

# Exposés sur L<sup>A</sup>T<sub>E</sub>X

## Cours 9 La classe beamer

Thierry Masson

Centre de Physique Théorique  
Campus de Luminy, Marseille



## Cours 9 – La classe beamer

- Un tour d'horizon des possibles
- Les commandes de base
- Modifier l'aspect de sa présentation
- Au delà de la présentation

“If your words or images are not on point,  
making them dance in color won't make them relevant.  
Audience boredom is usually a content failure,  
not a decoration failure.”

Edward R. Tufte

*PowerPoint Is Evil* ([www.wired.com](http://www.wired.com))

# Un tour d'horizon des possibles

Où l'on pèse d'abord les avantages et les inconvénients de préparer un exposé avec son ordinateur, et où l'on se demande alors quel logiciel utiliser...

# Une question fondamentale

## Pourquoi utiliser un ordinateur pour une présentation ?

- Parce qu'on n'a pas le choix : absence de tableau, absence de rétroprojecteur.
- Parce qu'on souhaite recycler :
  - soit la présentation elle-même, sous différentes formes modifiées ;
  - soit des morceaux d'une publication (formules, dessins, tableaux...).
- ➔ Requiert des logiciels compatibles pour les articles et les présentations...
- Parce qu'on a du matériel spécifique : formules compliquées, dessins, vidéos...
- Parce qu'on souhaite une présentation dynamique au service d'une pédagogie pensée.
- Pour lutter contre l'entropie qui place toujours les notes manuscrites et les transparents dans un désordre maximal.
- Parce qu'il est plus facile d'accélérer un exposé trop long en appuyant frénétiquement sur la touche "suivant" qu'en jonglant avec des transparents.
- Parce que c'est très reposant de n'avoir qu'à appuyer sur une touche...
- Parce qu'on est allergique à la craie.
- Parce qu'on a une écriture de cochon.
- Parce qu'on est tendance...



**PowerPoint is Evil**™ : une critique des présentations à l'ordinateur...

# Les logiciels de présentation

Principe des logiciels de présentation :

- Le document est découpé en **diapositives** (*slides*).
- Les diapositives sont projetées en mode plein écran grâce à un vidéo projecteur.

Quelques logiciels courants :

**PowerPoint**™ : ( ) Il fait partie de la suite bureautique **Office**™ de Microsoft.

Quelques problèmes de compatibilités entre les versions **WINDOWS** et **MAC**.

**Impress**™ : (  ) C'est le module de présentation de **OpenOffice**.

Il est plus ou moins compatible avec le format de **PowerPoint**.

**Keynote**™ : () Il fait partie de la suite bureautique **iWork**™ d'Apple.

N'existe pas sur d'autres systèmes d'exploitation.

**Leurs avantages évidents** : Mise en page aisée, animations diverses à l'intérieur des diapositives et entre les diapositives, insertion possible de vidéos, choix d'un thème général qui modifie l'aspect global de la présentation, export en **PDF**...

**Leurs défauts principaux** : Logiciel compatible requis sur l'ordinateur de la présentation.

Difficile d'insérer des formules de mathématique, surtout dans le flux de texte.

➔ Les formules **LAT<sub>E</sub>X** peuvent être insérées en tant qu'images.

➔ Utiliser les mêmes polices que celles de **LAT<sub>E</sub>X** : **LATIN MODERN**™ en **OPENTYPE**™.

Ne semblent pas fournir une table des matières avec hyperliens internes.

# La classe beamer

La classe **beamer** répond à un besoin très ancien de créer une présentation à l'ordinateur à partir d'un document écrit en **L<sup>A</sup>T<sub>E</sub>X**.

Quelques ancêtres de **beamer** : la classe **slides** (1997), la classe **prosper** (2003), le package **texpower** (2005), Foil<sub>T</sub>E<sub>X</sub> et sa classe **foils** (2008)...

- ➔ **beamer** les dépasse tous en fonctionnalités.
- ➔ **beamer** propose un mode de compatibilité avec ces anciennes approches.

Les principes généraux de **beamer** sont les suivants :

- Le document produit est du **PDF**, qui est projeté avec n'importe quel logiciel capable d'afficher un **PDF** en plein écran.
  - ➔ Aucune dépendance à un logiciel particulier, portabilité du document.
- La classe **beamer** ajoute des commandes spécifiques, mais l'essentiel reste du **L<sup>A</sup>T<sub>E</sub>X** tout à fait standard.
  - ➔ Apprentissage basique très rapide.
  - ➔ Possible de copier-coller directement à partir d'un autre document **L<sup>A</sup>T<sub>E</sub>X**.
- La classe **article**, associée au package **beamerarticle**, produit un document à imprimer reprenant le texte de la présentation.
  - ➔ Possibilité de distribuer des notes imprimées plus fournies en texte.
- Notion de thèmes, modifiables à volonté avec le langage **L<sup>A</sup>T<sub>E</sub>X**.

# Les particularités de beamer

Une présentation à l'ordinateur admet certaines contraintes :

- Une présentation à l'ordinateur se fait en mode “paysage”.  
Les projecteurs ont en général une résolution de  $1024 \times 768$  pixels (rapport  $4 \times 3$ ).  
→ **beamer** fixe la taille de la page à  $128\text{mm} \times 96\text{mm}$ .
- Il est préférable d'utiliser des fontes sans empattements, y compris pour les mathématiques. → **beamer** se charge de basculer vers ces fontes.

 éviter `\mathrm{-}` → `\text{-}`, `\DeclareMathOperator{-}{-}`.

- La taille des caractères doit être assez grande.  
→ **beamer** crée un document en 11pt de taille “physique” réduite :  
à la projection, l'ensemble est agrandi et le texte est très lisible.  
Pour un projecteur de  $1024 \times 768$  pixels, les caractères font 31 pixels de haut.  
 À l'impression, penser à agrandir aussi les pages !
- Le texte est collé à gauche et non pas justifié.
- Il est ennuyeux d'avoir à appuyer de façon répétitive sur une touche (flèches) pour passer d'une diapositive à une autre afin de répondre à une question.  
**beamer** est capable de produire automatiquement une table des matières et de fournir une barre de navigation avec des hyperliens internes.  
→ Navigation très aisée au sein du document.

# Quelques problèmes de beamer

Du fait de sa dépendance au format de fichier **PDF**, la classe **beamer** souffre des quelques problèmes suivants :

**Les effets de transition :** Le format **PDF** n'est pas conçu pour disposer d'effets de transition à l'intérieur des diapositives ou entre les diapositives.

