

# Apuntes de L<sup>A</sup>T<sub>E</sub>X

## Capítulo 10: Documentos navegables y presentaciones

### 1. El paquete `hyperref`

Hacer un documento navegable es tan sencillo como cargar el paquete `hyperref`, automáticamente las referencias en el texto y la tabla de contenidos se convierten en “links” que nos llevan hasta la referencia o la sección correspondiente. Puesto que el paquete `hyperref` redefine muchos comandos L<sup>A</sup>T<sub>E</sub>X es conveniente que sea el **último** paquete en cargarse.

Para cargarlo, se utiliza la sintaxis habitual:

```
\usepackage[opciones]{hyperref}
```

Para especificar opciones, alternativamente se puede utilizar el comando:

```
\hypersetup{opcion1=valor1,opcion2=valor2,etc...}
```

Las opciones básicas de controlador disponibles son las siguientes:<sup>1</sup>

---

**hypertex** DVI processors conforming to the HyperT<sub>E</sub>X guidelines (i.e. `xdvi`, `dvips` (with the `-z` option), `OzTEX`, and `Textures`)

**dvips** produces `\special` commands tailored for `dvips`

**dvipsone** produces `\special` commands tailored for `dvipsone`

**ps2pdf** a special case of output suitable for processing by earlier versions of Ghostscript’s PDF writer; this is basically the same as that for `dvips`, but a few variations remained before version 5.21

**tex4ht** produces `\special` commands for use with T<sub>E</sub>X4ht

---

<sup>1</sup>Tomado de la documentación del paquete en `/texmf/doc/latex/hyperref/manual`

**pdftex** pdf<sub>T</sub>E<sub>X</sub>, T<sub>E</sub>X variant that writes PDF directly

**dvipdf** produces `\special` commands for the DVI to PDF driver **dvipdf**

**dvipdfm** produces `\special` commands for Mark Wicks' DVI to PDF driver **dvipdfm**

**dviwindo** produces `\special` commands that Y&Y's Windows previewer interprets as hypertext jumps within the previewer

**vtex** produces `\special` commands that MicroPress' HTML and PDF-producing T<sub>E</sub>X variants interpret as hypertext jumps within the previewer

**textures** produces `\special` commands that Textures interprets as hypertext jumps within the previewer

---

Uno de los comandos básicos para crear enlaces de hipertexto es:

`\href{Tipo:Dirección}{Objeto}`

Que asigna un hiperenlace a *Objeto*, redirigiéndolo a *Dirección*. Podemos elegir entre los siguientes tipos:

**http** Tipo por defecto, que no necesita ser indicado. Tampoco es necesario incluir los dos puntos ó doble barra que suelen aparecer en las direcciones de internet. Por ejemplo:

`\href{www.uva.es}{Universidad de Valladolid}`

**ftp** Para conectar con servidores ftp:

`\href{ftp://ftp.rediris.es/pub/}{Servidor Rediris}`

**mailto** Para enviar correos electrónicos:

`\href{mailto:lmolina@fta.uva.es}{Enviar correo al profesor}`

**file** Abre un archivo con el programa asociado:

`\href{file:c:/camino/archivo}{Nombre del archivo}`

**run** Permite ejecutar aplicaciones:

`\href{run:c:/windows/notepad.exe}{Abrir el bloc de notas}`

Aparte de los hiperenlaces creados automáticamente por **hyperref** a partir de los enlaces básicos de L<sup>A</sup>T<sub>E</sub>X, se pueden incluir hiperenlaces con la pareja de comandos:

`\hypertarget{etiqueta}{texto de llegada}`

`\hyperlink{etiqueta}{texto del enlace}`

Para enlazar con otra página del documento se puede usar:

`\hyperlink{page.N}{Número de página}`

que enlazaría con la página N.

Se pueden colorear los enlaces con la opción:

`colorlinks=true`

y cambiar los colores por defecto (entre paréntesis) con las siguientes opciones:

`linkcolor=color` (red) links internos

`citecolor=color` (green) color de citaciones

`filecolor=color` (magenta) ficheros locales

`pagecolor=color` (red) otras páginas

`urlcolor=color` (cyan) color de direcciones de internet url externas

Otras opciones:

`backref=true` para añadir un link de retorno en la bibliografía .

