

Formation à L^AT_EX
Télécom Paris
Jeudi 4 mars 2005

`bruno@enix.org`

Partie 1

Un exemple de session

L_AT_EX

1.1 — Création du texte source

Créez le document avec Emacs (ou tout autre éditeur):

```
\documentclass{article}
\begin{document}
C'est juste un essai!
\end{document}
```

Choisissez un nom de fichier de la forme `toto.tex`.

1.2 — Compilation

Ouvrez une console; passez la commande `latex toto`.

Observez que les fichiers `toto.dvi` et `toto.aux` ont été créés.

1.3 — Visualisation

Visualisez avec `xdvi toto`. Jouez avec la loupe (boutons de la souris), appréciez la finesse des caractères

Variante: recompilez avec `pdflatex toto`, puis visualisez avec `xpdf toto`.

1.4 — Impression (au besoin)

Méthode 1: imprimez depuis `xdvi`

Méthode 2: `dvips toto` ou `dvips toto -o toto.ps` puis jetez le fichier PS sur une imprimante PostScript

Partie 2

Fonctionnement

2.1 — Saisie du texte source

Le texte source peut être saisi avec n'importe quel éditeur (évitiez toutefois Notepad, de chez Bilou).

Une fin de paragraphe doit être signalée par `\`, ou en sautant une ligne.

Attention: plusieurs espaces consécutifs sont considérés comme un seul espace! Vous n'êtes pas face à une machine à écrire.

De même, plusieurs lignes vides entre deux paragraphes sont considérées comme une seule.

2.2 — Saisie des lettres accentuées (1)

Avec le paquetage `inputenc`, on saisit les lettres accentuées directement au clavier. Choisir l'option `latin1` sur une machine Unix (ceci inclut Linux et Mac OS X), `applemac` sur un Mac sous Mac OS 9 ou `ansinew` sur une machine Windows.

Avantage: plus confortable, on voit les éèçàù et consorts dans le texte source.

Inconvénient: non portable.

2.3 — Saisie des lettres accentuées (2)

Solution portable:

```
\‘{a} \’{a} \^{a} \"{a} \~{a} \={a} \.{a}  
\u{a} \v{a} \H{a} \t{aa} \c{a} \d{a} \b{a}
```

On obtient à á â ä ã ā à ã ă á âa ą ą ą.

On peut poser n’importe lequel des quatorze accents sur n’importe quel caractère: ò ó ô õ ö ÿ Ź Ź Ī Ÿ ă ąp ą ą \$.

2.4 — Caractères spéciaux (1)

Beaucoup de caractères jouent un rôle spécial: nous avons déjà vu \backslash , $\}$ et $\{$.

De même $\$$, $\#$, $\%$ et $\&$ sont des métacaractères.

Pour obtenir ces derniers: $\backslash\$$, $\backslash\#$, $\backslash\%$ et $\backslash\&$.

Le caractère $_$ sert pour indiquer un indice dans les formules; pour faire apparaître ce caractère, on utilisera $\backslash_$.

2.5 — Caractères spéciaux (2)

Le caractère ~ sert à indiquer un *espace insécable*. Exemples d'utilisations:

Louis~XIV il est né le 2~mars mise à~jour

Le caractère % sert à placer des commentaires: ce caractère et tous ceux qui le suivent sur **la même ligne** sont ignorés.

Utiliser \~ et \% pour avoir ces caractères tels quels!

2.6 — Rôle des balises

Les balises (ou *macros*) sont des noms de la forme `\toto`, `\a`, `\LaBaliseEnCarton`. Donc: le caractère *backslash*, puis une suite de lettres.

`\toto` et `\Toto` sont deux balises différentes.

Dans le texte source, les espaces qui suivent une balise sont mangés. Donc on écrira `Mon {\toto}` est plus beau que le vôtre. Variante: `Mon \toto\` est plus beau que le vôtre.

2.7 — Compilation, le fichier LOG

La compilation (ou composition) active T_EX proprement dit, c'est-à-dire le moteur de composition typographique.

Elle peut être lancée depuis l'environnement de travail (cf. vue suivante), ou depuis la console: `latex mon-document`.

La composition produit un fichier intermédiaire (DVI) ou final (PDF), ainsi qu'un fichier de compte-rendu (LOG), un fichier auxiliaire (AUX) et éventuellement d'autres fichiers annexes (table des matières, liste des figures, etc.)