**Adobe Reader**® accepte quelques effets entre diapositives.

⚠ Ces effets ne fonctionnent pas nécessairement dans d'autres logiciels.

➔ **beamer** permet de les utiliser.

➔ Certains logiciels permettent d'ajouter des effets "après coup" à la main.

**Le multimédia :** Les afficheurs **PDF** n'ont pas tous des fonctionnalités multimédia.

**Adobe Reader** est capable de présenter certaines vidéos incluses dans le document sur **WINDOWS** et **MAC**, mais pas sur **LINUX**.

➔ Il est possible de déléguer ce travail à un logiciel externe.

**Des polices parfois absentes :** Parfois certains caractères d'écriture ne s'affichent pas de la même façon d'un ordinateur à un autre (symboles **ZAPF DINGBATS** par exemple).

Le document **PDF** n'inclut pas nécessairement toutes les fontes utilisées.

Les moteurs **pdf<sub>l</sub>atex** et **dvipdfm** insèrent toutes les fontes, mais pas **dvips**.

L'option `dvips -Pdownload35` permet d'insérer toutes les fontes.

Voir aussi le fichier de configuration **updmap.cfg**.

# Les commandes de base

Où l'on apprend à réaliser sa présentation avec quelques principes de base et à l'aide de quelques commandes nouvelles...

# La structure d'une présentation

Une présentation sous **beamer** peut être découpée globalement avec des commandes `\part`, `\section` et `\subsection` (`\subsubsection` existe mais l'éviter).

➔ `\part` n'est utile que dans des séries d'exposés (cours par exemple).

L'unité élémentaire d'une présentation est la **diapositive**, définie par la commande `\frame[-]{-}` ou l'environnement `\begin{frame}[-]{-} ... \end{frame}`.

➔ Dans la suite on ne considère que l'environnement `frame`.

Les éléments suivants peuvent être générés automatiquement :

**Une diapositive de titre** : il faut renseigner le titre, l'auteur, l'institut, la date...

**Des tables des matières** : il est possible de créer des diapositives contenant des tables de matières de différents niveaux et aspects.

**Des diapositives "section"** : elles peuvent marquer l'entrée dans une section.

**Une annexe** : une annexe peut être créée pour "cacher" des diapositives placées au delà de la fin de l'exposé (réserve pour les questions).

➔ Absente des tables des matières, non comptabilisée dans le total des diapositives.

**Concept de "pauses"** : chaque diapositive peut être subdivisée pour faire apparaître du texte étape par étape (1 étape = 1 page PDF).

Le compteur `beamerpauses` indexe ces étapes : il augmente d'une unité par étape définie.

# Un code source type

```
\documentclass[10pt]{beamer}
\usepackage[utf8]{inputenc}
\usepackage[TS1,T1]{fontenc}
\usepackage[english,french]{babel}
\usepackage{lmodern}

\usetheme{Goettingen}

\title[Beamer en action]{Beamer en action}
\author[T. Masson]{Thierry Masson}
\institute[CPT]{Centre de Physique Théorique}
\date[mai 2011]{mai 2011}

\begin{document}
\section{Une première section}
\begin{frame}{Une première diapositive}
  Bonjour !
\end{frame}
\end{document}
```

**beamer** charge les *packages* suivants :

**geometry**

**graphicx**

**xcolor**

**pgf**

**hyperref**

**amssymb**

**amsmath**

**amsfonts**

**amsthm**

**translator**

**enumerate**

# La structure d'une diapositive en images

barre de navigation

entête

corps du texte

pied

symboles de navigation

The diagram illustrates the structure of a Beamer slide. It shows a slide titled "Proof That There Is No Largest Prime Number" with a subtitle "A proof using *reductio ad absurdum*". The slide is divided into several sections:

- barre de navigation**: Located at the top left, it contains navigation symbols and the slide title.
- entête**: The main title and subtitle of the slide.
- corps du texte**: The main content area, containing the theorem and proof.
- pied**: The footer area, containing the author and institution information.
- symboles de navigation**: Located at the bottom right, it contains navigation symbols.

The slide content is as follows:

Results  
 Proof of the Main Theorem

**Proof That There Is No Largest Prime Number**  
 A proof using *reductio ad absurdum*.

**Theorem**  
*There is no largest prime number.*

**Proof.**

- Suppose  $p$  were the largest prime number.
- Let  $q := 1 + \prod_{i=1}^p i = 1 + p!$ .
- Then  $q$  is not divisible by any  $p' \in \{1, \dots, p\}$ .
- Thus  $q > p$  is also prime.  $\square$

Euklid of Alexandria  
 Department of Mathematics, University of Alexandria

# La structure d'une diapositive en théorie

Une diapositive est constituée de différents éléments “physiques” :

**Le barre de navigation :** cette barre contient en général le rappel de la structure logique de la présentation : sections, sous sections, diapositives.

Cette barre peut être horizontale ou verticale selon le thème choisi.

**L'entête :** c'est l'endroit usuel pour placer le titre (et le sous titre) de la diapositive.

**Le corps du texte :** c'est le contenant de ce qui est présenté dans la diapositive.

**Le pied :** on peut y placer le titre de la conférence, le nom du présentateur et son institut, la date, le numéro de la diapositive...

**Les symboles de navigation :** ce sont des symboles (discrets) cliquables permettant la navigation dans la présentation : début, section, table des matières...

Ces éléments ont une apparence et une position déterminées par le thème choisi.

Ils peuvent être personnalisés grâce à des modèles (*templates*) facile à modifier.

De nombreux textes (titre, sections, sous sections...) et symboles insérés dans ces éléments sont des hyperliens internes au document.

➔ Accès facile et permanent à diverses parties du document.

# L'environnement `frame`

Plusieurs syntaxes sont possibles pour l'environnement `frame` :

```
\begin{frame}{Titre}
...
\end{frame}
```

```
\begin{frame}{Titre}{Sous Titre}
...
\end{frame}
```

```
\begin{frame}
\frametitle{Titre}
\framesubtitle{Sous Titre}
...
\end{frame}
```

Les commandes `\frametitle` et `\framesubtitle` sont optionnelles.

L'option `[plain]` ôte les barres de navigation et le pied de la diapositive.

```
\begin{frame}[plain]{Titre}
...
\end{frame}
```

```
\begin{frame}[plain]
...
\end{frame}
```

➔ Gain d'espace pour le contenu de la diapositive (présentation d'un graphique par ex.).

Par défaut, le texte est centré verticalement dans une diapositive.

Les options `t`, `b` et `c` (défaut) de `frame` peuvent modifier ce comportement.