`pagebackref=true` lo mismo que el anterior pero ligado a la página

## 1.1. Visualización de PDF con Acrobat

El comportamiento de un fichero PDF, al abrirlo con Acrobat, se pueden controlar; aquí solamente veremos algunas opciones (más información en el manual de `hyperref`):

**bookmarks (true)** Se muestran marcadores (opción por defecto)

**bookmarkopen (false)** Si se activa, se muestran los marcadores completamente expandidos

**pdfpagemode** Determina el modo en el que se abrirá el archivo con Acrobat:

- None
- UseThumbs
- UseOutlines (defecto)
- FullScreen

Se puede acceder a los menús de Acrobat mediante enlaces dentro del texto, con el comando:

`\Acrobatmenu{Menuitem}{Texto}`

La siguiente tabla detalla las diferentes posibilidades para `Menuitem`:

File	Open, Close, Scan, Save, SaveAs, Optimizer:SaveAsOpt, Print, PageSetup, Quit
File→Import	ImportImage, ImportNotes, AcroForm:ImportFDF
File→Export	ExportNotes, AcroForm:ExportFDF
File→DocumentInfo	GeneralInfo, OpenInfo, FontsInfo, SecurityInfo, Weblink:Base, AutoIndex:DocInfo
File→Preferences	GeneralPrefs, NotePrefs, FullScreenPrefs, Weblink:Prefs, AcroSearch:Preferences(Windows) or, AcroSearch:Prefs(Mac), Cpt:Capture
Edit	Undo, Cut, Copy, Paste, Clear, SelectAll, Ole:CopyFile, TouchUp:TextAttributes, TouchUp:FitTextToSelection, TouchUp:ShowLineMarkers, TouchUp:ShowCaptureSuspects, TouchUp:FindSuspect, Properties
Edit→Fields	AcroForm:Duplicate, AcroForm:TabOrder
Document	Cpt:CapturePages, AcroForm:Actions, CropPages, RotatePages, InsertPages, ExtractPages, ReplacePages, DeletePages, NewBookmark, SetBookmarkDest, CreateAllThumbs, DeleteAllThumbs
View	ActualSize, FitVisible, FitWidth, FitPage, ZoomTo, FullScreen, FirstPage, PrevPage, NextPage, LastPage, GoToPage, GoBack, GoForward, SinglePage, OneColumn, TwoColumns, ArticleThreads, PageOnly, ShowBookmarks, ShowThumbs
Tools	Hand, ZoomIn, ZoomOut, SelectText, SelectGraphics, Note, Link, Thread, AcroForm:Tool, Acro_Movie:MoviePlayer, TouchUp:TextTool, Find, FindAgain, FindNextNote, CreateNotesFile
Tools→Search	AcroSrch:Query, AcroSrch:Indexes, AcroSrch:Results, AcroSrch:Assist, AcroSrch:PrevDoc, AcroSrch:PrevHit, AcroSrch:NextHit, AcroSrch:NextDoc
Window	ShowHideToolBar, ShowHideMenuBar, ShowHideClipboard, Cascade, TileHorizontal, TileVertical, CloseAll
Help	HelpUserGuide, HelpTutorial, HelpExchange, HelpScan, HelpCapture, HelpPDFWriter, HelpDistiller, HelpSearch, HelpCatalog, HelpReader, Weblink:Home
Help(Windows)	About

## 2. Presentaciones con prosper

Cargando la clase de documento prosper podemos crear presentaciones electrónicas de alta calidad, en formato PDF. Al utilizar esta clase se cargan automáticamente los paquetes `seminar`, `PSTricks`, `graphicx`, `hyperref`, `amssymb` y `times` (por tanto, no es necesario cargarlos explícitamente con `\usepackage`). Es importante tener en cuenta que los tamaños de página, tipo de letra, etc..., son automáticamente redefinidos por **prosper** a estándares apropiados a una presentación.

