
  \item<3-> Tres
  \item<4-> Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+>`, con el mismo efecto.

```
\begin{itemize}
  \item<1-> Uno
  \item<2-> Dos
  \item<3-> Tres
  \item<4-> Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+>`, con el mismo efecto.

```
\begin{itemize}
  \item<1-> Uno
  \item<2-> Dos
  \item<3-> Tres
  \item<4-> Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+>`, con el mismo efecto.

```
\begin{itemize}
  \item<1-> Uno
  \item<2-> Dos
  \item<3-> Tres
  \item<4-> Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro



# Especificaciones incrementales

- Las especificaciones con números absolutos no son robustas frente a cambios.
- Hay ciertas configuraciones típicas que son farragosas de escribir.
- Por eso se puede incorporar en el entorno `itemize` o `enumerate` la **especificación incremental** `<+-->`, con el mismo efecto.

```
\begin{itemize}[<+-->]  
  \item Uno  
  \item Dos  
  \item Tres  
  \item Cuatro  
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+ - | alert@+>`.

```
\begin{itemize}[<+ - | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno

- Dos

- Tres

- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification[<+ - | alert@+>].
```

Nota: pierde su efecto en los frames `[fragile]`.



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+- | alert@+>`.

```
\begin{itemize}[<+- | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification{<+- | alert@+>}
```

Nota: pierde su efecto en los frames `[fragile]`.



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+- | alert@+>`.

```
\begin{itemize}[<+- | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno

- Dos

- Tres

- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification{<+- | alert@+>}
```

Nota: pierde su efecto en los frames `[fragile]`.



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+- | alert@+>`.

```
\begin{itemize}[<+- | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification{<+- | alert@+>}
```

Nota: pierde su efecto en los frames `[fragile]`.



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+- | alert@+>`.

```
\begin{itemize}[<+- | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification{<+- | alert@+>}
```

Nota: pierde su efecto en los frames `[fragile]`.



# Especificaciones de acción

- Los entornos `itemize` o `enumerate` también admiten la especificación de acción `<+- | alert@+>`.

```
\begin{itemize}[<+- | alert@+>]
  \item Uno
  \item Dos
  \item Tres
  \item Cuatro
\end{itemize}
```

- Uno
- Dos
- Tres
- Cuatro

## Especificación global

Se puede predeterminar un comportamiento para todos los entornos de este tipo en una presentación estableciendo en el preámbulo del documento el parámetro global

```
\beamerdefaultoverlayspecification{<+- | alert@+>}
```

Nota: pierde su efecto en los frames `[fragile]`.



# Overlays con imágenes

- Problema 1: el **velo** no funciona sobre la imagen.
- Problema 2: el contenido estático se mueve.

```
|\only<2>{\includegraphics{fig}}|  
|\uncover<2>{\includegraphics{fig}}|  
|\visible<2>{\includegraphics{fig}}|
```





# Overlays con imágenes

- Problema 1: el **velo** no funciona sobre la imagen.
- Problema 2: el contenido estático se mueve.

```
|\only<2>{\includegraphics{fig}}|  
|\uncover<2>{\includegraphics{fig}}|  
|\visible<2>{\includegraphics{fig}}|
```





# Imágenes con velo

## Ejemplo 3

Solución problema 1: Un truquillo utilizando el paquete **TikZ** [Tantau, 2008]

```
\usepackage{tikz} % preámbulo
...
\begin{itemize}[<+>-]
  \item Una visualización paso a paso
  \item queda muy desagradable
  \item cuando hay elementos gráficos
    {\centering
     \includegraphics{fig} \par}
  \item a menos que se ponga
  \item como un nodo tikz
\begin{center}
  \begin{tikzpicture}
    \alt<5->
      {\node[opacity=1]
        {\includegraphics{fig}};}
      {\node[opacity=.15]
        {\includegraphics{fig}};}
  \end{tikzpicture}
\end{center}
\end{itemize}
```

- Una visualización paso a paso
- queda muy desagradable
- cuando hay elementos gráficos
- a menos que se ponga
- como un nodo tikz





# Imágenes con velo

## Ejemplo 3

Solución problema 1: Un truquillo utilizando el paquete **TikZ** [Tantau, 2008]

```
\usepackage{tikz} % preámbulo
...
\begin{itemize}[<+>-]
  \item Una visualización paso a paso
  \item queda muy desagradable
  \item cuando hay elementos gráficos
    {\centering
     \includegraphics{fig} \par}
  \item a menos que se ponga
  \item como un nodo tikz
\begin{center}
  \begin{tikzpicture}
    \alt<5->
      {\node[opacity=1]
        {\includegraphics{fig}};}
      {\node[opacity=.15]
        {\includegraphics{fig}};}
  \end{tikzpicture}
\end{center}
\end{itemize}
```

- Una visualización paso a paso
- queda muy desagradable
- cuando hay elementos gráficos
- a menos que se ponga
- como un nodo tikz





# Imágenes con velo

## Ejemplo 3

Solución problema 1: Un truquillo utilizando el paquete **TikZ** [Tantau, 2008]

```
\usepackage{tikz} % preámbulo
...
\begin{itemize}[<+>-]
  \item Una visualización paso a paso
  \item queda muy desagradable
  \item cuando hay elementos gráficos
    {\centering
     \includegraphics{fig} \par}
  \item a menos que se ponga
  \item como un nodo tikz
\begin{center}
  \begin{tikzpicture}
    \alt<5->
      {\node[opacity=1]
        {\includegraphics{fig}};}
      {\node[opacity=.15]
        {\includegraphics{fig}};}
  \end{tikzpicture}
\end{center}
\end{itemize}
```

- Una visualización paso a paso
- queda muy desagradable
- cuando hay elementos gráficos



- a menos que se ponga
- como un nodo tikz





# Imágenes con velo

## Ejemplo 3

Solución problema 1: Un truquillo utilizando el paquete **TikZ** [Tantau, 2008]

```
\usepackage{tikz} % preámbulo
...
\begin{itemize}[<+>-]
  \item Una visualización paso a paso
  \item queda muy desagradable
  \item cuando hay elementos gráficos
    {\centering
     \includegraphics{fig} \par}
  \item a menos que se ponga
  \item como un nodo tikz
\begin{center}
  \begin{tikzpicture}
    \alt<5->
      {\node[opacity=1]
        {\includegraphics{fig}};}
      {\node[opacity=.15]
        {\includegraphics{fig}};}
  \end{tikzpicture}
\end{center}
\end{itemize}
```

- Una visualización paso a paso
- queda muy desagradable
- cuando hay elementos gráficos
- a menos que se ponga
- como un nodo tikz





# Imágenes con velo

## Ejemplo 3

Solución problema 1: Un truquillo utilizando el paquete **TikZ** [Tantau, 2008]

```
\usepackage{tikz} % preámbulo
...
\begin{itemize}[<+>-]
  \item Una visualización paso a paso
  \item queda muy desagradable
  \item cuando hay elementos gráficos
    {\centering
     \includegraphics{fig} \par}
  \item a menos que se ponga
  \item como un nodo tikz
\begin{center}
  \begin{tikzpicture}
    \alt<5->
      {\node[opacity=1]
        {\includegraphics{fig}};}
      {\node[opacity=.15]
        {\includegraphics{fig}};}
  \end{tikzpicture}
\end{center}
\end{itemize}
```

- Una visualización paso a paso
  - queda muy desagradable
  - cuando hay elementos gráficos
- 
- a menos que se ponga
  - como un nodo tikz
- 



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\  

  \only<+>{como por ejemplo\<\  

    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\  

  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\<\  

    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado
- para que las cosas no salten