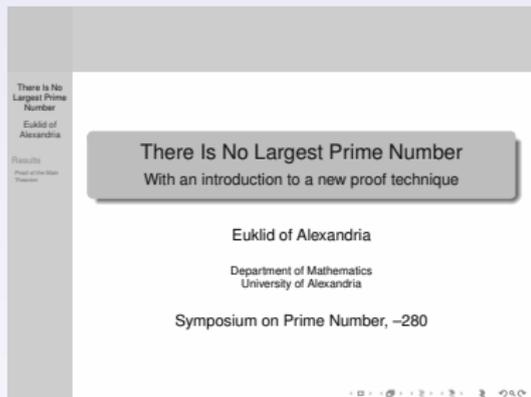
# La diapositive de titre

La diapositive de titre peut-être créée par :

```
\begin{frame}[plain]
  \titlepage
\end{frame}
```

On peut écrire plus simplement :

```
\maketitle
```

Les champs à renseigner pour créer la diapositive de titre sont :

`\title[-]{-}` définit le titre de la présentation.

`\subtitle[-]{-}` définit le sous titre de la présentation.

`\author[-]{-}` désigne la liste des auteurs séparés par `\and`.

`\institute[-]{-}` désigne la liste des instituts séparés par `\and`.

`\date[-]{-}` définit la date, et on ajoute souvent le lieu de la conférence.

`\titlegraphic{-}` désigne un graphique de titre.

Les arguments optionnels dans les commandes ci-dessus sont utilisés pour le rappel des données sur chaque diapositive. ➔ Fournir une version plus courte.

La composition de la diapositive de titre dépend du thème choisi.

Il existe des commandes `\partpage`, `\sectionpage` et `\subsectionpage`.

# Les tables des matières

Les tables des matières sont composées des titres des sections et des sous-sections.

`\tableofcontents [options]` avec les options :

`sectionstyle=` détermine le style de la section courante et des autres sections :

*sect. courante/autres sect.* avec les valeurs `show`, `shaded`, `hide`.

Si le second est omis, il prend la valeur du premier.

`subsectionstyle=` détermine de même l'aspect des sous sections :

*ss-sect. courante/autres ss-sect. dans la sect. courante/autres ss-sect..*

Si le troisième est omis, il prend la valeur du second.

Si le second est omis aussi, les trois prennent la valeur du premier.

`currentsection` équivaut à

`sectionstyle=show/shaded,subsectionstyle=show/show/shaded.`

`currentsubsection` équivaut à `subsectionstyle=show/shaded.`

`hideallsubsections` équivaut à `subsectionstyle=hide.`

`hideothersubsections` équivaut à `subsectionstyle=show/show/hide.`

Avec ces options, les tables des matières n'affichent pas nécessairement la même chose à différents endroits de la présentation.

D'autres options sont disponibles.

Les tables des matières ont des hyperliens internes au document.

# Les colonnes

**beamer** propose une mise en page (locale) sur plusieurs colonnes.

```
\begin{columns}[t]
  \begin{column}[c]{0.6\textwidth}
    ...
  \end{column}
  \begin{column}{0.35\textwidth}
    ...
  \end{column}
\end{columns}
```

L'option de l'environnement **columns** définit la position horizontale des colonnes qu'il contient : **t**, **b**, **c** (défaut), **T** (alignement haut première ligne).

**columns** supprime les marges gauche et droite ➔ on dispose de toute la page en largeur.

L'option **onlytextwidth** réduit à la zone de texte.

L'option de l'environnement **column** définit la position horizontale (mêmes options).

On peut placer autant de **column** qu'on veut (peut) dans **columns**.

Les **column** contiennent n'importe quel matériel **L<sup>A</sup>T<sub>E</sub>X** : texte, tableau, graphique...

On peut placer plusieurs **columns** (l'un en dessous de l'autre) dans la même diapositive.

➔ À utiliser sans modération pour des vis-à-vis "texte"/"graphique".

## Avancer par étapes

Certaines commandes usuelles et certains environnements courants de  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  ont été étendus pour accepter un nouvel argument optionnel  $\langle \rangle$  :

$\backslash\text{cmd}\langle\text{étapes}\rangle([\text{-}]\{\text{-}\})$  où les autres arguments conservent leur place et nombre.

$\langle\text{étapes}\rangle$  définit le comportement de  $\backslash\text{cmd}$  par rapport aux étapes de la diapositive.

➔ Fait référence au compteur `beamerpauses` :

$\langle m-n \rangle$  active la commande entre les valeurs  $m$  et  $n$  du compteur `beamerpauses`.

$\langle m- \rangle$  active la commande à partir de la valeur  $m$  du compteur `beamerpauses`.

$\langle -n \rangle$  active la commande jusqu'à la valeur  $n$  du compteur `beamerpauses`.

$\langle .- \rangle$  active la commande à partir de la valeur courante du compteur `beamerpauses`.

$\langle +- \rangle$  active la commande à partir de la valeur courante du compteur `beamerpauses` et incrémente `beamerpauses` d'une unité. ➔ **Syntaxe la plus utile!**

$\langle\text{étapes1}, \text{étapes2}, \dots \rangle$  combine plusieurs intervalles.

Le compteur `beamerpauses` est défini à 1 au début de l'environnement `frame` et atteint la valeur maximale requise par les commandes  $\backslash\text{cmd}\langle\text{étapes}\rangle([\text{-}]\{\text{-}\})$  rencontrées.

La diapositive sera répartie sur autant de pages physiques PDF que la valeur maximale atteinte par ce compteur.

⚠ La syntaxe  $\langle\text{étapes}\rangle$  est en fait plus générale, et accepte une spécification du mode : `presentation`, `beamer`, `trans`, `handout` et `article`.

Une commande  $\backslash\text{pause}$  permet de gérer les étapes d'une diapositive ➔ non évoquée ici.

## Quelques commandes gérant les étapes

`\textbf`, `\textit`, `\textsl`, `\textrm`, `\textsf` acceptent *<étapes>* comme premier argument : `\textbf<2-4>{En gras de 2 à 4}`

`\color<étapes>[modèle]{spécifications}` active la couleur définie par le modèle et les spécifications sur les étapes spécifiées par *<étapes>*.

`\label<étapes>{label}` insère le label *label* sur la page correspondant à l'étape spécifiée par *<étapes>*.

⚠ *étapes* ne doit contenir qu'une seule étape !

`\uncover<étapes>{texte}` affiche *texte* sur les étapes spécifiées par *étapes*.

En dehors de ces étapes, *la place pour texte est réservée*.

Il est possible d'afficher en "transparent" le texte caché (`\setbeamercovered{-}`).

`\only<étapes>{texte}` affiche *texte* sur les étapes spécifiées par *étapes*.