2.8 — Le mode \LaTeX d'emacs

Ce mode est activé si l'on ouvre un fichier dont le nom se termine par `.tex`.

On lance la composition avec `Ctrl-C Ctrl-F`; on visualise (via `xdvi`) avec `Ctrl-C Ctrl-V`.

2.9 — Rôle du fichier dvi; alternative: pdflatex

Le fichier DVI (DeVice Independent) est structuré en pages.

Chaque page contient une liste de quadruplets (x, y, c, p) qui signifient: à l'abscisse x , à l'ordonnée y , placer le caractère c de la police p .

Avec `pdflatex`, on fabrique directement un fichier pdf. Avantage: le fichier contient les fontes requises, il peut être mis en ligne ou mailé, car il sera lisible sur n'importe quelle plate-forme.

2.10 — Visualisation & impression

Le fichier DVI peut être visualisé avec XDVI (sous Unix); avec YAP ou WinDvi ou DviWiIn (sous Windows); avec Textures Viewer sous Mac OS/9; avec Aperçu sous Mac OS/X.

Autres techniques: en produisant un PostScript avec `dvips`; en produisant directement un PDF avec `pdflatex`.

De chacune de ces méthodes, on déduit une technique d'impression.

Partie 3

Structure du texte source

3.1 — Préambule, corps du document

Le texte source a l'allure suivante:

```
\documentclass{nom-du-fichier-source}
```

Ici, le préambule

```
\begin{document}
```

Ici, le corps du document

```
\end{document}
```

3.2 — Classes

La première ligne indique la *classe* du document: *article*, *book*, *report* et *slides* sont les classes les plus courantes.

Utilisez la classe *article*!

On peut spécifier des *options de classe*:

```
\documentclass[a4paper,landscape,12pt]{article}
```

Montrer le fichier

`exemple-de-transparent.tex`

3.3 — Paquetages

En anglais, *packages*.

Chaque paquetage apporte des fonctionnalités supplémentaires, ou modifie le fonctionnement standard de L^AT_EX.

```
\usepackage{indentfirst}  
\usepackage[latin1]{inputenc}  
\usepackage[français]{babel}  
\usepackage{amsmath,amsfonts,amssymb}
```

3.4 — Définitions

Pour définir ses propres balises. Syntaxe:

```
\newcommand{nom-de-la-balise}{définition}
```

Exemples:

```
\newcommand{\Tcheb}{Tchebitcheff\xspace}
```

```
\newcommand{\NotaBene}[1]{\textbf{NB: } \textit{#1}}
```

La première balise évite d'avoir trois orthographes différentes du même nom dans un manuscrit. Elle requiert le paquetage

`xspace`. ?

Partie 4

Structuration du document

4.1 — Différences entre les classes

`article`: impression recto seul

`book`: impression recto-verso (marges)

`slides`: pour préparer des transparents; chaque vue est obtenue avec un environnement `slide`; ce que vous regardez en ce moment a été compilé avec la classe `slides`.

4.2 — Parties, chapitres, sections ...

La structuration utilise les balises `part`, `chapter`, `section`, `subsection`, `subsubsection` et `paragraph`.

Exemple:

```
\documentclass{book}
\begin{document}
\chapter{Rubrique culturelle}
\section{Les films du mois}
\subsection{21 grammes}
\subsection{35 tonnes}
\subsection[Anodin]{Partie classée X}
\subsection*{Ne sera pas numérotée}
\section{Notes de lecture}
\end{document}
```

4.3 — Listes numérotées

Avec l'environnement `enumerate`

```
\begin{enumerate}
  \item Des pommes
  \item Des poires
  \item Et des scoubidoubidous Ouah
\end{enumerate}
```

Les retraits devant les trois `\item` ne sont pas indispensables, mais améliorent nettement la lisibilité de votre source.

4.4 — Listes non numérotées

Avec l'environnement `itemize`

```
\begin{itemize}
  \item Producteur: Nicolas Puech
  \item Réalisateur: Bruno Petazzoni
  \item Accessoires: Donald, Tom, John...
\end{itemize}
```

4.5 — Descriptions

Avec l'environnement `description`

```
\begin{description}
  \item[Matériel] biprocesseur Céléron
  \item[Syntème] RedHat 7.3
  \item[Interface] KDE
\end{description}
```

`enumerate`, `itemize` et `description` peuvent être imbriqués jusqu'à quatre niveaux.