La estructura básica de un documento prosper se muestra en la siguiente figura:

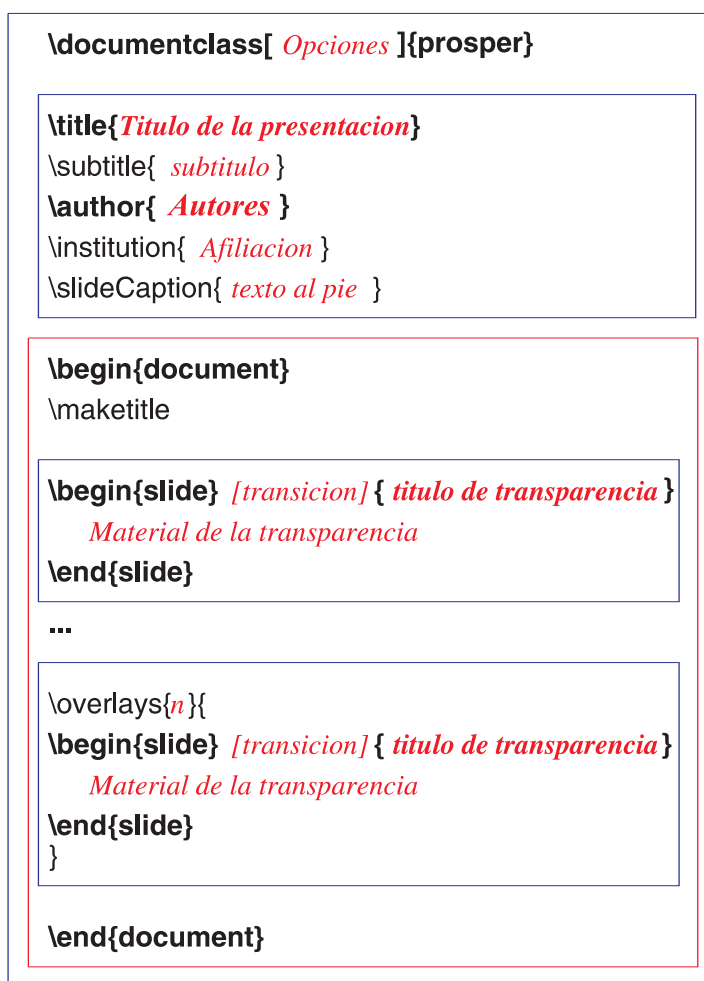


Figura 1: Estructura de un archivo  $\text{\LaTeX}$  usando prosper

El cuerpo del documento contendrá tantos entornos `slide` como transparencias tenga la presentación. Los comandos `\title{...}`, `\subtitle{...}`, `\author{...}` e `\institution{...}` se utilizan para construir la primera transparencia (con títulos). El comando `\slideCaption{...}` se utiliza para colocar una leyenda al pie de todas

las transparencias. Todos estos comandos deben ser incluidos en el *preámbulo* del documento.

## 2.1. Estilo de la presentación

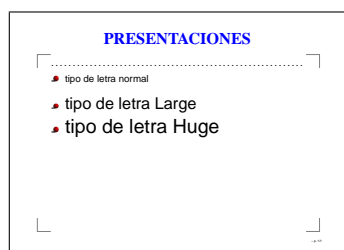
Podemos elegir entre diversos estilos predefinidos para la presentación, incluyendo en nombre de estilo (gyom, azure, etc...) en el argumento optativo [Opciones] del comando `\documentclass`. Se dispone de estas opciones principales:

- default                      • autumn                      • azure                      • contemporain                      • darkblue
- alienglow                      • nuancegris                      • troisponts                      • lignesbleues                      • frames

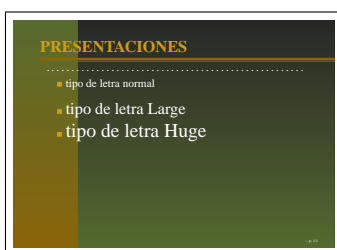
Tomando el siguiente documento de ejemplo:

```
\documentclass[pdf,colorBG,slideColor,xxxxxx]{prosper}
\begin{document}
\begin{slide}{PRESENTACIONES}
\dotfill
\begin{itemize}
\item tipo de letra normal
\item \Large tipo de letra Large
\item \Huge tipo de letra Huge
\end{itemize}
\end{slide}
\end{document}
```