En dehors de ces étapes, *la place pour texte n'est pas réservée*.

`\temporal<étapes>{texte avant}{texte pendant}{texte après}` affiche *texte avant* avant les étapes spécifiées par *étapes*, *texte pendant* sur les étapes spécifiées par *étapes* et *texte après* après la dernière étape spécifiée par *étapes*.

D'autres commandes existent pour dévoiler et cacher du texte.

Les arguments des commandes `\uncover`, `\only` et `\temporal` peuvent contenir du texte, des tableaux, des images et même des définitions de commandes...

Ces commandes peuvent s'emboîter.

# Les listes

**beamer** retient les trois sortes de liste de **L<sup>A</sup>T<sub>E</sub>X** : **itemize**, **enumerate** et **description**.

L'apparence de ces listes et de leurs items est modifiable.

Les items des listes acceptent l'argument *<étapes>* :

```
\begin{itemize}
  \item<1-> Un
  \item<2-> Deux
\end{itemize}
```

```
\begin{enumerate}
  \item<+> Un
  \item<+> Deux
\end{enumerate}
```

```
\begin{description}
  \item<1-3> [Un]
  \item<2-4> [Deux]
\end{description}
```

Autre syntaxe :

```
\begin{itemize}[<+>]
  \item Un
  \item Deux
  \item Trois
\end{itemize}
```

```
\begin{enumerate}[<+>]
  \item Un
  \item<1-> Deux
  \item Trois
\end{enumerate}
```

➔ Raccourci qui fait apparaître les items les uns après les autres sauf mention contraire.

L'énumération reprend les options du *package* **enumerate** (déjà chargé) :

```
\begin{enumerate}[<+>][(i)] ... \end{enumerate}
```

## Mettre en valeur du texte

Les thèmes installent une couleur dominante, appelée “couleur de structure”.

`\structure<>{-}` met du texte dans la couleur de structure.

`\structure{Couleur structure}` → Couleur structure

`\alert<>{-}` met en valeur du texte en le colorant en rouge par défaut.

`\alert{En alerte !}` → En alerte!

**beamer** propose de mettre en valeur des blocs de texte de plusieurs lignes :

```
\begin{block}{Un titre}
  Du texte en boîte
\end{block}
```

```
\begin{alertblock}{Un titre}
  Du texte en alerte
\end{alertblock}
```

```
\begin{exampleblock}{Un titre}
  Du texte en exemple
\end{exampleblock}
```

Ces trois types de blocs ont chacun une couleur assignée (modifiable).

⚠ Le terme “bloc” est excessif : dans certains thèmes il n’y a pas de bloc dessiné...

Ces trois environnements acceptent un argument `<>` :

```
\begin{block}<>{-} ... \end{block}
```

→ N’affiche le bloc que selon les spécifications de `<>`, sa place est réservée.

## Les espaces réservés

Ajouter ou ôter du texte produit une recomposition de la diapositive ➔ non souhaitable.

```
\begin{overlayarea}{⟨dimh⟩}{⟨dimv⟩} ... \end{overlayarea}
```

réserve une zone de travail de largeur  $\langle dimh \rangle$  et de hauteur  $\langle dimv \rangle$ .

➔ Dans cette zone, des `\only` ne changeront pas la mise en page globale.

```
\begin{overlayarea}{\textwidth}{3cm}
  \only<1> {Lorem ipsum dolor sit amet.}
  \only<2->{Fusce pretium ullamcorper neque sit amet luctus.}
\end{overlayarea}
```

`\begin{overprint}[-] ... \end{overprint}` réserve un espace vertical de hauteur minimale (calculée) adaptée à son contenu.

La largeur vaut `\textwidth` sauf mention contraire dans l'argument optionnel.

Les étapes du contenu sont définies par la commande `\onslide<>` qui se comporte un peu comme `\item<>`. ➔ Les `\onslide<étapes>` doivent être disjoints.

```
\begin{overprint}
  \onslide<1> Lorem ipsum dolor sit amet.
  \onslide<2-> Fusce pretium ullamcorper neque sit amet luctus.
\end{overprint}
```

# Les mathématiques

Les mathématiques dans **beamer** sont les mêmes que dans **L<sup>A</sup>T<sub>E</sub>X**.

Les *packages* **amsmath** et **amsthm** sont chargés par défaut (sauf avec l'option **noamsthm**).

➔ On peut utiliser **amssymb**, **mathtools**, **xy**...

**beamer** installe des polices sans sérif pour les mathématiques.

L'option **mathserif** de **beamer** laisse les polices avec sérif en mathématique.

➔ À éviter pour l'harmonie et la lisibilité.

Les environnements **theorem**, **corollary**, **definition**, **definitions**, **fact**, **example**, **examples** sont définis et se présentent comme des blocs.

➔ L'option **notheorems** de **beamer** désactive la définition de ces blocs.

**\newtheorem** et **\theoremstyle** sont disponibles (syntaxe de **amsthm**).

```
\theoremstyle{plain}
\newtheorem{proposition}[theorem]{\translate{Proposition}}
\theoremstyle{example}
\newtheorem{remark}[theorem]{\translate{Remark}}
```

Voir **\translate{-}** plus loin.

Ces environnements acceptent automatiquement un argument  $\langle \rangle$  :

```
\begin{definition}\langle \rangle{-} ... \end{definition}
```

➔ Apparition du bloc en entier.

➔ Insérer des **\uncover\langle \rangle{-}** à l'intérieur pour affiner les étapes.

# Boutons et hyperliens internes

Il est possible et facile de définir des hyperliens internes entre pages.

La syntaxe est celle du package **hyperref** :

`\hypertarget<étapes>{label interne}{texte}` définit un but pour un hyperlien interne, de clé interne *label interne*, de but la page PDF résultant de *<étapes>*.

En dehors des spécifications de *<étapes>*, *texte* n'est pas affiché.

`\hyperlink<étapes>{label interne}{texte}` rend *texte* cliquable dans les spécifications *<étapes>* et renvoie vers le but défini par *label interne*.

Pour rendre les choses plus jolies, on peut placer des boutons dans `\hyperlink` :

`\beamerbutton{texte du bouton}` dessine un bouton.

`\beamergetobutton{texte du bouton}` bouton avec un flèche vers la droite.

`\beamerskipbutton{texte du bouton}` bouton avec une double flèche vers la droite.