Montrer le fichier
exemple-listes

4.6 — Page de titre

Utiliser les balises `\title`, `\author`, `\date` (facultative) et `\maketitle`.

```
\title{Le mot de Kolakoski \\  
Développements récents et perspectives}  
\author{Maurice Craffougnat, Ph. D. \\ Hector Berlozzi, Prof.}  
\date{29 février 2004}  
\maketitle
```

Suggestion: placez `\thispagestyle{empty}` juste après `\maketitle` si vous voulez forcer le style `empty` pour votre page de titre.

4.7 — Résumé

L'environnement `abstract` permet de placer un résumé (en général, dans la première page, juste après le `\maketitle`).

```
\begin{abstract}
```

```
Dans cet article, nous nous intéressons à l'influence  
des rayons gamma sur la croissance des marguerites.
```

```
\end{abstract}
```

Cet environnement n'est pas disponible dans la classe `slides`. ?

Montrer le fichier
exemple-page-de-titre

4.8 — Notes de bas de page

Ce transparent* vous explique comment insérer une note de bas de page.

Ce transparent\footnote{Qui utilise intensivement l'autoréférence} vous explique comment insérer une note de bas de page.

Ceux qui n'ont pas compris que nous parlions des notes de bas de page viendront me voir pour une explication de gravure! Remarque: dans la classe `slides`, les notes ne sont pas numérotées[†].



*Qui utilise intensivement l'autoréférence

†Tiens donc!

Montrer le fichier

`exemple-note-de-bas-de-page`

4.9 — Table des matières

Placer la balise `\tableofcontents` à l'endroit où devra apparaître la table des matières.

En général, la table des matières n'est pas *au milieu* du document; l'usage actuel est de la mettre au début.

Compiler deux fois pour être certain que la table est à jour. ?

Ne pas oublier l'option de francisation (`[français]{babel}`).

Vérifier qu'il n'y a pas `\nofiles` dans le préambule!

Montrer le fichier
exemple-tdm.tex

Partie 5

**Présentation,
enrichissement**

5.1 — Géométrie de la page

Les options `landscape` et `a4paper` sont suffisantes en général. Mais si vous voulez occuper plus de surface:

```
\setlength{\oddsidemargin}{0 pt}
\setlength{\topmargin}{0 pt}
\setlength{\headheight}{0 pt}
\setlength{\headsep}{0 pt}
\setlength{\textwidth}{170 mm}
\setlength{\textheight}{240 mm}
```

5.2 — Style de la page

`\pagestyle` permet de définir le *style* des pages qui suivent. Les styles disponibles sont:

- `plain`: folio en pied de page;
- `empty`: pas de folio;
- `headings`: avec en-têtes;
- `myheadings`: en-têtes personnalisés.

`\thispagestyle` permet de définir le style de la page courante.

Montrer le fichier
exemple-myheadings

5.3 — Enrichissement typographique

Par défaut, les textes sont composés en caractères droits (romains).

Pour obtenir des caractères *italiques*: `\textit{italiques}`.

Pour obtenir des caractères **gras**: `\textbf{gras}`.

Pour obtenir des Petites Capitales: `\textsc{Petites Capitales}`;
remarque: elles ne sont pas accessibles dans la classe `slides` (ce qui rend la présente vue totalement ridicule).

Soulignement: on ne souligne pas. Vous n'utilisez pas une machine à écrire. ?

5.4 — L'utilisation de `emph`

`\emph` permet de passer du style droit (ou romain) au style italique et inversement. On comprendra mieux avec cet exemple:

En première page du `\emph{Guardian}`, nous lisons ceci:

```
‘‘\emph{The \emph{première} of Clint Eastwood’s picture’’.
```

Ce qui nous donne:

En première page du *Guardian*, nous lisons ceci: “*The première of Clint Eastwood’s picture*” .

5.5 — Taille des caractères

```
{\tiny Consultez un ophtalmo} \\
{\small Petits caract\`eres} \\
Taille normale \\
{\large Plus grand!} \\
{\Large Encore plus grand!!} \\
{\LARGE De plus en plus grand!!} \\
{\huge Gigantesque!!!} \\
{\Huge Kolossal!!!!} \\
```

5.6 — Taille des caractères: le résultat

Consultez un ophtalmo

Petits caractères

Taille normale

Plus grand!

