

LaTeX

Michel Meynard

UM

Université Montpellier

Introduction

- système de composition de documents créé par Leslie Lamport
- collection de macro-commandes destinées au processeur de texte TeX de Donald Knuth (boîtes)
- LaTeX : abréviation de Lamport TeX
- Dernière version majeure LaTeX2e en attendant LaTeX3
- NON WYSIWYG (What You See Is What You Get) :
 - le rédacteur décrit la structure **logique** du document
 - LaTeX met en page la structure physique
- de nombreux types de document (documentclass) : articles, livres, présentations, rapports, lettres, étiquettes, pochettes de disque compact, posters, cartes de visite ...
- succès de LaTeX :
 - articles, thèses scientifiques (mode mathématique)
 - moins de perte de temps pour la mise en page
 - uniformité des documents produits

Table des matières

- 1 Introduction
- 2 Exemple
- 3 Le langage LaTeX
- 4 Beamer
- 5 Divers
- 6 Introduction à TikZ

Exemple - 1

- 1 Editer le texte suivant et le sauver dans exemple1.tex

```
\documentclass[a4paper]{article}           % commentaire
\usepackage[utf8]{inputenc}              % si utf8
\usepackage[T1]{fontenc}                 % pour lettres accentuées
\usepackage[français]{babel}            % règles de césure française
\title{Mon premier document \LaTeX{}}\author{Michel Meynard}

\begin{document}
\maketitle
\section{Première section}
Voici un texte accentué en français ! on peut écrire des
formules mathématiques telles que :  $\forall m \in \mathbb{N}, m$ 
est un palindrome ssi  $\forall i \in [1, m], m[i] = m[m+1-i]$ .
\section{Utilisation de la bibliographie}
Dans son roman \cite{VER1875}, \textsc{Jules Vernes} ...
\bibliographystyle{plain-fr}
\bibliography{mabiblio}
\end{document}
```

Exemple - 2

- ② Compiler `exemple1.tex` : `latex exemple1.tex`
- ③ Visualiser le fichier (DeVice Independant) `exemple1.dvi` : `xdvi exemple1.dvi &`
- ④ transformer en fichier Postscript imprimable : `dvips exemple1.dvi`
- ⑤ imprimer le fichier Postscript : `lpr exemple1.ps`

Autre solution avec `pdflatex` qui fournit un fichier pdf : `pdflatex exemple1.tex`

Les commandes et environnements L^AT_EX

- les caractères spéciaux : `$#%&~_^{\}` échappés par antislash
- toute commande est préfixée par **antislash** : `\TeX` qui donne T_EX, `\today` qui donne 3 janvier 2017
- une commande peut avoir de 1 à 9 paramètres
 - le premier peut être optionnel (valeur par défaut)
 - les autres sont obligatoires !
- `\documentclass[a4paper]{article}` a deux paramètres dont le premier est optionnel (par défaut format letter size)
- un environnement E débute par `\begin{E}` et se termine par `\end{E}`; l'emboîtement doit être respecté
- `document`, `itemize`, `enumerate`, `description` sont des environnements très fréquents

La structure d'un fichier .tex

- ① `\documentclass[a4paper]{article}` % type de document
- ② `\usepackage[français]{babel}` % les extensions à utiliser
- ③ `\usepackage[utf8]{inputenc}` % pur l'utf8
- ④ `\usepackage[T1]{fontenc}` % etc.
- ⑤ `\title{Mon article} \author{Michel Meynard}` % l'en-tête
- ⑥ `\begin{document}\maketitle` % début du corps du document
- ⑦ `texte et commandes LaTeX ...`
- ⑧ `\end{document}` % fin du document

Les paquets

- Les paquets sont des **extensions** du système L^AT_EX original et permettent de définir de **nouvelles commandes** et **environnements**
- leur nombre étant gigantesque, ils sont réunis au sein du Comprehensive TeX Archive Network (CTAN) <http://www.ctan.org/>
- `babel` est un paquetage qui recense les règles typographiques de différentes langues dont le français
- `graphicx` est un paquetage qui permet d'insérer un fichier graphique dans un document
- `verbatim` est un paquetage qui permet d'insérer un fichier de code C++ dans un document; `listings` est une autre extension permettant la même chose !