`\beamerreturnbutton{texte du bouton}` bouton avec une flèche vers la gauche.

```
\hyperlink<3->{labelA}{\beamergetobutton{C'est par là !}}
```

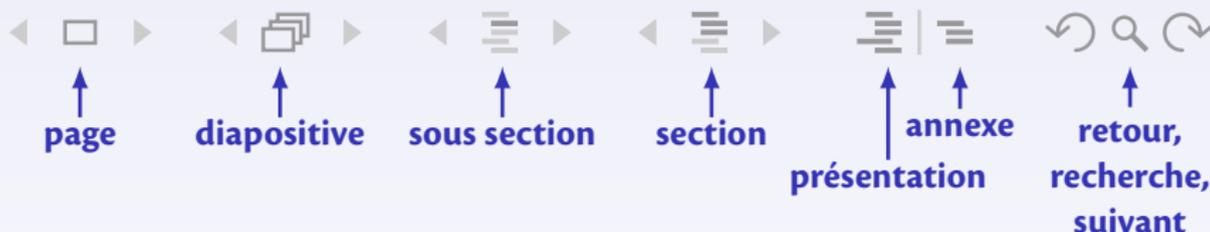
De nombreux buts sont déjà définis par **beamer** auxquels on peut accéder avec les commandes : `\hyperlinkslideprev{texte}`, `\hyperlinkslidenext{texte}`,

`\hyperlinkframestart{texte}`, `\hyperlinkframeend{texte}`,

`\hyperlinkframestartnext{texte}`, `\hyperlinkframeendprev{texte}`...

# La barre de navigation

Par défaut, une barre de navigation est insérée en bas à droite des diapositives. Elle permet de naviguer dans le document grâce à des hyperliens internes.



**page** est une page PDF,

**diapositive** symbolise l'ensemble des pages qui composent une unique diapositive,

**sous section** est un début de sous section,

**section** est un début de section,

**présentation** désigne la fin de la présentation (avant l'annexe),

**annexe** est le début de l'annexe.

**retour, recherche** et **suivant** permettent de naviguer directement dans le fichier PDF.

Les flèches à droite et à gauche des symboles font naviguer d'une entité à une autre.

On peut ôter cette barre en insérant dans le préambule :

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

## Quelques trucs en vrac

Il est possible d'écraser verticalement une diapositive qui serait trop longue :

```
\begin{frame}[squeeze]{Titre}
  Texte un peu trop long
\end{frame}
```

```
\begin{frame}[shrink=5]{Titre}
  Texte un peu trop long
\end{frame}
```

**squeeze** réduit certains espaces verticaux à zéro.

**shrink** réduit globalement le contenu de la diapositive du pourcentage donné.

➔ Le contenu de la diapositive est réellement réduit, y compris la taille des caractères !

Il est possible d'interrompre une diapositive, d'en insérer d'autres, puis de la continuer :

```
\begin{frame}<1-5>[label=diapo]{Titre}
  Contenu total
\end{frame}
autres diapositives
\againframe<6->{diapo}
```

La première diapositive contient *tout* le matériel.

D'abord, elle n'est présentée que pour les valeurs **beamerpauses**=1 à 5.

Plus loin, on termine cette diapositive à partir de **beamerpauses**=6.

➔ Possible d'utiliser plusieurs fois **\againframe**<>{-} pour la même diapositive, y compris pour des valeurs de **beamerpauses** se chevauchant.

## Quelques trucs en vrac (suite)

Ajouter automatiquement une table des matières à chaque début de section :

```
\AtBeginSection[]{\begin{frame}\frametitle{\insertsection}\tableofcontents[currentsection,hideothersubsections]\end{frame}}
```

`\AtBeginSection` installe du matériel à chaque début de section.

➔ Possible d'ajouter aussi une diapositive avec `\sectionpage`.

Il existe aussi `\AtBeginPart`, `\AtBeginSubsection` et `\AtBeginLecture`

```
\setbeamercovered{transparent}
```

rend les éléments cachés “transparent”, donc prévisualisables par le public.

L'inverse est `\setbeamercovered{invisible}` (défaut).

Par défaut, **beamer** prend en compte le texte entre les environnements `frame`.

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

➔ **beamer** ignore le texte hors des `frame` et `\section...`

## Quelques trucs en vrac (suite)

Pour gagner de la place verticalement entre du texte et des formules de mathématique :

```
\divide\abovedisplayskip by 2  
\divide\belowdisplayskip by 2
```

Utiliser des flèches claires et nettes (➡) et non des flèches mathématiques (⇒) :

```
\usepackage{pi font, relsize}  
\newcommand{\flechedroite}{\raisebox{-0.5ex}%  
{\relsize{2}\ding{217}}}\ }
```

➡ puiser dans **pi font** d'autres idées...

Ouvrir le document directement en plein écran :

```
\hypersetup{pdfpagemode=FullScreen}
```

Ceci ne fonctionne pas pour tous les visualisateurs PDF.

**On ne distribue pas le code source d'une présentation** : ne pas hésiter à utiliser des commandes personnelles pour la mise en page, l'apparence, une signalétique...

➡ Faire un fichier à part pour les utiliser régulièrement.



# Positionner des images dans la diapositive

Il est possible de placer un graphique en bout de ligne sans perturber la mise en page :

Du texte

```
\hfill\raisebox{-3pt}[0pt][0pt]{\makebox[0pt][r]{GRAPHIQUE}}
```

Suite du texte

Le graphique est au final une boîte de taille nulle collée à droite de la diapositive.

Le premier argument de `\raisebox` joue sur sa position verticale.

 Prévoir de la place sur la partie droite du texte. 

Utile aussi pour couvrir momentanément du texte avec un graphique grâce à `\only`.

Pour des mises en page plus complexes, utiliser un graphique `TikZ`.

➔ Possibilité de positionner dans un environnement `tikzpicture` du texte, des formules de mathématique, des graphiques et des liens logiques entre tous ces objets.

➔ Possibilité de dynamiser le graphique avec des pauses.

Utiliser les colonnes de `beamer` pour placer du texte et des graphiques en vis à vis.



# La traduction des mots clés

**beamer** est accompagné d'un *package* de traduction nommé **translator** (chargé).

**Principe** : des dictionnaires thématiques sont proposés dans différentes langues, "English" (défaut), "French", "German"...

Ces dictionnaires proposent des traductions pour des mots clés usuels : mois, chiffres, mathématique (théorème, définition...)...

La commande `\translate{-}` s'occupe de traduire des mots à l'aide de ces dictionnaires.

`\translate{Theorem}` ➔ Théorème (en français).

`\translate{July}` ➔ Juillet (en français).