# Contenido dinámico

## Solución problema 2: entorno `overlayarea`

```
\begin{itemize}[<+>]
\item El contenido dinámico \\
  \only<+>{como por ejemplo\\
    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\
  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\\
    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}
```

- El contenido dinámico como por ejemplo estas dos líneas
- necesita un espacio reservado
- para que las cosas no salten



# Contenido dinámico

## Solución problema 2: entorno `overlayarea`

```
\begin{itemize}[<+>]
\item El contenido dinámico \\
  \only<+>{como por ejemplo\\
    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\
  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\\
    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}
```

- El contenido dinámico que cambian a una sola
- necesita un espacio reservado
- para que las cosas no salten



# Contenido dinámico

## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\
  \only<+>{como por ejemplo\\
    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\
  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\\
    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico



o una imagen que aparece



- necesita un espacio reservado

- para que las cosas no salten



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\  

  \only<+>{como por ejemplo\\  

    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece  

    \includegraphics{fig}}
\item necesita un espacio reservado \\  

  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\\  

    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado
- para que las cosas no salten



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\  

  \only<+>{como por ejemplo\<\  

    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\  

  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\<\  

    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado
  - donde quepa todo el contenido dinámico
- para que las cosas no salten



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\  

  \only<+>{como por ejemplo\<\  

    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\  

  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\<\  

    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado  
que se quiera poner
- para que las cosas no salten



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\
  \only<+>{como por ejemplo\\
    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\
  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\\
    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado



- para que las cosas no salten



## Solución problema 2: entorno `overlayarea`

```

\begin{itemize}[<+>]
\item El contenido dinámico \\  

  \only<+>{como por ejemplo\ \  

    estas dos líneas}
  \only<+>{que cambian a una sola}
  \only<+>{o una imagen que aparece
    \includegraphics{fig}}
\item necesita un espacio reservado \\  

  \begin{center}
  \begin{overlayarea}{<width>}{<height>}
  \only<+>{donde quepa todo\ \  

    el contenido dinámico}
  \only<+>{que se quiera poner}
  \only<+>{\includegraphics{fig}}
  \end{overlayarea}
  \end{center}
\item para que las cosas no salten
\end{itemize}

```

- El contenido dinámico
- necesita un espacio reservado
- para que las cosas no salten



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin{frame}}
\alert<+>{ \frametitle{Transparencia}}
\alert<+>{ \framesubtitle{Beamer}}
\alert<+>{ Contenido primero}
\alert<+>{\end{frame}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin{frame}}
\alert<+>{ \frametitle{Transparencia}}
\alert<+>{ \framesubtitle{Beamer}}
\alert<+>{ Contenido primero}
\alert<+>{\end{frame}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin{frame}}
\alert<+>{ \frametitle{Transparencia}}
\alert<+>{ \framesubtitle{Beamer}}
\alert<+>{ Contenido primero}
\alert<+>{\end{frame}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin{frame}}
\alert<+>{ \frametitle{Transparencia}}
\alert<+>{ \framesubtitle{Beamer}}
\alert<+>{ Contenido primero}
\alert<+>{\end{frame}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin{frame}}
\alert<+>{ \frametitle{Transparencia}}
\alert<+>{ \framesubtitle{Beamer}}
\alert<+>{ Contenido primero}
\alert<+>{\end{frame}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```

\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}

```

```

\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}

```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Overlays con texto verbatim

- El entorno `verbatim` de L<sup>A</sup>T<sub>E</sub>X permite enseñar texto preformateado <sup>5</sup>.
- En BEAMER es necesario utilizar la opción `fragile` del frame.
- BEAMER también define el entorno `semiverbatim`, en el cual la retrobarra `\` y las llaves `{ }` mantienen su significado.
- Esto permite enfatizar, o mostrar paso a paso utilizando overlays.
- Los símbolos `\`, `{`, y `}` se imprimen anteponiendo una retrobarra.

```
\begin{semiverbatim}
\alert<+>{\begin\{frame\}}
\alert<+>{ \frametitle\{Transparencia\}}
\alert<+>{ \framesubtitle\{Beamer\}}
\alert<+>{ Contenido primero}
\alert<+>{\end\{frame\}}
\end{semiverbatim}
```

```
\begin{frame}
\frametitle{Transparencia}
\framesubtitle{Beamer}
Contenido primero
\end{frame}
```

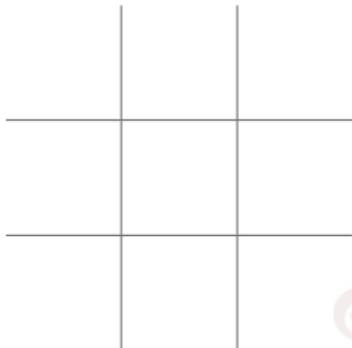
<sup>5</sup>También hay otros más poderosos, como `lstlisting`.



# Ejercicio 2

## Diapositivas dinámicas

1 Reproducir el siguiente **tres en raya**

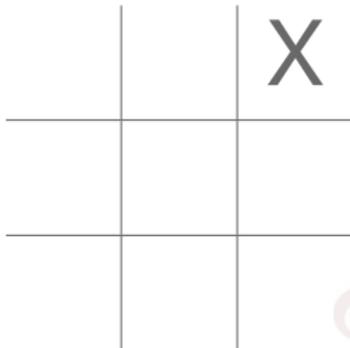




## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

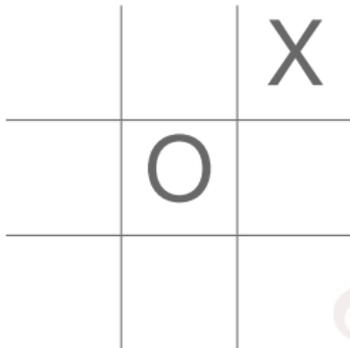




## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**





## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

		X
	O	
		X





## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

		X
	O	O
		X





## Ejercicio 2

### Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

		X
X	O	O
		X





## Ejercicio 2

### Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

		X
X	O	O
	O	X





## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

	X	X
X	O	O
	O	X





## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

O	X	X
X	O	O
	O	X





## Ejercicio 2

Diapositivas dinámicas

- 1 Reproducir el siguiente **tres en raya**

O	X	X
X	O	O
X	O	X





## Sección 3 | Handouts y modos





# Compilando handouts

```
\documentclass[handout]{beamer}
\mode<handout>{
  \usepackage{pgfpages}
  \pgfpagesuselayout{resize}[a4paper,border shrink=5mm,landscape]
  \pgfpagesuselayout{2 on 1}[a4paper,border shrink=5mm]
  \setbeamercolor{background canvas}{bg=black!5}
}
```

- La opción de clase **handout** colapsa las slides de cada frame en una única diapositiva
- BEAMER entra entonces en **modo handout**. El comando **mode** permite ajustar algunas configuraciones que se aplican solo bajo este modo. Por ejemplo, un color de fondo.
- Para la impresión, el paquete **pgfpages** [Tantau, 2008, s.46] permite ajustar el tamaño de las diapositivas.
- Incluso permite imprimir múltiples diapositivas por página también admite `{4 on 1}[landscape]`; `{8 on 1}` o `{16 on 1}[landscape]`.



# Compilando handouts