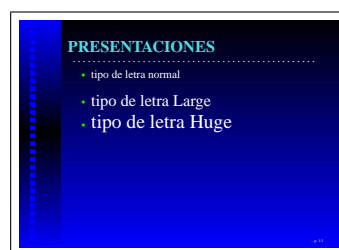
las figuras siguientes ilustran el resultado para cada una de los estilos (se sustituye “xxxxxx” por el nombre de estilo en particular):



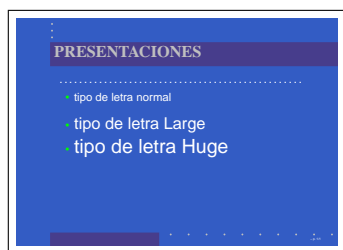
default



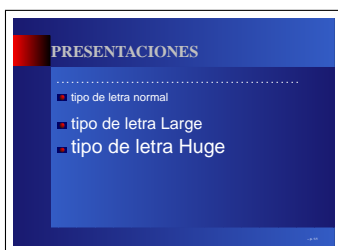
autumn



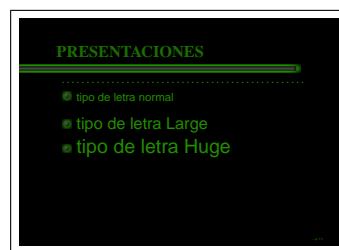
azure



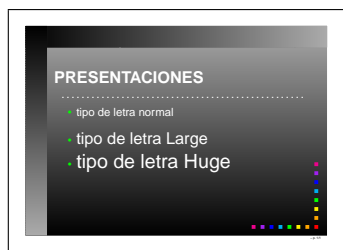
contemporain



darkblue



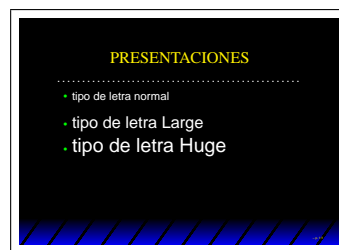
alienglow



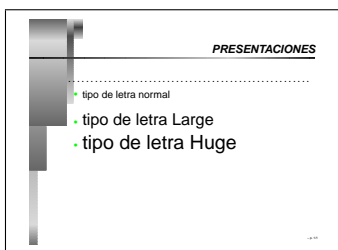
nuancegris



troispoints



lignesbleues



frames

Puede verse cómo el estilo determina las dimensiones del texto, la posición del título, el gráfico de fondo y los colores de las letras. Aparte de estos estilos estándar, existen otros, contribuciones adicionales de diferentes autores: gyom, rico, blends, capsules, corners, fyma, mancini, prettybox, serpaggi, thomasd, whitecross, winter y wj.

## 2.2. Opciones de la clase **prosper**

Además del estilo, se tienen las siguientes opciones generales de la clase **prosper** (en negrita, las opciones por defecto):

**ps** | pdf La opción **ps** se utiliza para producir un fichero PostScript con la presentación, mientras que la opción **pdf** está adaptada al formato electrónico PDF .

**draft** | **final** Esta opción tiene características similares a la correspondiente al paquete **graphicx** (no se incluyen los gráficos con la opción **draft**). Además, con una **u** otra opción se cambia la leyenda al pie de cada página.

**total** | **nototal** Con **total** aparece en la leyenda el número de página actual y el número total de páginas, mientras que con **nototal** sólo aparece el número de página en curso.

**slideBW** | **slideColor** La opción **slideBW** restringe el uso de colores, y está pensada para imprimir transparencias en blanco y negro. **slideColor** permite utilizar todos los colores.

**colorBG** | **nocolorBG** Con la opción **nocolorBG** el color de fondo de las páginas es siempre blanco, independientemente del estilo elegido.

**noaccumulate** | **accumulate** Con **accumulate** se agrupan todas las animaciones en la misma página (útil para una salida impresa), mientras que con **noaccumulate** se permiten las animaciones.