Dans le préambule, on charge les dictionnaires de langue française par

```
\uselanguage{French}
```

Ceci équivaut à

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

On active la langue choisie, dans le préambule ou localement dans le texte (basculement) :

```
\languagepath{French}
```

# Des vidéos avec beamer

Le package **multimedia**, fourni avec **beamer**, permet d'insérer des vidéos et du son dans des documents PDF.

`\movie[-]{poster}{fichier vidéo}` :

- Le code *poster* représente du texte ou une image (`\includegraphics`). Il symbolise l'élément multimédia sur la page.
- En cliquant sur *poster*, le fichier multimédia *fichier vidéo* se lance.
- Les options gèrent le comportement de la vidéo et l'aspect de *poster*.

⚠ Certains visualisateurs PDF ne sont pas capables de gérer des fichiers multimédia.

```
\usepackage{multimedia}
...
\movie[externalviewer,width=120pt]%
    {\includegraphics[width=120pt]{image.jpg}}%
    {video.mp4}
```

Avec l'option `externalviewer`, le visualisateur PDF délègue au système le soin de trouver le lecteur multimédia adéquat. ➡ Il faut régler ce problème au niveau du système.

La documentation de **multimedia** est insérée dans celle de **beamer**.

## Autres fonctionnalités à explorer

**TikZ** est compatible avec **beamer** (même créateur : Till Tantau).

Il fonctionne avec la commande `\uncover<>{-}`. ➔ Graphiques dynamiques.

**Position absolue** : avec le package **textpos**, il est possible de placer des objets de façon absolue dans une diapositive. ➔ Voir la documentation de ce package.

**Bibliographie** : les commandes `\cite` et l'environnement **thebibliography** sont définis mais **beamer** n'est pas compatible avec **bibtex**.

On peut modifier l'aspect de la bibliographie.

**Définitions de commandes** : il est possible de (re)définir des commandes et des environnements acceptant l'argument *<étapes>* : `\newcommand<>`, `\renewcommand<>`, `\newenvironment<>`, `\renewenvironment<>`.

⚠ Seul **Adobe Reader**<sup>®</sup> est 100% compatible avec les fonctionnalités ci-dessous :

**Transitions** : **beamer** fournit des commandes pour installer des transitions :

`\transblindshorizontal<>[-]`, `\transdissolve<>[-]`,  
`\transsplithorizontalin<>[-]`...

**Animations** : `\animate<>` et `\animatevalue<>{-}{-}{-}` permettent des effets d'animations en "jouant automatiquement" une série d'étapes (pages PDF).

Lire la documentation de **beamer** pour plus de renseignements.

## Conseils divers

- La mise en page sous forme de listes est très efficace pour segmenter les idées.
- Le découpage d'une diapositive par étapes est souhaitable pour synchroniser la parole avec le visuel.
  - ➔ Mais éviter les éléments cachés "transparents"...
- Les couleurs :
  - Ne surtout pas abuser de couleurs diverses et variées.
    - ➔ Codifier une fois pour toute le choix des couleurs.
  - Ne pas jouer trop finement avec les effets de couleurs : les vidéoprojecteurs ne sont pas fidèles en matière de rendu des couleurs et des contrastes.
    - ➔ Tester certaines couleurs avant de les adopter (ou de les rejeter).
  - Penser aux daltoniens...
- Si l'arrière plan des diapositives est coloré, prévoir des figures avec fond transparent.
  - ➔ Sinon encadrer l'image pour marquer ses bords.
- Ne pas surcharger une diapositive :
  - dans son contenu ;
  - dans les éléments structurants, informatifs, graphiques et visuels qui la composent.
- Faire la chasse aux polices à empattement, surtout dans le mode mathématique.
  - ➔ Éradiquer les `\mathrm{-}` de toutes les commandes personnelles !
- Souvent inutile de numéroter les équations...

# Modifier l'aspect de sa présentation

Où l'on apprend à modifier l'apparence locale ou globale de sa présentation en faisant bon usage des modèles et des thèmes...

# Modifier des éléments d'une présentation

**Règle :** presque tous les éléments d'une présentation **beamer** peuvent être modifiés.

Chaque élément requiert trois types de définition pour être rendu :

- un modèle (*template*), c'est à dire du code pour construire l'élément ;
- une couleur (*color*) ;
- une typographie (*font*).

Certains éléments n'ont pas besoin de ces trois types.

Une notion d'héritage est possible entre éléments (*parent*).

À chaque élément est associé un nom assez explicite :

**Éléments globaux :** `background`, `title page`, `section page`, `headline`, `footline`,  
`navigation symbols`, `section in head/foot`, `page number in head/foot`,  
`frametitle`, `framesubtitle`...

**Éléments locaux :** `itemize item`, `itemize subitem`, `enumerate item`,  
`enumerate subitem`, `description item`, `bibliography item`, `button`,  
`section in toc`, `block begin`, `block end`, `qed symbol`...

**Textes :** `normal text`, `structure`, `alerted text`, `example text`, `title`,  
`subtitle`, `author`, `institute`, `date`...

On modifie l'aspect d'un élément à l'aide du nom associé et de commandes spécifiques.

# Les commandes pour modifier les éléments

`\setbeamertemplate{nom de l'élément}{code}` définit le *template* de *nom de l'élément* par le code  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  *code*.

Selon le contexte, des commandes sont fournies pour remplir les *templates* :

`\inserttitle`, `\insertpagenumber`, `\insertsection`, `\insertshortauthor`,  
`\insertframenumbers`, `\insertframetitle`...

`\setbeamertemplate{nom de l'élément}[nom du modèle]` active un *template* avec un nom prédéfini.

`\setbeamercolor{nom de l'élément}{code}` définit la couleur *nom de l'élément* par une liste `fg=`, `bg=`, `parent=` et/ou `use=` séparés par des virgules.

`\setbeamerfont{nom de l'élément}{code}` définit la couleur *nom de l'élément* par une liste `family=`, `series=`, `shape=`, `size=` et/ou `parent=` séparés par des virgules.

```
\setbeamertemplate{navigation symbols}[only frame symbol]
\setbeamertemplate{itemize items}[square]
\setbeamerfont{title}{size=\LARGE,series=\bfseries}
\setbeamerfont{author}{size=\large,shape=\slshape}
\setbeamerfont{alerted text}{series=\bfseries,shape=\slshape}
```

# Les couleurs dans beamer : principes

Une couleur au sens de **beamer** est en réalité un couple de couleurs :

**Une couleur d'avant plan** qui correspond à la couleur de l'élément lui-même, par exemple à la couleur des lettres d'un texte.

Elle est désignée par **fg** (*foreground*).

**Une couleur de fond** qui correspond à la couleur d'arrière plan.

Elle est désignée par **bg** (*background*).