```

\documentclass[handout]{beamer}
\mode<handout>{
  \usepackage{pgfpages}
  \pgfpagesuselayout{resize}[a4paper,border shrink=5mm,landscape]
  \pgfpagesuselayout{2 on 1}[a4paper,border shrink=5mm]
  \setbeamercolor{background canvas}{bg=black!5}
}

```

- La opción de clase **handout** colapsa las slides de cada frame en una única diapositiva
- BEAMER entra entonces en **modo handout**. El comando **mode** permite ajustar algunas configuraciones que se aplican sólo bajo este modo. Por ejemplo, un color de fondo.
- Para la impresión, el paquete **pgfpages** [Tantau, 2008, s.46] permite ajustar el tamaño de las diapositivas.
- Incluso permite imprimir múltiples diapositivas por página también admite `{4 on 1}[landscape]`; `{8 on 1}` o `{16 on 1}[landscape]`.



# Compilando handouts

```
\documentclass[handout]{beamer}
\mode<handout>{
  \usepackage{pgfpages}
  \pgfpagesuselayout{resize}[a4paper,border shrink=5mm,landscape]
  \pgfpagesuselayout{2 on 1}[a4paper,border shrink=5mm]
  \setbeamercolor{background canvas}{bg=black!5}
}
```

- La opción de clase **handout** colapsa las slides de cada frame en una única diapositiva
- BEAMER entra entonces en **modo handout**. El comando **mode** permite ajustar algunas configuraciones que se aplican sólo bajo este modo. Por ejemplo, un color de fondo.
- Para la impresión, el paquete **pgfpages** [Tantau, 2008, s.46] permite ajustar el tamaño de las diapositivas.
- Incluso permite imprimir múltiples diapositivas por página también admite `{4 on 1}[landscape]`; `{8 on 1}` o `{16 on 1}[landscape]`.



# Compilando handouts

```
\documentclass[handout]{beamer}
\mode<handout>{
  \usepackage{pgfpages}
  \pgfpagesuselayout{resize}[a4paper,border shrink=5mm,landscape]
  \pgfpagesuselayout{2 on 1}[a4paper,border shrink=5mm]
  \setbeamercolor{background canvas}{bg=black!5}
}
```

- La opción de clase **handout** colapsa las slides de cada frame en una única diapositiva
- BEAMER entra entonces en **modo handout**. El comando **mode** permite ajustar algunas configuraciones que se aplican sólo bajo este modo. Por ejemplo, un color de fondo.
- Para la impresión, el paquete **pgfpages** [Tantau, 2008, s.46] permite ajustar el tamaño de las diapositivas.
- Incluso permite imprimir múltiples diapositivas por página también admite `{4 on 1}[landscape]`; `{8 on 1}` o `{16 on 1}[landscape]`.



# BEAMER modes

- BEAMER reconoce distintos **modos** de trabajo, que permiten hacer varias versiones de un mismo documento.
- Ya hemos visto una aplicación del modo **handout** y el comando **mode**.

all

presentation

- beamer - modo por defecto.
- second - segunda pantalla.
- handout - impresos.
- trans - transparencias.
- article - control transferido a otra clase (article, book, etc.)



# Comando mode

```
\mode<modos>{contenido}
```

El contenido sólo se incluye en los modos especificados

## Ejemplo

```
\mode<article| handout>{  
  Detalles adicionales mencionados sólo  
  en una versión de artículo, o comentarios  
  que se desean dejar disponibles en una  
  versión handout  
}
```

Notar que los modos se separan por una barra vertical y un espacio.



# Comando mode

```
\mode<modos>{contenido}
```

El contenido sólo se incluye en los modos especificados

## Ejemplo

```
\mode<article| handout>{  
  Detalles adicionales mencionados sólo  
  en una versión de artículo, o comentarios  
  que se desean dejar disponibles en una  
  versión handout  
}
```

Notar que los modos se separan por una barra vertical y un espacio.

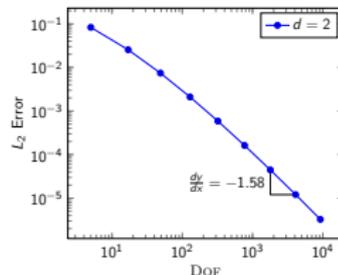


# Modes y overlays

## Ejemplo 4

Las especificaciones de overlay pueden ser distintas para cada modo. Una utilización típica del modo `handout` es hacer una versión estática de una diapositiva dinámica.

```
\includegraphics<1| handout:0>{fig-1.pdf}
\includegraphics<2| handout:0>{fig-2.pdf}
\includegraphics<3>{fig-3.pdf}
```



Esto significa: en el modo por defecto (`beamer`) se muestran las tres imágenes secuencialmente. En modo `handout` se suprimen las dos primeras.

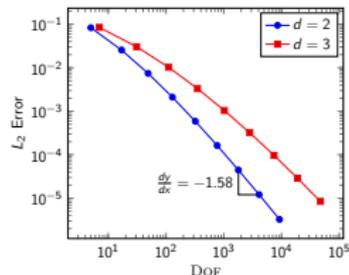


# Modes y overlays

## Ejemplo 4

Las especificaciones de overlay pueden ser distintas para cada modo. Una utilización típica del modo `handout` es hacer una versión estática de una diapositiva dinámica.

```
\includegraphics<1| handout:0>{fig-1.pdf}
\includegraphics<2| handout:0>{fig-2.pdf}
\includegraphics<3>{fig-3.pdf}
```



Esto significa: en el modo por defecto (`beamer`) se muestran las tres imágenes secuencialmente. En modo `handout` se suprimen las dos primeras.

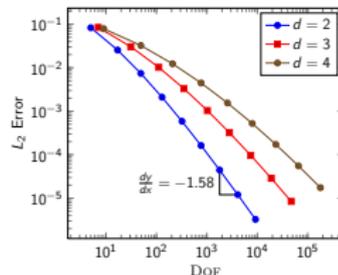


# Modes y overlays

## Ejemplo 4

Las especificaciones de overlay pueden ser distintas para cada modo. Una utilización típica del modo `handout` es hacer una versión estática de una diapositiva dinámica.

```
\includegraphics<1| handout:0>{fig-1.pdf}
\includegraphics<2| handout:0>{fig-2.pdf}
\includegraphics<3>{fig-3.pdf}
```



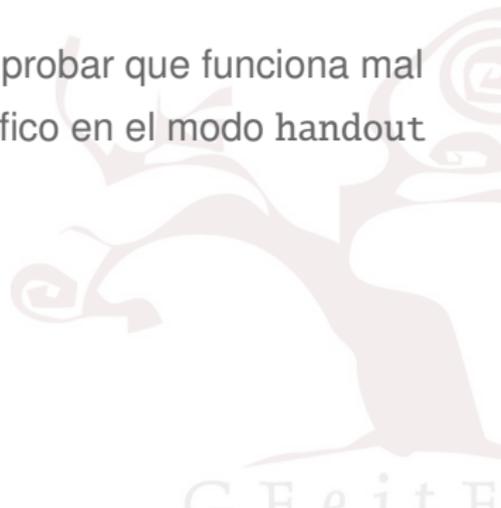
Esto significa: en el modo por defecto (`beamer`) se muestran las tres imágenes secuencialmente. En modo `handout` se suprimen las dos primeras.



# Ejercicio 3

## El modo Handout

- 1 Compilar la diapositiva del ejercicio 3 tal como está (modo beamer) y comprobar que funciona bien
- 2 Añadirle la opción de clase `handout` y comprobar que funciona mal
- 3 Corregirla para que sólo salga el tercer gráfico en el modo `handout`





## Sección 4 | Botones y enlaces





# Hyperlinks

## Ejemplo sencillo

Podemos crear **saltos** anticipadamente en la presentación usando hyperlinks.