El proceso de compilación de una presentación consiste en:

1. Compilar con  $\text{\LaTeX}$  el fichero fuente
2. Producir el fichero postscript con **dvips**
3. Convertir a formato PDF con **ps2pdf**

y debe tenerse en cuenta que no es posible utilizar  $\text{\PDF\LaTeX}$  para directamente obtener el archivo PDF, ya que prosper carga (y utiliza) recursos del paquete PSTricks.

## 2.3. Preámbulo del documento prosper; transiciones

En el preámbulo del documento, además de los comandos ya vistos para declarar el título de la presentación, autores, etc..., se pueden utilizar otros comandos:

- **\email{E-mails}** Para declarar direcciones de correo electrónico
- **\Logo{Objeto}** Coloca **Objeto** (texto, un gráfico, ...) en *todas* las transparencias, en la posición predeterminada por el estilo
- **\Logo(x,y){Objeto}** Como el anterior, pero permite colocar **Objeto** en la posición (x,y) que elijamos
- **\displayVersion** Hace que, con la opción final, aparezca en las páginas la misma leyenda que con draft, es decir, el título de la presentación, el autor, y la fecha
- **\DefaultTransition{Transición}** Cambia el efecto de transición por defecto entre transparencias (que se pueden también declarar de forma específica para transparencias individuales en el argumento optativo del entorno **slide**). Se puede elegir entre:
  - Blinds      - Box      - Dissolve      - Glitter      - Replace      - Split      - Wipe



## 2.4. Cambiando especificaciones

Si no nos gusta alguna de las especificaciones por defecto introducidas por el estilo, pueden cambiarse:

- `\FontTitle{FuenteColor}{FuenteBW}` Cambia el tipo de letra para el título de cada transparencia; hay que darle dos argumentos, utilizándose uno u otro dependiendo de que opción elijamos para la clase, `slideColor` ó `slideBW`
- `\FontText{FuenteColor}{FuenteBW}` Lo mismo que lo anterior, pero para el tipo de letra del contenido de las transparencias
- `\ColorFoot{Color}` Cambia el color de las leyendas a pie de página
- `\myitem{Nivel}{Definición}` Cambia las marcas por defecto para los distintos subniveles (`Nivel` puede tomar los valores 1, 2, 3) del entorno `itemize`. En `Definición` debemos especificar qué objeto utilizamos para la marca. Por ejemplo:

```
\myitem{1}{\includegraphics[width=.4cm]{red-bullet-on-blue.ps}}
\myitem{2}{\includegraphics[width=.3cm]{green-bullet-on-blue.ps}}
\myitem{3}{\includegraphics[width=.3cm]{yellow-bullet-on-blue.ps}}
```

(los archivos `red-bullet-on-blue.ps`, etc... se encuentran en el subdirectorio `img/` de la distribución de `prosper`, instalada en `c:/texmf/tex/latex/prosper`; existen otros: `red-bullet-on-white.ps`, etc...): Nótese que otros estilos pueden emplear especificaciones diferentes.

- `\def\colA{color}` `\def\colA{color}` Cambia el tipo de color para el fondo, que será una mezcla uniforme entre `ColA` y `ColB`.
- `\global\let\colC=\color` Cambia el color de los títulos de transparencia a `color`
- `\global\let\colD=\color` Cambia el color del texto a `color`

## 2.5. Animaciones

Para incluir efectos de animación, el comando básico es:

```
\overlays{Número}{\begin{slide} ... \end{slide}}
```

lo cual indica que la transparencia incluida en el entorno `slide` va a ser mostrada en el número de etapas `Número`. Es importante que no haya espacios ni saltos de línea entre `\overlays` y `\begin{slide}`

Empecemos por lo más simple; el entorno `itemstep` permite construir listas en las que los objetos van apareciendo en pantalla de forma incremental. Tiene la misma sintaxis que la del entorno `item`:

```

\begin{itemstep}
\item Primer objeto
\item Segundo objeto
...
\end{itemstep}

```

Dentro del entorno `slide`, una vez declarado el número total de etapas con `\overlays`, se pueden utilizar los siguientes comandos para presentar objetos por etapas:

<code>\fromSlide{p}{Objeto}</code>	<code>\fromSlide*{p}{Objeto}</code>	<code>\FromSlide{p}</code>
<code>\onlySlide{p}{Objeto}</code>	<code>\onlySlide*{p}{Objeto}</code>	<code>\OnlySlide{p}</code>
<code>\untilSlide{p}{Objeto}</code>	<code>\untilSlide*{p}{Objeto}</code>	<code>\UntilSlide{p}</code>

El comando `\fromSlide` escribe el argumento `Objeto` desde la etapa `p` hasta el final de la animación; el comando `\FromSlide` escribe *todo lo que sigue* desde la etapa `p` hasta el final de la animación. `\onlySlide` escribe el argumento *solamente en la etapa p*, mientras que `\OnlySlide` escribe todo lo que sigue *solamente en la etapa p*. Finalmente, `\untilSlide` escribe el argumento *desde el principio hasta la etapa p*, mientras que la versión `\UntilSlide` escribe todo lo que sigue *desde el principio hasta la etapa p*.

Las versiones con asterisco implican que lo que se escribe se coloca en una caja de espacio 0-dimensional (ésto es, *el cursor no se mueve*). Estas versiones deben emplearse, por tanto, para reemplazar objetos. Por ejemplo:

```

\onlySlide*{1}{\textcolor{blue}{Texto 1}}
\onlySlide*{2}{\textcolor{red}{Texto 2}}
\onlySlide*{3}{\textcolor{green}{Texto 3}}

```

reemplazaría sucesivamente “`Texto 1`” por “`Texto 2`” y por “`Texto 3`”

Veamos algunos ejemplos; copiar el siguiente documento fuente a un nuevo archivo y compilarlo:

---

```

\documentclass[pdf,colorBG,slideColor,default]{prosper}

\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[spanish]{babel}
\usepackage{pstricks,pst-node}
\usepackage[ps2pdf]{hyperref}

\begin{document}

```

```

\overlays{7}{%
\begin{slide}{Ejemplo}
\fromSlide*{2}{Objeto A}
\fromSlide*{2}{\rnode{N1}{}}}%

\fromSlide*{3}{Texto}

\fromSlide*{4}{Conectamos Objeto A con una X ... Aqui ...}
\fromSlide*{5}{(\rnode{N2}{X})}
\fromSlide*{5}{\ncurve[linecolor=blue,linewidth=4pt,
angleA=0,angleB=120]{->}{N1}{N2}}

\bigskip
\onlySlide*{6}{\textcolor{blue}{ponemos un texto en azul}}
\fromSlide*{7}{\textcolor{red}{y ahora lo cambiamos por otro rojo}}
\end{slide}}

\end{document}

```

---

ahora jugamos un poco con cambios de color; con el mismo preámbulo, compilar:

---

```

\begin{document}

\overlays{7}{%
\begin{slide}[Replace]{Ejemplo; cambio de color}

\bigskip
\fromSlide*{1}{Cambiamos el color de la siguiente \color{blue}
ecuación:}
\onlySlide*{2}{\color{blue}  $\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + \int g(x) dx$ }
\onlySlide*{3}{ $\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + \int g(x) dx$ }
\onlySlide*{4}{ $\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + \int g(x) dx$ }
\onlySlide*{5}{ $\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + \int g(x) dx$ }
\onlySlide*{3}{\color{red} rate of change of  $F$  w.r.t time}
\onlySlide*{4}{\hspace*{3.0cm}\color{red} diffusion of  $F$ }
\onlySlide*{5}{\hspace*{5cm}\color{red} integral of  $g(x)$ }

```

```
\fromSlide*{6}{
$$\frac{\partial F}{\partial t} = \frac{\partial^2 F}{\partial x^2} + \int g(x) dx$$
}
```

```
\bigskip
```

```
\fromSlide{6}{\color{black} \hspace{1cm} Sigue aqui mas texto...}
```

```
\bigskip
```

```
\fromSlide{7}{\color{black} \hspace{1cm} y aqui mas texto aun}
```

```
\end{slide}}
```

```
\end{document}
```

---