Quelques couleurs sont définies par défaut :

**Couleur du texte** : associée au nom **normal text**.

➔ le texte et le fond de la diapositive.

**Couleur de structure** : associée au nom **structure**.

➔ commande `\structure`, utilisée et déclinée dans de nombreux éléments.

**Couleur d'alerte** : associée au nom **alerted text**

➔ commande `\alert` et blocs d'alerte.

**Couleur d'exemple** : associée au nom **example text**

➔ blocs d'exemples.

Il est possible de définir des couleurs pour les mathématiques différentes de celles du texte !

Utiliser les noms (explicites) **math text**, **math text inlined**,

**math text displayed** et **normal text in math text**...

# Les couleurs dans beamer : définitions et utilisations

**Règle 1 :** rien n'empêche d'utiliser les commandes usuelles pour les couleurs.

**Règle 2 :** les couleurs peuvent être définies avec la syntaxe du package **xcolor** (chargé).

```
\setbeamercolor{normal text}{bg=red!20,fg=black!80}
\setbeamercolor{alerted text}{fg=magenta}
```

La syntaxe **bg=** (sans rien après le signe =) désigne une couleur "transparente".

**parent=** permet de faire un héritage de couleurs :

```
\setbeamercolor{maman}{bg=red,fg=black}
\setbeamercolor{fiston}{parent=maman,fg=purple}
```

➔ la couleur de fond est héritée, la couleur d'avant plan est redéfinie.

**use=** permet d'utiliser d'autres couleurs et de les modifier :

```
\setbeamercolor{papa}{use={structure,normal text},
fg=structure.fg!50,bg=normal text.bg!80!red}
```

`\usebeamercolor[fg,bg]{nom couleur}` installe une couleur et compose le texte avec.

L'option permet de n'installer que la couleur d'avant plan ou de fond.

`{\usebeamercolor[fg]{fiston} Fiston}` ➔ **Fiston**

## Changer certaines dimensions

**beamer** délègue à **geometry** (chargé) le soin de fixer la taille des diapositives.

On peut agrandir les diapositives :

```
\geometry{papersize={160mm,120mm}}
```

⚠ Conserver le rapport  $4 \times 3$ , sauf usage particulier.

D'autres longueurs peuvent être modifiées avec `\setbeamerwidth{-}`.

Certaines valeurs modifiables sont :

`text margin left=<dim>`, `text margin right=<dim>` définit les marges du texte (hors des barres de navigation).

`sidebar width left=<dim>`, `sidebar width right=<dim>` définit la largeur des barres de navigation à gauche et à droite.

`description width=<dim>`, `description width of={texte}` définit la largeur des items de description.

Changer la taille des marges :

```
\setbeamerwidth{text margin left=0.3cm, text margin right=0.3cm}
```

# Les couches d'une diapositive

**beamer** construit une diapositive en 3 couches :

**Le canevas de fond** : c'est le canevas sur lequel tout le reste est dessiné.

**background canvas** est un modèle, une couleur et une typographie.

Les modèles prédéfinis sont :

**default** installe un rectangle de couleur **background canvas**.

**vertical shading** installe un dégradé vertical où il est possible de désigner 3 couleurs et de fixer le point milieu du dégradé.

La couleur **background canvas** hérite de la couleur **normal text**

**Le fond principal** : c'est un "fond" installé au dessus du canevas de fond.

➔ prévu pour du matériel de type grille, image...

Les modèles prédéfinis sont **default** (vide) et **grid** (grille).

Prévoir du matériel **L<sup>A</sup>T<sub>E</sub>X** de taille `\paperwidth` × `\paperheight`.

```
\setbeamertemplate{background}%  
    {\includegraphics[width=\paperwidth]{image.jpg}}  
\begin{frame}[plain] ... \end{frame}  
\setbeamertemplate{background}[default]
```

installe une image de fond pour une seule diapositive.

**Les objets qui composent la diapositive** : c'est le reste du matériel placé au dessus...

## Des boîtes de couleur

De nombreux éléments sont construits à l'aide de boîtes définies par **beamer**.

```
\begin{beamercolorbox}[-]{-} ... \end{beamercolorbox}
```

créé une boîte colorée : l'argument obligatoire est une couleur **beamer**.

➔ la boîte colore le fond et le texte avec **bg** et **fg** respectivement.

Parmi les options on trouve :

**wd**=*<dim>*, **dp**=*<dim>*, **ht**=*<dim>* désignent la largeur, la profondeur et la hauteur finales.

**left**, **center**, **right** indique l'alignement du texte à l'intérieur de la boîte.

**shadow**, **rounded** installent un effet d'ombre et des bords arrondis.

**ignorebg** crée une boîte sans fond.

```
\begin{beamerboxesrounded}[-]{-} ... \end{beamerboxesrounded}
```

créé une boîte colorée, arrondie, avec titre (argument obligatoire, peut être vide).

**upper**=*couleur*, **lower**=*couleur* désignent les couleurs **beamer** du bandeau de titre et du contenu respectivement.

**width**=*<dim>* indique la largeur du texte.

**shadow**= installe (*true*) ou non (*false*) une ombre.

Ces commandes sont utilisables partout dans une présentation.

## Quelques personnalisations utiles

Placer le numéro de la diapositive dans le pied :

```
\setbeamertemplate{footline}{%
\begin{beamercolorbox}[ht=0ex,dp=0ex,
    leftskip=.3cm,rightskip=.3cm plus1fil]%
    {section in head/foot}%
\raisebox{1.125ex}%
{\tiny\bfseries\insertframenumber/\inserttotalframenumber}%
\hfill%
\end{beamercolorbox}}
```

➔ Pour un thème sans `footline` prédéfini, sinon adapter le modèle existant.

Installer des logos sur la page de titre :

```
\titlegraphic{%
\begin{center}
\includegraphics[height=1.2cm]{LogoCNRS}\hspace{8mm}
\includegraphics[height=1.2cm]{LogoCPT}
\end{center}}
```

➔ Inséré dans le `template title page` par `\inserttitlegraphic`.

# Généralité sur les thèmes

Il est fastidieux de redéfinir tous les éléments d'une présentation pour la mettre à son goût.

**beamer** fournit des thèmes à installer pour modifier tout ou partie des éléments.

Par défaut, un thème appelé **default** est installé (pas nécessaire de le charger).

Les thèmes sont classés en 4 catégories selon ce qu'ils modifient :

**Thèmes extérieurs** modifient l'aspect des éléments extérieurs de la présentation : entête, pied, barres de navigations, symboles de navigation, titre des diapositives...

`\useoutertheme[-]{-}` dans le préambule charge un tel thème.