```
\begin{frame}[label=ej_sencillo]{Hyperlinks}
  \hyperlink{ej_sofisticado}{{[Saltar al ejemplo sofisticado]}}
```

[Saltar al ejemplo sofisticado]

Observaciones:

- El destino debe estar etiquetado (label).
- El comando `\hyperlink` produce el enlace especificando destino y el texto activo.





# Hyperlinks

## Ejemplo sencillo

Podemos crear **saltos** anticipadamente en la presentación usando hyperlinks.

```
\begin{frame}[label=ej_sencillo]{Hyperlinks}
  \hyperlink{ej_sofisticado}{{[Saltar al ejemplo sofisticado]}}
```

[Saltar al ejemplo sofisticado]

Observaciones:

- El destino debe estar **etiquetado** (**label**).
- El comando `\hyperlink` produce el enlace especificando destino y el texto activo.



# Hyperlinks

## Ejemplo sencillo

Podemos crear **saltos** anticipadamente en la presentación usando hyperlinks.

```
\begin{frame}[label=ej_sencillo]{Hyperlinks}
  \hyperlink{ej_sofisticado}{{[Saltar al ejemplo sofisticado]}}
```

[Saltar al ejemplo sofisticado]

Observaciones:

- El destino debe estar **etiquetado** (**label**).
- El comando **\hyperlink** produce el enlace especificando destino y el texto activo.



# Hyperlinks

Ejemplo sofisticado

## Hyperlinks, overlays y botones

```
\begin{frame}[label=ej_sofisticado]{Hyperlinks}  
\hyperlink<3>{ej_sencillo<2>}{\beamerrturnbutton{Volver al ejemplo sencillo}}
```

### Observaciones:

- Ahora, el enlace no aparecerá hasta la slide #3.
- Además, nos mandará a la slide #2 del destino.
- Finalmente, los enlaces se pueden representar usando **botones**.



## Hyperlinks, overlays y botones

```
\begin{frame}[label=ej_sofisticado]{Hyperlinks}  
\hyperlink<3>{ej_sencillo<2>}{\beamerrturnbutton{Volver al ejemplo sencillo}}
```

### Observaciones:

- Ahora, el enlace no aparecerá hasta la slide #3.
- Además, nos mandará a la slide #2 del destino.
- Finalmente, los enlaces se pueden representar usando botones.



# Hyperlinks

Ejemplo sofisticado

## Hyperlinks, overlays y botones

```
\begin{frame}[label=ej_sofisticado]{Hyperlinks}  
\hyperlink<3>{ej_sencillo<2>}{\beamerreturnbutton{Volver al ejemplo sencillo}}
```

◀ Volver al ejemplo sencillo

### Observaciones:

- Ahora, el enlace no aparecerá hasta la slide #3.
- Además, nos mandará a la slide #2 del destino.
- Finalmente, los enlaces se pueden representar usando **botones**.



## Botones predefinidos

```
\beamerbutton{Texto}  
\beamergetobutton{Texto}  
\beamerskipbutton{Texto}  
\beamerreturnbutton{Texto}
```

Texto

▶ Texto

▶▶ Texto

◀ Texto

## Algunos<sup>6</sup> saltos predefinidos

```
\hyperlinkpresentationstart, \hyperlinkpresentationend,  
\hyperlinkframestartnext, \hyperlinkframeendprev, ...
```

---

<sup>6</sup>Hay varios, de remota utilidad. Ver BEAMER User's Guide [Tantau, 2010, s.11.1]



# Reanudar diapositivas inacabadas

Comando `againframe`

A veces puede interesar dejar parte del contenido de un frame para ser retomado más adelante (o no).

```
\begin{frame}<1-2>[label=volver]
\begin{enumerate}<+ - | alert@+>
  \item Restringir el rango...
  \item Etiquetar el frame
  \item Contenido adicional...
\end{enumerate}
\end{frame}
```

- Restringir el rango de slides que se visualizarán en primera instancia
- Etiquetar el frame
- Contenido adicional, para más tarde



# Reanudar diapositivas inacabadas

Comando `againframe`

A veces puede interesar dejar parte del contenido de un frame para ser retomado más adelante (o no).

```
\begin{frame}<1-2>[label=volver]
\begin{enumerate}<+- | alert@+>
  \item Restringir el rango...
  \item Etiquetar el frame
  \item Contenido adicional...
\end{enumerate}
\end{frame}
```

- 1 Restringir el rango de slides que se visualizarán en primera instancia.
- 2 Etiquetar el frame.
- 3 Contenido adicional, para más tarde.



# Reanudar diapositivas inacabadas

Comando `againframe`

A veces puede interesar dejar parte del contenido de un frame para ser retomado más adelante (o no).

```
\begin{frame}<1-2>[label=volver]
\begin{enumerate}<+- | alert@+>
  \item Restringir el rango...
  \item Etiquetar el frame
  \item Contenido adicional...
\end{enumerate}
\end{frame}
```

- 1 Restringir el rango de slides que se visualizarán en primera instancia.
- 2 **Etiquetar el frame.**
- 3 Contenido adicional, para más tarde.



# Reanudar diapositivas inacabadas

Comando `againframe`

```
\begin{frame}
```

Aquí se proporciona información más detallada sobre el punto 2.

```
\end{frame}
```

```
\againframe<3>{volver}
```

Aquí se proporciona información más detallada sobre el punto 2.

- El comando `againframe` se pone **fuera** del frame, en el lugar de la próxima diapositiva.
- Se indica el número de slide a partir del cual se quiere **reanudar**...
- ... la diapositiva con la etiqueta (**label**) especificada



# Reanudar diapositivas inacabadas

Comando `againframe`

```
\begin{frame}
```

Aquí se proporciona información más detallada sobre el punto 2.

```
\end{frame}
```

```
\againframe<3>{volver}
```

Aquí se proporciona información más detallada sobre el punto 2.

- El comando `againframe` se pone **fuera** del frame, en el lugar de la próxima diapositiva.
- Se indica el número de slide a partir del cual se quiere **reanudar**...
- ... la diapositiva con la etiqueta (`label`) especificada



# Reanudar diapositivas inacabadas

Comando `againframe`

```
\begin{frame}
```

Aquí se proporciona información más detallada sobre el punto 2.

```
\end{frame}
```

```
\againframe<3>{volver}
```

Aquí se proporciona información más detallada sobre el punto 2.

- El comando `againframe` se pone **fuera** del frame, en el lugar de la próxima diapositiva.
- Se indica el número de slide a partir del cual se quiere **reanudar**...
- ... la diapositiva con la etiqueta (**label**) especificada



# Reanudar diapositivas inacabadas

Comando `againframe`

A veces puede interesar dejar parte del contenido de un frame para ser retomado más adelante (o no).

```
\begin{frame}<1-2>[label=volver]
\begin{enumerate}<+- | alert@+>
  \item Restringir el rango...
  \item Etiquetar el frame
  \item Contenido adicional...
\end{enumerate}
\end{frame}
```

- 1 Restringir el rango de slides que se visualizarán en primera instancia.
- 2 Etiquetar el frame.
- 3 **Contenido adicional, para más tarde.**



# Ejercicio 4

## Botones y enlaces

- 1 Abrir y compilar el ejercicio 4. Consiste en un enunciado y su demostración, y otra diapositiva que representa el resto de la presentación.
- 2 Hacer que la demostración sólo sea visible después de acabada la presentación, previendo el caso de que el público requiera más detalles, pero omitiéndola en una primera instancia (comando `againframe`)
- 3 En la diapositiva 1, añadir un botón que permita enseñar la demostración, en caso de que alguien reclame detalles en ese momento.
- 4 Este botón debe desaparecer en cuanto se muestren los detalles.
- 5 Añadir otro botón para volver atrás y continuar con la presentación.
- 6 Evitar que el enunciado del Teorema «salte»(si lo hace) cuando se presentan y se quitan los detalles.

Una solución se encuentra en el [ejemplo 5](#).



## Sección 5 | Efectos y animaciones



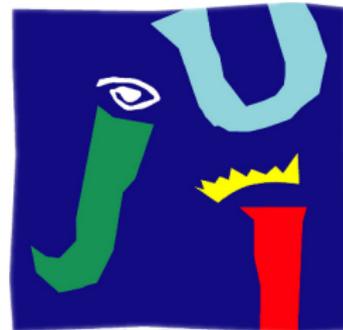


# Zooming

Las figuras (o cualquier cosa) pueden **ampliarse** utilizando **framezoom**.

