

# Préparer une présentation simplement

md2pdf -p

**Bruno BEAUFILS**

**John MAC FARLANE**

**Leslie LAMPORT**

**Février 2021**



**Université  
de Lille**



1. Introduction
2. Faire une présentation avec `md2beamer`
3. Une courte référence

# 1. Introduction

## 2. Faire une présentation avec `md2beamer`

## 3. Une courte référence

## Un exemple

- ▶ Cette présentation est produite à partir de `modele-presentation.md`
- ▶ Elle est conçue pour des présentations simples

## Un modèle

- ▶ Elle peut constituer un *modèle* pour faire d'autres présentations :
  1. créer un nouveau dossier
  2. y copier, renommer puis **modifier** le fichier `modele-diaporama.md`
    - ▶ copier également les images (dossier `img`) si nécessaire
  3. créer la présentation via la commande  
`md2pdf nom-du-fichier.md`

## 1. Introduction

## 2. Faire une présentation avec `md2beamer`

## 3. Une courte référence

# Introduction

- ▶ **Markdown** est un langage de **balisage simple**
  - ▶ facile à lire et écrire par les humains et les programmes
  - ▶ qui permet de transformer un fichier balisé vers HTML ou PDF *automatiquement*
    - ▶ initialement pensé uniquement pour la transformation vers HTML
  - ▶ dont il existe plusieurs **variantes**
- ▶ **L<sup>A</sup>T<sub>E</sub>X** est un outil de composition de texte
- ▶ **Beamer** est un paquet L<sup>A</sup>T<sub>E</sub>X pour préparer des présentations
- ▶ **Pandoc** est un logiciel de conversion de documents
  - ▶ il permet de convertir des fichiers d'un format de balisage à un autre
  - ▶ il utilise notamment L<sup>A</sup>T<sub>E</sub>X et Beamer pour la production de présentation
- ▶ **md2pdf** est un simple script shell qui *enveloppe* un appel à pandoc
  - ▶ permet la conversion d'un fichier Markdown en PDF (présentation ou rapport)
  - ▶ paramètre `pandoc` pour personnaliser la conversion

# Utiliser md2pdf pour faire une présentation

## ► Conseils

1. construire la **structure (plan)** de la présentation en premier
2. remplir le contenu ensuite sans trop surcharger le texte
3. ne pas s'occuper de la mise en forme (md2pdf s'en occupe)
4. avoir un peu de **discipline**
  - être attentif aux **indentations** des lignes
  - espacer les sections de diapositives
  - préférer les images vectorielles (SVG) aux matricielles
  - préférer le format PNG pour les images matricielles

## ► Avantages

- construction **très rapide** de présentation propre
- style (*forme*) déporté vers le thème
- rapide à prendre en main
- sources réutilisables ailleurs

## ► Inconvénients

- pas de constructions complexes (solution : utiliser  $\text{\LaTeX}$ /Beamer directement)
- pas d'utilisation simple de caractères unicode exotique
- difficile à déboguer sans maîtriser la chaîne pandoc/pdflatex
  - passer par  $\text{\LaTeX}$  via peut aider (md2pdf -L)

# Utiliser md2pdf pour faire une présentation

## ► Conseils

1. construire la **structure (plan)** de la présentation en premier
2. remplir le contenu ensuite sans trop surcharger le texte
3. ne pas s'occuper de la mise en forme (md2pdf s'en occupe)
4. avoir un peu de **discipline**
  - être attentif aux **indentations** des lignes
  - espacer les sections de diapositives
  - préférer les images vectorielles (SVG) aux matricielles
  - préférer le format PNG pour les images matricielles

## ► Avantages

- construction **très rapide** de présentation propre
- style (*forme*) déporté vers le thème
- rapide à prendre en main
- sources réutilisables ailleurs

## ► Inconvénients

- pas de constructions complexes (solution : utiliser  $\text{\LaTeX}$ /Beamer directement)
- pas d'utilisation simple de caractères unicode exotique
- difficile à déboguer sans maîtriser la chaîne `pandoc/pdflatex`
  - passer par  $\text{\LaTeX}$  via peut aider (`md2pdf -L`)