Les classes de document

- book** se décompose en une page de garde, table des matières (tableofcontents), index, bibliographie, parties (part), chapitres (chapter), sections (section), sous-sections (subsection), sous-sous-sections (subsubsection)
- article** a son titre sur la même page que le début du texte qui doit être court (dizaine de pages)
- beamer** pour faire des présentations découpées en frame telles que celle-ci
- lettre** pour écrire des courriers en français : possibilité de courrier de masse ;

Les mathématiques

- pour insérer une formule mathématique dans du texte, il suffit d'encadrer la formule par dollar \$
- pour insérer une ou plusieurs lignes de math, il faut passer en mode math grâce à un encadrement par double dollar \$\$
- indice x_i x_i et exposant x^2 x^2
- fraction, racines carrées : $\frac{n!}{(n-p)!}$ $\sqrt{x+y}$
- lettres grecques : α β γ Γ
- des environnements d'équation, de systèmes d'équation, des symboles nombreux et variés !

Le texte et les paragraphes

- une phrase est composée de mots, de ponctuations, d'espaces
- les espaces multiples comptent pour 1
- un retour à la ligne dans le source ne sera pas pris en compte
- un double retour à la ligne ou `\\` signifient un retour en début de ligne
- un changement de paragraphe est réalisé par `\par`
- un tilde " " représente un espace insécable comme dans M.~Meynard qui produit M. Meynard
- la gestion de l'espacement avant et après les ponctuations est réalisée par L^AT_EX
- l'indentation de paragraphe (`\parindent`), l'espace vertical entre paragraphe (`\parskip`) et bien d'autres paramètres de mise en page sont bien entendu modifiables
- `\newpage` insère un saut de page

Les environnements courants - 1

Les listes

- itemize** liste à puce ou chaque nouvelle ligne est introduite par `\item`
 - description** dictionnaire où chaque mot est suivi de sa définition `\item[xyz] définition de xyz (nous y sommes actuellement)`
 - enumerate** liste à numéros
- Autres environnements utiles
- verbatim** pour mettre du texte non interprété par L^AT_EX
 - abstract** résumé d'un article imprimé au début

Les tableaux

Pour un tableau aligné à gauche (left), centré, aligné à droite :

```
\begin{tabular}{|l|c|r|}
\hline
colonne 1 & colonne 2 & colonne 3 \\
\hline
1.1 & 1.2 & 1.3 \\
\hline
\end{tabular}
```

colonne 1	colonne 2	colonne 3
1.1	1.2	1.3

Les références

Table 1: Mon tableau

a	b	c
---	---	---

Dans le tableau 1 page 17, on voit que ...

les commandes `label`, `ref`, `pageref` sont également valables pour référencer les chapitres, sections, graphiques, items.

La commande `\listoftables` permet de générer une liste des table généralement en fin d'ouvrage

Les flottants

Par défaut, T_EX place ses “boîtes” de gauche à droite puis de haut en bas. Les objets flottants vont aller se positionner à des endroits permettant une plus jolie mise en page. Ainsi les tableaux et les graphiques sont souvent des flottants. Il faut alors pouvoir s’y référer dans le corps du texte !

```
\begin{table}[htbp]
\caption{\label{adrsmontab} Mon tableau}
\begin{tabular}{|l|c|r|}
...
\end{tabular}
\end{table}
```

Dans le tableau~\ref{adrsmontab} page~\pageref{adrsmontab}, ...

Une double compilation est nécessaire pour les références arrières !

Graphiques - 1

L'inclusion de graphiques nécessite le paquetage `\usepackage{graphicx}`

Pour insérer : `\includegraphics{monimage}`

On distingue deux codages d'images utilisables :

- si vous compilez avec `latex`, seulement des capsules PostScript (avec l'extension `.eps`) ;
- si vous compilez avec `pdflatex`, des images PNG (extension `.png`), JPEG (extension `.jpg` ou `.jpeg`) ou des fichiers PDF (extension `.pdf`) mais pas d'images PostScript ; les images `.bmp` sont intégrables mais il faut en définir la hauteur et la largeur.

Généralement, on omettra l'extension du fichier lors de l'inclusion. Si le fichier existe dans plusieurs formats, le compilateur choisira le fichier qui lui convient.