```
\begin{frame}
  \framezoom<0><2>[border] (9cm,0cm) (2cm,1.5cm)
  \framezoom<1><0>[border] (8cm,.6cm) (1.3cm,3cm)
  \framezoom<4><3>(7cm,3.5cm) (5cm,2cm)

  \pgfimage[width=\textwidth]{img/logoUJI}
% o también:
%\includegraphics[width=\textwidth]{img/logoUJI}
\end{frame}
```



UNIVERSITAT  
JAUME · I



# framezoom.





# UNIVERSITAT JAUME I



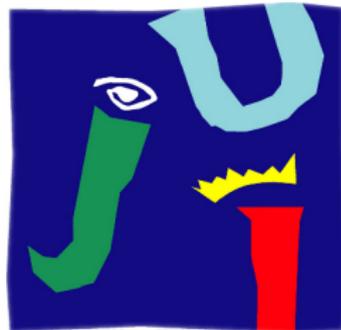


# Zooming

Las figuras (o cualquier cosa) pueden **ampliarse** utilizando **framezoom**.

```
\begin{frame}
  \framezoom<0><2>[border] (9cm,0cm) (2cm,1.5cm)
  \framezoom<1><0>[border] (8cm,.6cm) (1.3cm,3cm)
  \framezoom<4><3>(7cm,3.5cm) (5cm,2cm)

  \pgfimage[width=\textwidth]{img/logoUJI}
% o también:
%\includegraphics[width=\textwidth]{img/logoUJI}
\end{frame}
```



UNIVERSITAT  
JAUME·I

La primera especificación de overlay indica en qué slide se activan los **botones** para hacer click.

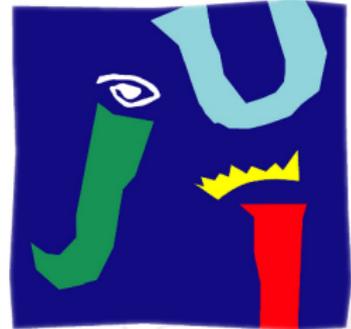


# Zooming

Las figuras (o cualquier cosa) pueden **ampliarse** utilizando **framezoom**.

```
\begin{frame}
  \framezoom<0><2>[border](9cm,0cm)(2cm,1.5cm)
  \framezoom<1><0>[border](8cm,.6cm)(1.3cm,3cm)
  \framezoom<4><3>(7cm,3.5cm)(5cm,2cm)

  \pgfimage[width=\textwidth]{img/logoUJI}
% o también:
%\includegraphics[width=\textwidth]{img/logoUJI}
\end{frame}
```



UNIVERSITAT  
JAUME · I

La segunda, en qué slide **se muestra** la ampliación correspondiente.

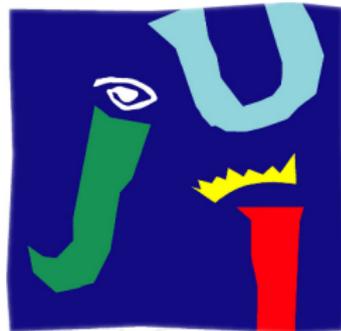


# Zooming

Las figuras (o cualquier cosa) pueden **ampliarse** utilizando **framezoom**.

```
\begin{frame}
  \framezoom<0><2>[border](9cm,0cm)(2cm,1.5cm)
  \framezoom<1><0>[border](8cm,.6cm)(1.3cm,3cm)
  \framezoom<4><3>(7cm,3.5cm)(5cm,2cm)

  \pgfimage[width=\textwidth]{img/logoUJI}
% o también:
%\includegraphics[width=\textwidth]{img/logoUJI}
\end{frame}
```



UNIVERSITAT  
JAUME·I

El botón es visible sólo si se utiliza la opción **border**.



# Zooming

Las figuras (o cualquier cosa) pueden **ampliarse** utilizando **framezoom**.

```
\begin{frame}
  \framezoom<0><2>[border] (9cm, 0cm) (2cm, 1.5cm)
  \framezoom<1><0>[border] (8cm, .6cm) (1.3cm, 3cm)
  \framezoom<4><3>(7cm, 3.5cm) (5cm, 2cm)

  \pgfimage[width=\textwidth]{img/logoUJI}
% o también:
%\includegraphics[width=\textwidth]{img/logoUJI}
\end{frame}
```



UNIVERSITAT  
JAUME · I

Coordenadas de la esquina superior izquierda, y tamaño de la ampliación.



# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.





# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.

```
\begin{frame}  
  \transdissolve  
  \transblindshorizontal<3-4>  
  \transwipe[duration=5]<5-6>  
\end{frame}
```

- Se especifican con un comando que afecta a todo el frame...
- ... o a slides específicas.
- Opcionalmente se puede determinar la duración, en segundos, del efecto.



# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.





# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.

```
\begin{frame}  
  \transdissolve  
  \transblindshorizontal<3-4>  
  \transwipe[duration=5]<5-6>  
\end{frame}
```

- Se especifican con un comando que afecta a todo el frame...
- ... o a slides específicas.
- Opcionalmente se puede determinar la duración, en segundos, del efecto.



# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.





# Transiciones entre diapositivas

- El formato PDF permite definir efectos de **transición** entre diapositivas.
- **Cuidado!** Los distintos visualizadores de PDF tienen **diferentes interpretaciones y niveles de soporte** de estos efectos.