**Thèmes intérieurs** modifient l'aspect des éléments intérieurs de la présentation : listes, blocs, environnements de théorèmes, tables des matières...

`\useinnertheme[-]{-}` dans le préambule charge un tel thème.

**Thèmes de couleurs** modifient les couleurs des éléments.

`\usecolortheme[-]{-}` dans le préambule charge un tel thème.

**Thèmes de fontes** modifient les fontes utilisées dans les éléments.

`\usefonttheme[-]{-}` dans le préambule charge un tel thème.

La classe **beamer** accepte les tailles de **8pt** à **12pt** (jusqu'à **20pt** avec **extsize**).

On peut installer des *packages* de polices globaux : **modern** (recommandé), **helvet**, **libertine**, **arev**...

Les thèmes extérieurs et intérieurs ne touchent pas aux couleurs et aux fontes.

# Les thèmes de couleurs

Il y a trois types de thèmes de couleurs :

**Thèmes couleurs extérieurs** pour les éléments extérieurs (noms : animaux marins...).

**Thèmes couleurs intérieurs** pour les éléments intérieurs (noms : fleurs...).

**Thèmes couleurs complets** pour tout à la fois (noms : animaux volants...).

**beamer** installe deux palettes de couleurs (*i.e.* une couleur de base déclinée).

`palette primary,`      `palette sidebar primary,`  
`palette secondary,`    `palette sidebar secondary,`  
`palette tertiary,`      `palette sidebar tertiary,`  
`palette quaternary,`   `palette sidebar quaternary.`

Ces couleurs n'ont pas de couleur de fond par convention.

Les thèmes extérieurs doivent utiliser `palette primary` et ses déclinaisons.

`primary` correspond aux éléments devant le plus attirer l'attention,

`quaternary` aux éléments devant le moins attirer l'attention.

`palette sidebar primary` et ses déclinaisons sont réservées aux barres de navigations sur le côté de la diapositive.

Un thème couleur extérieur installe en général la palette à partir de la couleur `structure`.

Les thèmes couleurs internes utilisent en plus les couleurs `normal text`, `alerted text` et `example text`. ➔ On peut modifier soi-même ces couleurs.

# Les thèmes clés en main

Des **thèmes de présentation** sont aussi définis.

Ils regroupent de façon cohérente des ensembles de 4 thèmes ci-dessus.

`\usetheme[-]{-}` dans le préambule charge un tel thème.

Ces thèmes de présentation portent des noms de villes, sauf `default` et `boxes`. (`Marseilles` est encore disponible...)

`boxes` est conçu pour ajouter des boîtes dans la barre de navigation et le pied.

```
\usetheme[headheight=2.5ex]{boxes}
\addheadbox{section in head/foot}%
           {\footnotesize\quad\insertsection}
\addfootbox{structure}%
           {\tiny\quad \insertshorttitle, \insertshortauthor}
```

On peut surcharger un thème clés en main :

```
\usetheme{Warsaw}
\usecolortheme{crane}
\setbeamerfont{frametitle}{size=\Large,series=\bfseries}
```

La documentation de **beamer** expose en détail les thèmes possibles.

Le dossier `texmf-dist/tex/latex/beamer/themes/` contient tous les thèmes.

➔ Possible de voir comment ils sont construits pour d'éventuelles modifications.

## Au delà de la présentation

Où l'on découvre qu'il est aussi possible de générer des notes écrites avec le texte de la présentation ou bien que l'on peut composer un poster à l'aide de diapositives...

## La notion de mode

**beamer** définit 5 modes pour la création d'un document :

**beamer** est le mode par défaut, il correspond à la présentation.

**second** est le mode utilisé pour une diapositive sur un second écran.

**handout** est une version de la présentation où les pauses sont supprimées.

➔ Possibilité de proposer une version papier et un poster avec **pdfpages**...

**trans** est le mode de création de transparents (pauses supprimées).

➔ Impression sur transparents, prévoir thème adapté.

**article** est un mode où la classe du document n'est pas **beamer**.

➔ Version possiblement élargie de la présentation, avec une autre mise en page.

Le mode **all** les représente tous, le mode **presentation** exclut **article**.

`\mode<mode>{texte}` ne compose *texte* que dans le mode sélectionné.

➔ On peut donc ajouter sélectivement du texte à un des modes.

➔ *texte* peut contenir n'importe quel code **L<sup>A</sup>T<sub>E</sub>X** : commandes, *packages*...

Certaines commandes acceptent une spécification de mode :

`\section<presentation>{Nom de la section pour les présentations}`

`\section<article>{Nom de la section pour l'article}`

La syntaxe `<>` utilisée pour les pauses accepte aussi les modes :

`\uncover<beamer:2-|article|handout>{texte}`

➔ le texte est toujours présent en modes **article** et **handout**.

# Préparer la version article

On peut préparer une version article en même temps que la version présentation :

- Séparer le code source en plusieurs morceaux :
  - deux fichiers maîtres pour les modes `beamer` et `article` ;
  - le contenu de la présentation (`\begin{document} ... \end{document}`).  
→ Fichier `contenu.tex`.
- Baliser `contenu.tex` de commandes `\mode<article>{-}` et `\mode<beamer>{-}`.  
Si nécessaire, ajouter des modes dans les `\uncover`, `\only`, `\item...`

```
\documentclass{beamer}  
...  
\input{contenu.tex}
```

Fichier maître `beamer`

```
\documentclass[12pt]{article}  
\usepackage{beamerarticle}  
...  
\input{contenu.tex}
```

Fichier maître `article`

`beamerarticle` redéfinit une grande partie des commandes utilisées dans `beamer`.

⚠ Certains environnements ne sont pas redéfinis et sont seulement désactivés : `frame` et les colonnes par exemple.

Le fichier `contenu.tex` peut contenir plus : *packages*, définitions des commandes...

On peut prévoir d'autres versions, `handout` et `trans`, de la même façon.

# Transformer une présentation en poster

Choisir un thème avec seulement les titres des diapositives, pas de numéros de page...

Créer une version **handout** ➔ `handout.pdf`

```
\documentclass{article}
\usepackage[a1paper,margin=2cm]{geometry}
\usepackage{graphicx,xcolor}
\pagecolor{blue!10} % couleur de fond
\thispagestyle{empty}
% redéfinir la taille normale du texte

\begin{document}
% code pour le titre, le résumé, les logos

\hfill%
\includegraphics[width=0.45\textwidth,page=1]{handout.pdf}%
\hfill%
\includegraphics[width=0.45\textwidth,page=2]{handout.pdf}%
\hfill\hbox{}

...
\end{document}
```