La conversion d'une image matricielle (`bmp`, `jpg`, `png`) en un format PostScript encapsulé (`eps`) peut être réalisée par The Gimp. Pour le dessin vectoriel, on pourra utiliser `xfig` ou `inkscape`

Graphiques - 2

- Chemin d'accès (relatif) aux images : `\graphicspath{{Figures/}}`
- Zoomer l'image par une des options `largeur`, `hauteur`, `échelle` de `includegraphics` :

```
\includegraphics[width=5in]{maphoto}
\includegraphics[height=4cm]{exemple1}
\includegraphics[scale=0.25]{reduction}
```



Figures flottantes

Comme pour les tableaux, les graphiques peuvent flotter dans le document grâce à l'environnement `figure` :

```
\begin{figure}[htbp]
  \caption{\label{fig:ex1}Mon premier exemple en \LaTeX}
  \fbox{\includegraphics[width=1.5cm]{exemple1}}
\end{figure}
```

Dans la figure `\ref{fig:ex1}` page `\pageref{fig:ex1}`, on ...

Dans la figure 1 page 21, on voit le fichier `exemple1.pdf` résultant de la compilation par `latexpdf` du fichier `exemple1.tex`. La commande `\listoffigures` permet d'afficher la liste des figures flottantes.



L'environnement minipage

- L'environnement `minipage` permet d'effectuer une mise en page en plusieurs colonnes ;
- le premier paramètre optionnel précise le positionnement tandis que le second indique la largeur ;

Exemple du transparent précédent

```
\begin{minipage}[t]{6cm}
  \begin{figure}[htbp]
    \caption{\label{fig:ex1}Mon premier exemple en \LaTeX}
    \fbox{\includegraphics[height=1.5cm]{exemple1}}
  \end{figure}
\end{minipage}
```

```
\begin{minipage}[t]{5.5cm}
```

Dans la figure `\ref{fig:ex1}` page `\pageref{fig:ex1}`, on voit le fichier `\texttt{exemple1.pdf}` résultant de la ...

```
\end{minipage}
```

Bibliographie -1

- un fichier unique contenant toute la bibliographie, et auquel se réfèrent différents documents LaTeX
- description des ouvrages dans le fichier `mabiblio.bib`

```
@book{VER1875,
  author="Verne, Jules",
  title="Michel {Strogoff}",
  year="1875",
  publisher="Le livre de poche"
}
@article{MERQUI2007,
  author="Merchet, Jean-Dominique and Quinot, Paul",
  title="L'\e}lection dans le miroir des sondages",
  journal="Lib{\e}ration",
  number="21 f{\e}vrier",
  year="2007"
}
```

Bibliographie - 2

- pour citer une référence bibliographique on utilise `\cite{VER1875}`
- à la fin du document tex, on insère la bibliographie :


```
\bibliographystyle{plain-fr}
\bibliography{mabiblio}
```
- compiler une première fois le document LaTeX avec latex (repère les cite)
- puis bibtex, et enfin une ou deux compilations avec latex !

Index, liste des tables et des figures

- `\tableofcontents` génère la table des matières (book)
- `\listoftables` génère la liste des tables flottantes
- `\listoffigures` génère la liste des figures flottantes
- `\input{premierchapitre}` permet de découper un livre en plusieurs fichiers
- `\include{premierchapitre}` presque identique à `\input` sauf si `\includeonly{chapitre3, chapitre5}` dans le préambule
- on peut produire un index ou glossaire avec `makeindex`

Introduction à Beamer I

Paquetage permettant de réaliser une présentation (diaporama) et de produire un polycopié (*handout*) en utilisant *pdf_latex*.

① *spécifier le type de production*

```
\documentclass{beamer} % cette ligne pour transp. ou 3 suiv
```

```
% \documentclass[handout]{beamer} % polycopié
% \usepackage{pgfpages}
% \pgfpagesuselayout{4 on 1}[a4paper,border shrink=5mm,
landscape] % 4 par 4
```

② *indiquer la mise en page ...*

```
\usetheme[secheader]{Madrid} % plusieurs thèmes possibles
\useoutertheme{infolines} % ligne d'en-tête et de pied
```

③ *autres paquetages, titre, auteur,...*

Introduction à Beamer II

```
\usepackage[français]{babel} ...
```

④ *premier transparent de titre général*

```
\begin{document}
\begin{frame}
\titlepage % Premier transparent de titre
\end{frame}
```