```
\begin{frame}  
  \transdissolve  
  \transblindshorizontal<3-4>  
  \transwipe[duration=5]<5-6>  
\end{frame}
```

- Se especifican con un comando que afecta a todo el frame...
- ... o a slides específicas.
- **Opcionalmente se puede determinar la duración, en segundos, del efecto.**



Para hacer animaciones con BEAMER hay varios métodos, pero ninguno de ellos del todo satisfactorio

## ■ Insertar vídeo

Paquete `multimedia` / comando `movie`

- No es general. Sólo última versión del Acrobat Reader.
- (contra)Ejemplo 6: [ejemplos/6-video.pdf](#)

Paquete `movie15` / comando `includemovie`

## ■ Animar texto

Comandos `animate` / `animatevalue`

## ■ Animar gráficos

Paquete `xmpmulti` / comando `multiinclude`

Paquete `animate` / comando `animategraphics`

Más info.: BEAMER User's guide [Tantau, 2010, s.14].



Para hacer animaciones con BEAMER hay varios métodos, pero ninguno de ellos del todo satisfactorio

## ■ Insertar vídeo

Paquete `multimedia` / comando `movie`

Paquete `movie15` / comando `includemovie`

- Más general: utiliza un reproductor externo.
- El vídeo se incluye en el PDF. No es necesario pasar el archivo aparte.
- Ejemplo 7: [ejemplos/7-movie.pdf](#)

## ■ Animar texto

Comandos `animate` / `animatevalue`

## ■ Animar gráficos

Paquete `xmpmulti` / comando `multiinclude`

Paquete `animate` / comando `animategraphics`

Más info.: BEAMER User's guide [Tantau, 2010, s.14].



Para hacer animaciones con BEAMER hay varios métodos, pero ninguno de ellos del todo satisfactorio

## ■ Insertar vídeo

Paquete `multimedia` / comando `movie`

Paquete `movie15` / comando `includemovie`

## ■ Animar texto

Comandos `animate` / `animatevalue`

- Efecto creado a través de la sucesión rápida de slides.
- No se puede adelantar o retroceder diapositivas naturalmente. Hay que recurrir a botones o los símbolos de navegación de BEAMER.
- La velocidad de la sucesión depende del visualizador.
- Ejemplo 8: [ejemplos/8-animate.pdf](#)

## ■ Animar gráficos

Paquete `xmpmulti` / comando `multiinclude`

Paquete `animate` / comando `animategraphics`

Más info.: BEAMER User's guide [Tantau, 2010, s.14].



Para hacer animaciones con BEAMER hay varios métodos, pero ninguno de ellos del todo satisfactorio

- Insertar vídeo

  - Paquete `multimedia` / comando `movie`

  - Paquete `movie15` / comando `includemovie`

- Animar texto

  - Comandos `animate` / `animatevalue`

- Animar gráficos

  - Paquete `xmpmulti` / comando `multiinclude`

    - No es general. Sólo imagenes creadas con MetaPost.

  - Paquete `animate` / comando `animategraphics`

Más info.: BEAMER User's guide [Tantau, 2010, s.14].



Para hacer animaciones con BEAMER hay varios métodos, pero ninguno de ellos del todo satisfactorio

- Insertar vídeo

Paquete `multimedia` / comando `movie`

Paquete `movie15` / comando `includemovie`

- Animar texto

Comandos `animate` / `animatevalue`

- Animar gráficos

Paquete `xmpmulti` / comando `multiinclude`

Paquete `animate` / comando `animategraphics`

- Sucesión de cualquier conjunto de imágenes.

Más info.: BEAMER User's guide [Tantau, 2010, s.14].



# Colores alternados en tablas

- BEAMER carga automáticamente el paquete **xcolor** [Kern, 2007].
- Esto permite combinar los **nombres** de los colores con porcentajes, utilizando el especificador **!**. Por ejemplo, `\color{BlueViolet!30}`.
- La opción global **table** carga el paquete **colortbl**, que proporciona el comando **rowcolors**. Este comando permite alternar dos colores en una tabla.

```
\documentclass[xcolor=dvipsnames,table]{beamer}
...
\rowcolors{2}{RoyalBlue!5}{RoyalBlue!20}
\begin{tabular}{rll} \hline
Hora & & Actividad \\ \hline
10-14 & & Curso BEAMER \\
14-16 & & Comida \\
16-18 & & Taller \\
\ldots & & \ldots \\ \hline
\end{tabular}
```

Hora	Actividad
10-14	Curso BEAMER
14-16	Comida
16-18	Taller
...	...



# Colores alternados en tablas

- BEAMER carga automáticamente el paquete **xcolor** [Kern, 2007].
- Esto permite combinar los **nombres** de los colores con porcentajes, utilizando el especificador **!**. Por ejemplo, `\color{BlueViolet!30}`.
- La opción global **table** carga el paquete **colortbl**, que proporciona el comando **rowcolors**. Este comando permite alternar dos colores en una tabla.

```
\documentclass[xcolor=dvipsnames,table]{beamer}
...
\rowcolors{2}{RoyalBlue!5}{RoyalBlue!20}
\begin{tabular}{rll} \hline
Hora & & Actividad \\ \hline
10-14 & & Curso BEAMER \\
14-16 & & Comida \\
16-18 & & Taller \\
\ldots & & \ldots \\ \hline
\end{tabular}
```

Hora	Actividad
10-14	Curso BEAMER
14-16	Comida
16-18	Taller
...	...



# Colores alternados en tablas

- BEAMER carga automáticamente el paquete `xcolor` [Kern, 2007].
- Esto permite combinar los **nombres** de los colores con porcentajes, utilizando el especificador `!`. Por ejemplo, `\color{BlueViolet!30}`.
- La opción global **table** carga el paquete `colortbl`, que proporciona el comando `rowcolors`. Este comando permite alternar dos colores en una tabla.

```
\documentclass[xcolor=dvipsnames,table]{beamer}
...
\rowcolors{2}{RoyalBlue!5}{RoyalBlue!20}
\begin{tabular}{rll} \hline
Hora & & Actividad \\ \hline
10-14 & & Curso BEAMER \\ \hline
14-16 & & Comida \\ \hline
16-18 & & Taller \\ \hline
\ldots & & \ldots \\ \hline
\end{tabular}
```

Hora	Actividad
10-14	Curso BEAMER
14-16	Comida
16-18	Taller
...	...



# Colores alternados en tablas

- BEAMER carga automáticamente el paquete **xcolor** [Kern, 2007].
- Esto permite combinar los **nombres** de los colores con porcentajes, utilizando el especificador **!**. Por ejemplo, `\color{BlueViolet!30}`.
- La opción global **table** carga el paquete **colortbl**, que proporciona el comando **rowcolors**. Este comando permite alternar dos colores en una tabla.

```
\documentclass[xcolor=dvipsnames,table]{beamer}
...
\rowcolors{2}{RoyalBlue!5}{RoyalBlue!20}
\begin{tabular}{rll} \hline
Hora & Actividad \\ \hline
10-14 & Curso BEAMER \\
14-16 & Comida \\
16-18 & Taller \\
\ldots & \ldots \\ \hline
\end{tabular}
```

Hora	Actividad
10-14	Curso BEAMER
14-16	Comida
16-18	Taller
...	...



# Comienzos de secciones

Habitualmente interesa utilizar una diapositiva especial que sirva de **introducción** a cada sección. El comando `AtBeginSection{}` permite especificar el contenido.





# Comienzos de secciones

Habitualmente interesa utilizar una diapositiva especial que sirva de **introducción** a cada sección. El comando `AtBeginSection{}` permite especificar el contenido.

```
\AtBeginSection
{
  \begin{frame}<beamer>
    \frametitle{Outline}
    \tableofcontents[currentsection]
  \end{frame}
}
```

Comienza cada sección con una diapositiva mostrando la tabla de contenidos con la sección actual destacada, sólo en modo beamer.

Beamer básico   Temas y personalización   Notas y handouts   Overlays   Beamer modes   Botones y enlaces   Efectos y animaciones	
1	Beamer básico
2	Temas y personalización
3	Notas y handouts
4	Control avanzado de overlays
5	Beamer modes
6	Botones y enlaces
7	Efectos y animaciones

Facundo Muñoz | Curso avanzado de Beamer 8/24

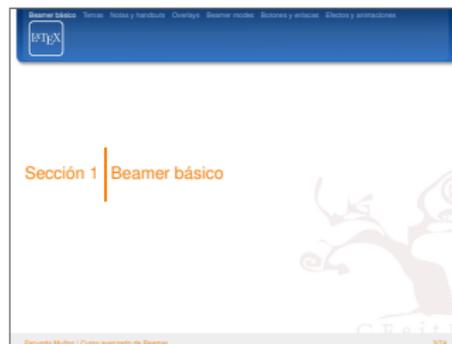


# Comienzos de secciones

Habitualmente interesa utilizar una diapositiva especial que sirva de **introducción** a cada sección. El comando `AtBeginSection{}` permite especificar el contenido.