## 1. Introduction

## 2. Faire une présentation avec md2beamer

## 3. Une courte référence

# Markdown

La syntaxe de Markdown permet

- ▶ de donner une **structure** au document
  - ▶ **paragraphes** : séparation par des lignes vides
  - ▶ **titres de différents niveaux** : lignes débutant par des #
  - ▶ **listes libres** : indentation de paragraphe avec lignes débutant par des – ou des \*
  - ▶ **listes ordonnées** : indentation avec lignes débutant par des 1 .
  - ▶ **blocs de citations** : paragraphes préfixés par “>”
  - ▶ **blocs de codes** : groupes de paragraphes entourés de lignes vides **et**
    - ▶ indentés par 4 espaces
    - ▶ ou **encadrés** par des lignes avec 3 accent graves (*backquotes*) ````` ou tildes `~~~`
- ▶ d'inclure des éléments de mise en forme logique  
*emphase*, **accentuation forte** et code
- ▶ d'inclure des liens vers des URL
  - ▶ <https://daringfireball.net/projects/markdown>
  - ▶ Pandoc Markdown
- ▶ d'inclure des images via des liens spécifiques :  
! [Texte alternatif] (img/tex-friendly-zone.svg)

Elle est décrite en détails ailleurs (cf liens de cette diapo)

# Extensions Pandoc à Markdown

Pandoc ajoute des fonctionnalités à la syntaxe Markdown

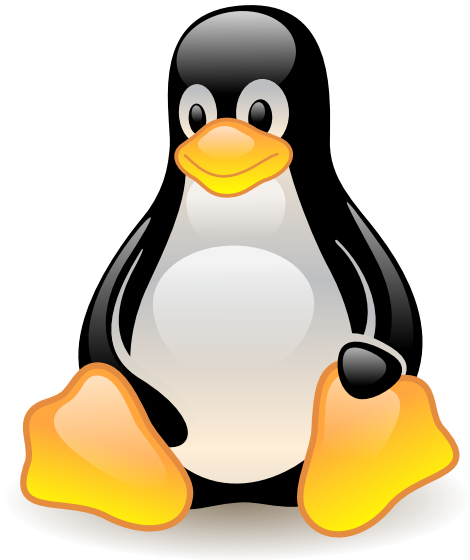
- ▶ Précision sur la nature des éléments
  - ▶ zone entourée d'accolades { . . . } juste après l'élément à qualifier
  - ▶ attribut=valeur
  - ▶ .classe
- ▶ Philosophie un peu étendue
  - ▶ transformation vers plein de formats différents (pas seulement HTML)
- ▶ Détails dans la doc de la [variante Pandoc de Markdown](#)

- ▶ format Markdown :  
! [TeX Zone] (img/tex-friendly-zone.svg){ width=30% .center }\



La séquence \ en fin de ligne est utilisée pour éviter le traitement en figure

- ▶ on peut spécifier une largeur ou une hauteur avec l'attribut `width` ou `height`
  - ▶ relative à la largeur ou hauteur de la diapo
- ▶ on peut centrer horizontalement avec la classe `.center`
- ▶ conversion automatique des références `.svg` en `.pdf` (qui doivent exister)



Une diapo nommé `{.plain}` n'a ni titre, ni pied de page

- ▶ Plusieurs types de support des tableaux
  - ▶ exemple avec le mode `pipe_tables` de `pandoc`

| Outil                 | Utilité  |
|-----------------------|--|
| <code>pandoc</code>   | conversion de formats  |
| <code>pdflatex</code> | composition de $\text{\LaTeX}$ en PDF                                |
| <code>beamer</code>   | jeux de balisage $\text{\LaTeX}$ pour la composition de présentation |

- ▶ [pandoc#tables](#) liste les autres manières de saisir des tableaux

# Code source

Le code source peut être coloré en fonction de la syntaxe du langage

```
#include <stdio.h>
int main(int argc, char ** argv) {
    printf("Hello world !\n");
}
```

Il peut être inclus depuis un fichier (grâce au filtre [include-code-files](#))

```
#!/bin/bash
for a in "$@"
do
    echo $a
done
```

Les divisions sont des blocs entourés par `:::`.

Elles peuvent être centrées en leur attribuant la classe `center`.