⑤ *second transp. : table des matières*

```
\begin{frame} % 2eme transparent TDM générale
\frametitle{Table des matières}
\tableofcontents[hideallsubsections] %ou [pausesections]
\end{frame}
```

⑥ *Sections*

```
\section{Introduction}
\subsection{Graphiques}
```

Introduction à Beamer III

⑦ les trames et les enchaînements de transparents ...

```

\begin{frame}[fragile]{Mon titre de trame}
\begin{itemize}[<+| alert@>]
\item un qui apparaîtra en premier
\item deux qui apparaîtra ensuite
\end{itemize}
\end{frame}
...
\end{document}

```

Frame, slide

frame une trame (*frame*) est un modèle pour un ou plusieurs transparents (slide) qui peuvent être découverts progressivement ;

slide transparent qui peut se superposer à d'autres grâce aux commandes de superposition telle que `\uncover<1,2>{Text}` : du slide 1 au 2.

Exemple `\begin{itemize}[<+| alert@>]`
`\item premier point visible du début à la fin et en rouge sur le 1er slide`

commandes de superposition le plus simple est d'indiquer le type de superposition et de mise en exergue (alert) pour un environnement comme ci-dessus ; mais on peut aller plus loin (uncover, only, ...)

superposition pour les nuls En indiquant l'option `<+| alert@>` :
`\begin{frame}[<+| alert@>][fragile]{Frame, slide}`

Notes pour aider le conférencier

Dans le corps du document :

- `\note{Bonjour, Mesdames et Messieurs, je vous ...}`
- ou bien :
`\note[item]{En ce qui concerne le premier point, ...}` ce qui générera les notes dans une liste à puces.

Si des notes sont dans une trame, celles-ci s'accumulent sur une page de note spécifique après la trame courante ; sinon, chaque note résidera sur une page propre !

Dans l'en-tête du document :

- `\setbeameroption{hide notes}` : les notes n'apparaissent pas dans le fichier PDF ;
- `\setbeameroption{show notes}` : les notes apparaissent dans le fichier PDF, entre les diapositives ;
- `\setbeameroption{show only notes}` : le fichiers PDF ne comprend que les notes.

FAQ

verbatim l'utilisation de texte *verbatim* nécessite l'option `fragile`
`\begin{frame}[fragile]` ; l'option `containsverbatim` n'est plus documenté mais fonctionne mieux !

numéroter tables et figures `\setbeamertemplate{caption}[numbered]`
titre de trame soit par `\frametitle{Titre}` mais aussi par `\begin{frame}{Titre}`.

1 trame sur plusieurs pages l'option `allowframebreaks` permet de définir une trame qui tiendra sur plusieurs pages avec le titre de trame qui sera suffixé de nombres romains (comme dans Introduction à Beamer) ; dans ce cas, la **superposition n'est plus possible**.

```
\begin{frame}[fragile,allowframebreaks]{Intro...}
```

verbatim et superposition L'utilisation de `verbatim` fait parfois planter la superposition (comme dans cette FAQ) !

Production de supports

Beamer permet de produire différents documents :

- diaporama** présentation = suite de transparents ;
- polycopié** à distribuer aux spectateurs ;
- notes** de lecture pour le locuteur ;
- transparentes physiques** version de secours si vidéo-projecteur en panne ;
- article** une version papier sous forme d'article différente du polycopié ;

Transitions

Beamer permet de réaliser des transitions entre transparents seulement visibles en mode plein écran :

- `\transdissolve<1>`
- `\transblindshorizontal<2>`
- `\transblindsvvertical<3>`
- `\transboxin<4>`
- `\transboxout<5>`
- `\transglitter<6>`

Organisation des fichiers

Afin de créer le diaporama ou bien le polycopié, on créera 3 fichiers :

- **diapo.tex** : à compiler pour produire le diaporama ;
`\documentclass{beamer}`
`\setbeameroption{hide notes}`
`\input{contenu}`
- **poly.tex** : à compiler pour produire le polycopié ;
`\documentclass[handout]{beamer}`
`\usepackage{pgfpages}`
`\pgfpagesuselayout{4 on 1}[a4paper,border shrink=5mm, landscape]`
`\setbeameroption{hide notes}`
`\input{contenu}`
- **contenu.tex** : en-tête, sections et trames : le contenu.
- **notes.tex** : à compiler pour produire les notes de lecture ;
`\documentclass{beamer}`
`\setbeameroption{show only notes}`
`\input{contenu}`