```
\AtBeginSection
{
  \begin{frame}
    \begin{structureenv}
      \Large Sección \thesection\
      rule[-.6cm]{2pt}{1.5cm}
      \insertsection
    \end{structureenv}
  \end{frame}
}
```

La configuración utilizada en esta presentación.



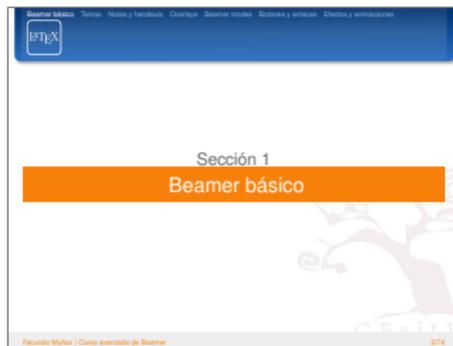


# Comienzos de secciones

Habitualmente interesa utilizar una diapositiva especial que sirva de **introducción** a cada sección. El comando `AtBeginSection{}` permite especificar el contenido.

Una alternativa.

```
\AtBeginSection
{
  \begin{frame}
    \begin{center}
      {\Large Sección $\thesection$}
      \begin{beamercolorbox}[
        sep=8pt,center]{part title}
        \usebeamerfont{part title}
        \insertsection
      \end{beamercolorbox}
    \end{center}
  \end{frame}
}
```





# Imágenes con reflejo

## Ejemplo 9

- El paquete de dibujo **Tikz** [Tantau, 2008], permite hacer cosas verdaderamente sofisticadas, como añadir un efecto reflejo a texto o imágenes.
- Aunque algunas características gráficas, como el **degradado**, pueden no ser soportadas en todos los visualizadores.





# Señalamiento dinámico

## Ejemplo 10

- Este ejemplo destaca paso a paso los términos de una ecuación, y los conecta sucesivamente con la correspondiente descripción.
- Extraído de la extraordinaria galería de ejemplos **Tikz** [Tantau, 2008]:

<http://www.texample.net/>

### Rigid body dynamics

- Coriolis acceleration

$$\vec{a}_p = \vec{a}_o + \frac{b_d^2}{dt^2} \vec{r} + 2\vec{\omega}_{ib} \times \frac{b_d}{dt} \vec{r} + \vec{\alpha}_{ib} \times \vec{r} + \vec{\omega}_{ib} \times (\vec{\omega}_{ib} \times \vec{r})$$

Author: ()
Presentation title
January 27, 2011 1 / 1



# Señalamiento dinámico

## Ejemplo 10

- Este ejemplo destaca paso a paso los términos de una ecuación, y los conecta sucesivamente con la correspondiente descripción.
- Extraído de la extraordinaria galería de ejemplos **Tikz** [Tantau, 2008]:

<http://www.texample.net/>

**Rigid body dynamics**

- Coriolis acceleration

$$\vec{a}_p = \vec{a}_o + \frac{b}{dt^2} \vec{r} + 2\vec{\omega}_{ib} \times \frac{b}{dt} \vec{r} + \vec{\alpha}_{ib} \times \vec{r} + \vec{\omega}_{ib} \times (\vec{\omega}_{ib} \times \vec{r})$$

- Transversal acceleration

◀ ▶ ⏪ ⏩ 🔍  
 Author: ()      Presentation title      January 27, 2011      1 / 1



# Señalamiento dinámico

## Ejemplo 10

- Este ejemplo destaca paso a paso los términos de una ecuación, y los conecta sucesivamente con la correspondiente descripción.
- Extraído de la extraordinaria galería de ejemplos **Tikz** [Tantau, 2008]:

<http://www.texample.net/>

**Rigid body dynamics**

- Coriolis acceleration
- Transversal acceleration
- Centripetal acceleration

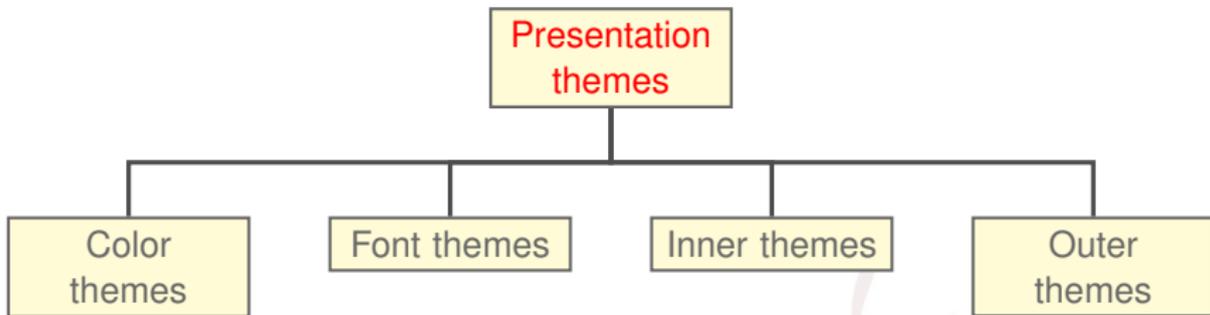
$$\vec{a}_p = \vec{a}_o + \frac{b}{dt^2} \vec{r} + 2\vec{\omega}_{ib} \times \frac{b}{dt} \vec{r} + \vec{\alpha}_{ib} \times \vec{r} + \vec{\omega}_{ib} \times (\vec{\omega}_{ib} \times \vec{r})$$

◀ ▶ ⏪ ⏩ ↺ ↻
Author: ()
Presentation title
January 27, 2011
1 / 1



## Sección 6 | Temas y personalización





- Es posible modificar algunos aspectos concretos del tema seleccionado, especificando un **subtema** con los comandos `\usecolortheme`, `\usefonttheme`, `\useinnertheme` o `\useoutertheme`.



# El colortheme structure

## Ejemplo 11

Personaliza los colores utilizados para los elementos estructurales.

Experimentando con themes

Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}
\usetheme{Madrid}
...
```

(Geotema - UV) Curso avanzado de Beamer Febrero, 2011 1 / 1



# El colortheme structure

## Ejemplo 11

Personaliza los colores utilizados para los elementos estructurales.

Experimentando con themes

Objetivos

1. Modificar un theme especificando subthemes
2. Visualizar el resultado

```
\documentclass[dvipsnames]{beamer}
\usetheme[secheader]{Madrid}
\usecolortheme[
  named=MidnightBlue]{structure}
...
```

(Geotema – UV) Curso avanzado de Beamer Febrero, 2011 1 / 1



# El colortheme structure

## Ejemplo 11

Personaliza los colores utilizados para los elementos estructurales.