Elles peuvent contenir plusieurs paragraphes ou autres éléments.

| Outil    | Utilité                               |
|----------|---------------------------------------|
| pandoc   | conversion de formats                 |
| pdflatex | composition de $\text{\LaTeX}$ en PDF |
| beamer   | jeux de balisage $\text{\LaTeX}$ pour |



# Spécificité pour les présentations

## Listes

- ▶ imbrication de listes limitée à une profondeur de 3

## Affichage incrémental

- ▶ affichage incrémental

# Spécificité pour les présentations

## Listes

- ▶ imbrication de listes limitée à une profondeur de 3

## Affichage incrémental

- ▶ affichage incrémental
  - ▶ pause via un paragraphe avec une ligne de trois points  
...
    - ▶ des items de listes dans des blocs avec la classe `incremental`

```
::: {.incremental}
- un
- deux
:::
```
  - ▶ ne pas trop en abuser !

# Spécificité pour les présentations

## Listes

- ▶ imbrication de listes limitée à une profondeur de 3

## Affichage incrémental

- ▶ affichage incrémental
  - ▶ pause via un paragraphe avec une ligne de trois points  
.  
.  
.  
    - ▶ des items de listes dans des blocs avec la classe `incremental`

```
::: {.incremental}
- un
- deux
:::
```
- ▶ ne pas trop en abuser !

# Spécificité pour les présentations

## Listes

- ▶ imbrication de listes limitée à une profondeur de 3

## Affichage incrémental

- ▶ affichage incrémental
  - ▶ pause via un paragraphe avec une ligne de trois points  
.  
. . .
  - ▶ des items de listes dans des blocs avec la classe `incremental`

```
::: {.incremental}  
- un  
- deux  
:::
```

- ▶ ne pas trop en abuser !

# Spécificité pour les présentations

## Listes

- ▶ imbrication de listes limitée à une profondeur de 3

## Affichage incrémental

- ▶ affichage incrémental
  - ▶ pause via un paragraphe avec une ligne de trois points  
.  
. . .
  - ▶ des items de listes dans des blocs avec la classe `incremental`

```
::: {.incremental}
- un
- deux
:::
```
- ▶ ne pas trop en abuser !

## Les diapositives sont les éléments de niveau 2

- ▶ niveaux supérieurs sont les sections du diaporama
- ▶ niveaux inférieurs sont des blocs
- ▶ modifiable via
  - ▶ option `-P --slide-level=N` passée à md2pdf
  - ▶ la variable d'environnement `$SLIDELEVEL`
  - ▶ variable `slide-level` dans le document

## Transformation avec un *template* ad-hoc

- ▶ variables prises en charge
  - ▶ `title`, `subtitle`, `author`, `date`
  - ▶ `titlegraphic` : images (type logos) sur la page de titre
    - ▶ `file` : fichier sans extension
    - ▶ `width` : largeur en pourcentage (<1) de largeur diapo (0.1 par défaut)
    - ▶ `nl` : passage à la ligne après le logo
  - ▶ `titlebackground` : image de fond de la page de titre (fichier sans extension)
  - ▶ `theme`, `themeoptions` : thème et options du thème Beamer
  - ▶ `toc`, `toc-title` : générer une diapo de plan

- ▶ plusieurs thèmes inclus
  - ▶ `md2pdf -t list`
- ▶ autres thèmes utilisables
  - ▶ doivent être accessibles (dossier courant ou sous-dossiers)
  - ▶ format Beamer (`themebeamerTHEME.sty`)
- ▶ choix du thème via
  - ▶ option `-t` passée à `md2pdf`
  - ▶ variable d'environnement `$THEME` et `$THEMEOPTIONS`
  - ▶ variables `theme` et `themeoptions` dans le document
- ▶ par défaut utilise le thème inclus `ulille`

## ► réglages md2pdf

- réglages pandoc utilisés visibles dans `etc/makefile-pandoc`
- ajustables via
  - l'option `-P` passée à `md2pdf`
  - la variable d'environnement `$EXTRAS`
  - bloc de variables **YAML** dans le source (*généralement en début de fichier*)
- template utilisé par défaut : `etc/pandoc-beamer.tex`

## ► documentation

- intégrée
  - `md2pdf -h`
  - `md2pdf --man`
- lire la section *Structuring the slide show* de `pandoc(1)` aide aussi