Outils

outils d'édition/compilation IDE

- **texmaker** pour l'édition, la composition et la prévisualisation sous Linux, Mac, Windows ;
- **mode auctex** sous emacs ;
- **sharelatex** sur le web ;

Autres outils

- **LyX** du LaTeX en WYSIWYG (bof)
- **inkscape** dessin vectoriel en SVG
- **xfig** sous linux, dia, WinFig sous windows : dessin vectoriel avec polices LaTeX

Références

LaTeX <http://fr.wikibooks.org/wiki/LaTeX>

CTAN <http://www.ctan.org/search/>

La bible 1 LaTeX, Leslie Lamport, Addison-Wesley

La bible 2 LaTeX companion, Goossens Mittelbach Samarin, Addison-Wesley

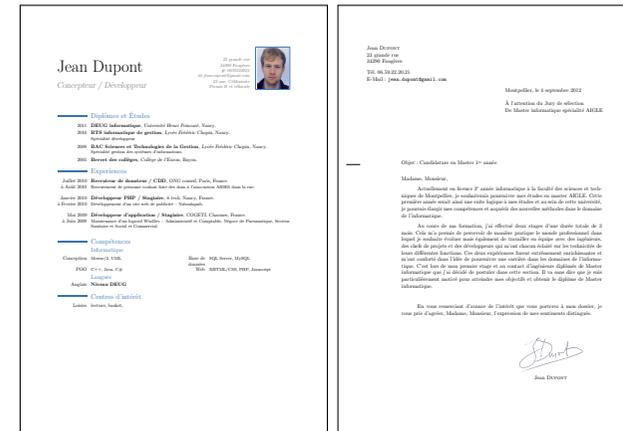
beamer la doc. : beameruserguide.pdf

Astuces

- Error Latex TeX capacity exceeded : souvent lié à un oubli de paramètre obligatoire d'une macro, par exemple titre d'un environnement de bloc block ;
-

Exercice : CV et lettre de motivation

- réaliser un curriculum vitae qui vous sera utile lors de la recherche de stage et d'emploi ;
- produire une lettre de motivation pour l'entrée en Master 1^{re} année.



Qu'est-ce que TikZ

- “TikZ ist kein Zeichenprogramm” TikZ n'est pas un programme de dessin !
- écrit par Till Tantau, l'auteur de Beamer
- repose sur une sous-couche PGF “portable graphics format”
- `\usepackage{tikz}` dans le préambule du document L^AT_EX
- `\draw` est obtenu par `\tikz \draw (0pt,0pt) -- (20pt,6pt);`
- ou bien dans un environnement : `\begin{tikzpicture}`
liste de commandes finissant par un `;`
`\end{tikzpicture}`

Qu'est-ce que TikZ

compléments

- extensible avec des bibliothèques :
`\usetikzlibrary{shapes,positioning,graphs,calc,chains,arrows,through}`
- des paquetages latex utilisant TikZ :
`\usepackage{tikz-qtrees}`

Coordonnées cartésiennes, polaires

un point peut être représenté :

- coord. cart. par deux dimensions T_EX : (0cm,5pt) ou (0cm,5em)
- par défaut, les dimensions sont 1cm vers la droite, 1cm vers le haut

`\tikz \draw (0,0) -- (1,1);`
- coord. pol. par un angle et une longueur : (60:10pt)

`\tikz \draw (0,0) -- (60:1) -- (120:1) -- (0,0);`
- coordonnées cart. relatives au dernier point utilisé en préfixant par ++

`\tikz \draw (0,0) -- ++(.5,.5) -- ++(-.5,.5) -- ++(-.5,-.5) -- (0,0);`
- coordonnées 3-dimensions (x,y,z); ancre relative à une position dans une forme; ...

Chemins (path) I

Un chemin est une suite de points reliés par des lignes droites ou courbes.