Experimentando con themes

Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}
\usetheme[secheader]{Madrid}
\usecolortheme[
  RGB={128,37,92}]{structure}
...
```

(Geotema - UV) Curso avanzado de Beamer Febrero, 2011 1 / 1



# Tuneando los themes

## ■ Inner themes

Especificación del formato de la portada, entornos tipo itemize, bloques, figuras y tablas, footnotes y bibliografía.

- `\useinnertheme{default}`
- `\useinnertheme{circles}`
- `\useinnertheme{rectangles}`
- `\useinnertheme[shadow]{rounded}`

## ■ Outer themes

## ■ Más detalles





# Tuneando los themes

## ■ Inner themes

## ■ Outer themes

Especificación del formato de la cabecera y del pie de página, las barras laterales, el logo, y el título de los frames.

- `\useoutertheme{default}`
- `\useoutertheme{infolines}`
- `\useoutertheme[opciones]{miniframes}`
- `\useoutertheme[opciones]{smoothbars}`
- `\useoutertheme[opciones]{sidebar}`
- `\useoutertheme{split}`
- `\useoutertheme[opciones]{shadow}`
- `\useoutertheme[opciones]{tree}`
- `\useoutertheme[opciones]{smoothtree}`

## ■ Más detalles





# Tuneando los themes

- Inner themes
- Outer themes
- **Más detalles**

BEAMER User's Guide [Tantau, 2010]

Código fuente de los themes ya definidos



## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}  
  \usetheme{Madrid}  
  ...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}  
  \usetheme [secheader] {Madrid}  
  ...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}  
  \usetheme[secheader]{Madrid}  
  \usecolortheme{beaver}  
  ...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}  
  \usetheme[secheader]{Madrid}  
  \usecolortheme{lily}% reset inner  
  \usecolortheme{beaver}  
  ...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}
\usetheme[secheader]{Madrid}
\usecolortheme{lily}% reset inner
\usecolortheme{beaver}
\useinnertheme{rectangles}
...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}
\usetheme[secheader]{Madrid}
\usecolortheme{lily}% reset inner
\usecolortheme{beaver}
\useinnertheme{rectangles}
\useoutertheme{shadow}
...
```

# Experimentando con themes

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}
\usecolortheme{beaver}
\useinnertheme{rectangles}
\useoutertheme{shadow}
...
```

## Objetivos

- 1 Modificar un theme especificando subthemes
- 2 Visualizar el resultado

```
\documentclass{beamer}  
  \usetheme{Madrid}  
  ...
```



- Otra alternativa es utilizar un tema externo, diseñado por otro y que sea de libre utilización
- Este curso incluye dos temas externos dentro de la carpeta **themes**
  - TUGraz Style  
Desarrollado por Martin Weiglhofer, basado en el estilo corporativo de la Graz University of Technology (Austria).  
<http://www.ist.tugraz.at/staff/weiglhofer/misc/tugbeamer/>
  - Oxygen Style  
Desarrollo corporativo de KDE.  
<http://www.kde.org/kdeslides/>
- El tema utilizado en este curso es una modificación de otro, desarrollado por Antonio Gabriel López, Sergio Alonso y Carlos Porcel [1], de la Universidad de Granada, que a su vez está basado en el Oxygen Style.



Dos alternativas:

- 1 Incluir los archivos de estilos en el mismo directorio que el fuente
- 2 Copiar los estilos en el directorio correspondiente de la instalación de BEAMER<sup>7</sup> y actualizar la base de datos de LATEX (mktexlsr).

---

<sup>7</sup>Por ejemplo, en Linux, `/usr/share/texmf/tex/latex/beamer/themes/`, y en Windows

`C:/Archivos de Programa/MikTeX 2.9/tex/latex/beamer/base/themes/`



# Mayor personalización: **templates**

Los templates definen el aspecto de cada elemento de una presentación

## Fondo

```
\setbeamertemplate{background}
{
  \parbox[c][\paperheight]{\paperwidth}
  {
    \vfill \hfill
    \begin{tikzpicture}
      \node[opacity=.07]
      {
        \includegraphics[height=5cm]{img/LogoGeeitema.jpg}
      };
    \end{tikzpicture}
    \vspace{.5cm} \hspace{-3cm}
  }
}
```



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}

```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]

```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de **templates**

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}

```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]

```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de **templates**

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- \setbeamertemplate{items}[circle]
- \setbeamertemplate{items}[ball]
- \setbeamertemplate{items}[square]

El comando **setbeamertemplate** se pone en el preámbulo o dentro del documento (pero fuera de los frames).



## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`



# Opciones de **templates**

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de `templates`

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando `setbeamertemplate` se pone en el preámbulo o dentro del documento (pero fuera de los frames).



# Opciones de **templates**

## Símbolos de navegación

```
\setbeamertemplate{navigation symbols}[default]
\setbeamertemplate{navigation symbols}[vertical]
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{navigation symbols}{}
```

## Bloques

```
\setbeamertemplate{blocks}[default]
\setbeamertemplate{blocks}[rounded]
\setbeamertemplate{blocks}[rounded][shadow=true]
```

## Listas

- `\setbeamertemplate{items}[circle]`
- `\setbeamertemplate{items}[ball]`
- `\setbeamertemplate{items}[square]`

El comando **setbeamertemplate** se pone en el preámbulo o dentro del documento (pero fuera de los frames).



## Ejercicio 5

- 1 Compilar el ejercicio 5. Notar que tiene el contenido de la diapositiva anterior.
- 2 Reproducir el comportamiento dinámico, usando `againframe`.





## Ejercicio 6

En este ejercicio aplicaremos los **temas externos** a las diapositivas que tenemos del ejercicio 1.

- 1 Copiar los archivos de **estilo** (.sty) en el directorio de ejercicios
- 2 Sustituir el nombre del tema en el comando **usetheme** sucesivamente por **Graz** y por **Oxygen**
- 3 Renombrar la versión modificada del **beamerouterthemetugraz.sty**, y sustituir la original. Recompilar para ver las diferencias. Buscar la modificación en el código fuente.



## Sección 7 | Taller de fin de curso





# Ejercicio final

Acabaremos el curso con un ejercicio que nos dejará un material de uso práctico para nuestras presentaciones.

Nos fabricaremos un **template** que podremos utilizar cada vez que necesitemos hacer una presentación.

En él dejaremos abiertas las opciones que nos pueden interesar y le daremos un estilo personalizado.

- 1 Hacer una copia del código fuente de esta presentación y vaciarla de contenido.
- 2 Comentar las opciones del preámbulo que nos pueda interesar utilizar
- 3 Utilizar y/o modificar un tema que nos guste.



## Sección 8 | Material de referencia





El material de este curso se ha compilado en base a los siguientes manuales y tutoriales



### Till Tantau.

User's guide to the BEAMER class, version 3.01, July 2010.  



### Charles T. Batts.

A Beamer Tutorial in Beamer, 2007.

Department of Computer Science, University of North Carolina at Greensboro.  



### Ki-Joo Kim.

Beamer v3.0 Guide, November 2004.  



Antonio G. López, Sergio Alonso, and Carlos Porcel.

Edición Avanzada de Textos Científicos con LaTeX y Gestión Bibliográfica, 2009.  



Andrew Mertz and William Slough.

Beamer by example.

In Barbara Beeton and Karl Berry, editors, *Practical TeX 2005*, volume 26, pages 68–73. T<sub>E</sub>X Users Group, June 2005.  



Walter Mora Flores.

Cómo hacer transparencias con la clase Beamer de LaTeX, 2007.

Instituto Tecnológico de Costa Rica, Escuela de Matemática.  



Rouben Rostamian.

A Beamer Quickstart, January 2011. 



Además, he hecho referencia a los siguientes paquetes de L<sup>A</sup>T<sub>E</sub>X



**Uwe Kern.**

*Extending L<sup>A</sup>T<sub>E</sub>X's color facilities: The xcolor package*, January 2007.



**Andreas Matthias.**

*The pdfpages Package*, December 2010.  



**Till Tantau.**

*The TikZ and PGF Packages. Manual for version 2.00.*

Institut für Theoretische Informatik, Universität zu Lübeck, February 2008.  



## Usted es libre de:



copiar, distribuir y comunicar públicamente la obra



hacer obras derivadas



## Bajo las condiciones siguientes:



**Reconocimiento.** Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).



**Compartir bajo la misma licencia.** Si transforma o modifica esta obra para crear una obra derivada, sólo puede distribuir la obra resultante bajo la misma licencia, una similar o una compatible.