Revenir au point initial ne ferme pas un chemin : utiliser `-cycle`

- on le définit avec la commande `\path` et on agit dessus avec les options : `draw`, `fill`, `clip`, `shade`

`\tikz \path[draw,fill,shade] (0,0) -- (45:.5) -- (90:.5) -- (135:.5) -- cycle;`
`% cycle indique la fermeture du chemin`
- la commande `\draw` n'est qu'une abréviation de `\path[draw]` !
- on peut définir des paramètres graphiques en option :

`\tikz \draw[line width=2pt,color=red] (1,0) -- (0,0) -- (0,1) -- cycle;`

Chemins (path) II

- courbes de béziers .. avec 2 points de contrôle

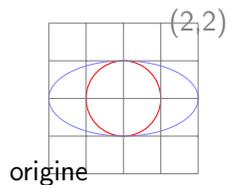

`\tikz \path[fill,color=orange] (0,0) .. controls (1,1) and (2,-2) .. (3,.5) --cycle;`

- angles droits (`-|` `|-`), rectangle, arc (ang.dép. :ang.arriv. :rayon)


`\tikz \path[fill,color=green] (0,0) -| (1,1) rectangle (2,2) arc (0:180:1) --cycle;`

Chemins (path) III

- cercles, ellipses, grille, noeud



```
\begin{tikzpicture}
\draw[red] (1,1) circle (5mm);
\draw[blue!50] (1,1) ellipse (1cm and 5mm);
\draw[black!50,thin] (0,0) grid[step=5mm]
(2,2) node{(2,2)};
\draw (0,0) node {origine};
\end{tikzpicture}
```

- décalage vertical : pour simplifier, on peut indiquer les coordonnées par rapport à l'origine et décaler tout le chemin d'un espace vertical : `\path[yshift=1cm] (0,0) ...`

Noeud et arête (node and edge)

- Un noeud est une forme contenant du texte (sommet d'un graphe, noeud d'un arbre, étiquette d'un arc ...);
- Il est spécifié comme une opération de chemin MAIS il est indépendant du chemin ;


```
\path ... node<foreach statements>[<options>] (<name>) at (<coordinate>) {<node text>} ...;
```
- seuls `node` et le texte du noeud sont obligatoires!
- le nom d'un noeud est utile afin de pouvoir réutiliser ce noeud ;
- l'opération `foreach` permet de répéter des noeuds :


```
1cm 2cm 3cm
\tikz \draw (0,0) node foreach \x in {1,2,3} at (\x,0)
{\x cm};
```

Noeuds, formes, ancres, positionnement

- forme (shape) pour l'afficher; option de `tikzpicture`

```
[every node/.style={draw}]
```

```
\begin{tikzpicture}[every node/.style={draw}]
\draw[shape=rectangle] (0,0) node(orig)
{0} -- (1,1) [shape=circle]node{(1,1)};
\end{tikzpicture}
```

- Un noeud possède des ancres : `north west`, `north`, `north east`, `west`, `center`, `east`, `south west`, ... Par défaut, l' ancre `center` est placée sur le point du chemin, mais on peut ancrer ailleurs.

```
\draw[shape=rectangle] (0,0) node[anchor=north
east](orig){0} -- (1,.5) [shape=circle,
anchor=south west]node{(1,1)};
```

Positionnement

Plutôt que d'ancrer le noeud, on peut définir sa position relativement à un point mais aussi à un autre noeud en utilisant la bibliothèque `positioning` (`above`, `above right`, `left`, `below left`) :

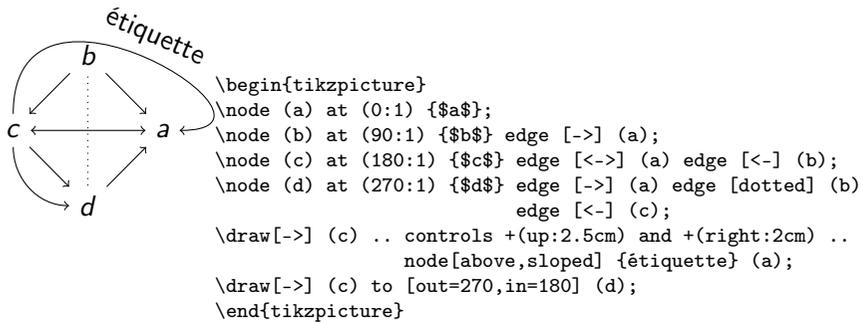
- position relative à un point :


```
\tikz \fill (0,0) circle (2pt) node[above
above right right] {above right};
```
- position relative à un autre noeud : utilisation de `/tikz/node distance = 1cm and 1cm (défaut)`

```
\begin{tikzpicture}[every node/.style={draw,rectangle}]
\draw[help lines] (0,0) grid (2,3);
\begin{scope}[node distance=5mm and 5mm]
\node (b) at (1,2) {b};\node [left-of b] {1};
\node [right-of b] {2};\node [above-of b] {3};
\node [above left-of b] {5}; \node [above right-of b] {6};
\end{scope}\begin{scope}[node distance=5mm]
\node (a) at (1,1) {a};\node [left-of a] {1};
\node [right-of a] {2};\node [below-of a] {4};
\node [below left-of a] {7}; \node [below right-of a] {8};
\end{scope}
\end{tikzpicture}
```

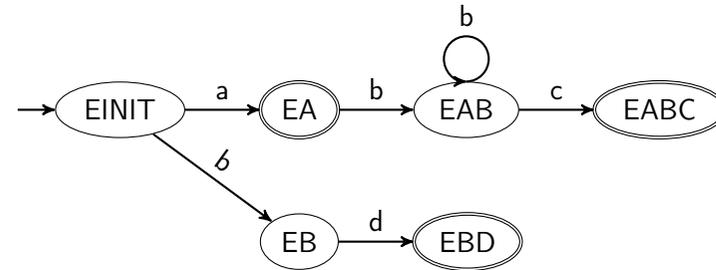
opérations to et edge

- les segments entre noeuds permettent de s'arrêter sur la forme !
- to et edge sont des opérations permettant de tracer des arêtes :



- Chaque edge peut avoir son propre style, alors que les segments d'un chemin le partagent !

Exemple d'automate

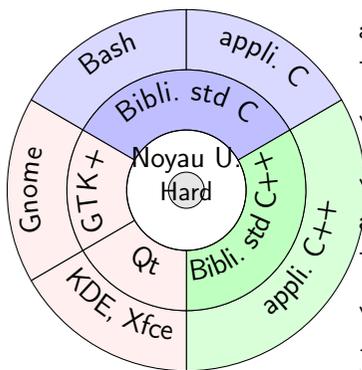


```

\tikzstyle{sommet}=[ellipse,draw] % état
\tikzstyle[final]=[ellipse,double,draw] % état final double cerclé
\tikzstyle[arc]=[->,>=stealth',thick] % transition flèche grasse
\tikzstyle{etiquette}=[sloped,midway,above] % incliné milieu centré
\begin{tikzpicture}
\node[sommet] (einit) at (0,0)[right]{EINIT}; % placé à droite
\draw[arc] (-.5,0) -- (einit); % flèche d'état initial
\node[final] (ea) [right= of einit] {EA}; % ea à droite de einit (right=)
\draw[arc] (einit) -- (ea) node[etiquette]{a}; % arc étiqueté
\node[sommet] (eab) [right= of ea] {EAB};
\draw[arc] (ea) -- (eab) node[etiquette]{b}; % arc étiqueté

```

Cercles concentriques

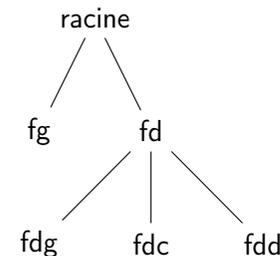


```

\draw[thin,fill=blue!15] (150:2) arc(150:90:2)
-- (90:3) arc(90:150:3) -- cycle; % portion
\draw (150:2.5) decorate [decoration={text
along path,text align=center, text=Bash}]
{arc(150:90:2.5)}; % texte
\draw[thin,fill=blue!15] (90:2) arc(90:30:2) --
(30:3) arc(30:90:3) -- cycle; % portion
\draw (90:2.5) decorate [decoration={text
along path,text align=center, text=appli. C}]
{arc(90:30:2.5)}; % texte
\draw[thin,fill=green!15] (-90:2) arc(-90:30:2)
-- (30:3) arc(30:-90:3) -- cycle; % portion
\draw (-90:2.5) decorate [decoration={text
along path,text align=center, text= appli. C++}]
{arc(-90:30:2.5)}; % texte

```

Arbre



```

\begin{tikzpicture}
\node at (0,0) {racine}
child {node {fg}}
child {node {fd}
child {node {fdg}}
child {node {fdc}}
child {node {fdd}}
};
\end{tikzpicture}

```