Encore plus grand!!

De plus en plus grand!!

Gigantesque!!!

Kolossal!!!!

Partie 6

Mathématiques

6.1 — Formules en ligne

Une formule en ligne est délimitée par `\(` et `\)`, voire tout simplement par `$` et `$`. Elle est composée dans le paragraphe. Exemple:

Si `\(a\)` et `\(b\)` commutent, alors `\((a+b)^2=a^2+2a b+b^2\)`.

Nous obtenons:

Si a et b commutent, alors $(a + b)^2 = a^2 + 2ab + b^2$.

Remarques: l'exposant est marqué avec `^`; les espaces dans la formule sont ignorés; au besoin, la formule est coupée.

6.2 — Formules en exergue

Une formule en exergue est délimitée par `\[` et `\]`, voire tout simplement par `$$` et `$$`. Elle est composée dans un paragraphe centré. Exemple:

```
\[\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}\]
```

Nous obtenons:

$$\sum_{k=1}^{\infty} \frac{1}{k^2} = \frac{\pi^2}{6}$$

Remarques: `\sum`, `\infty`, `\frac`, `\pi`. L'indice est marqué avec

`_`.

6.3 — Opérations

$+$, $-$ et $/$ sont obtenus banalement.

$\backslash(E \times F \backslash)$ et $\backslash(s * t \backslash)$ donnent $E \times F$ et $s * t$.

$\backslash(A \cap B \backslash)$ et $\backslash(X \cup Y \backslash)$ donnent $A \cap B$ et $X \cup Y$.

6.4 — Opérateurs de relation

`<`, `>` et `=` sont obtenus banalement.

`\(a \neq b\)` donne $a \neq b$; notez l'espace après `\neq`.

`\(x \leq y\)` et `\(a \geq b\)` donnent $x \leq y$ et $a \geq b$.

Avec les paquetages `amsmath` et `amssymb`, `\(u \leqslant v\)` et `\(x \geqslant y\)` donnent $u \leqslant v$ et $x \geqslant y$.

`\(P \Rightarrow Q\)` et `\(A \iff B\)` donnent $P \Rightarrow Q$ et $A \iff B$.

`\(A \subset B\)` et `\(x \in E\)` donnent $A \subset B$ et $x \in E$. Pour obtenir le contraire, utiliser `\not`; ainsi, `\(Z \not\subset X\)` donne $Z \not\subset X$.

6.5 — Fonctions

Les fonctions usuelles sont prédéfinies:

`\sin(a-b)` donne $\sin(a - b)$

Comparez avec `\cos(a-b)`, qui donne $\cos(a - b)$.

Avec `\sin`, le nom de la fonction est composé en caractères romains (droits) et est suivi d'un espace (tout petit).

On peut définir des noms de fonctions pour ses propres besoins.

6.6 — Symboles divers

`\nabla` donne ∇ .

`\partial` donne ∂ .

`\emptyset` donne \emptyset .

`\forall` et `\exists` donnent \forall et \exists .

`\ell` donne ℓ ; comparer avec `\l` qui donne l .

6.7 — Symboles de taille variable

`sum`, `prod`, `int`, `bigcap`, `bigcup` voient leur taille varier selon le style de la formule.

`\(I = \int_0^1 f\)` donne $I = \int_0^1 f$.

`\(\displaystyle I = \int_0^1 f\)` donne $I = \int_0^1 f$.

`\(\sum_{k=1}^n k^2\)` donne $\sum_{k=1}^n k^2$.

`\(\displaystyle \sum_{k=1}^n k^2\)` donne $\sum_{k=1}^n k^2$.

6.8 — Chapeaux et consorts

`\(\overline{pqr}\)` donne \overline{pqr}

Truc technique: `\(\overline{a}-\overline{b}\)` donne $\overline{a}-\overline{b}$; mais
`\(\overline{\vphantom{b}a}-\overline{b}\)` donne $\overline{a}-\overline{b}$

`\(\overbrace{ax-by}\)` donne $\overbrace{ax-by}$

`\(\underbrace{a \dots z}_x\)` donne $\underbrace{a \dots z}_x$

`\(\sqrt{b^2-4ac}\)` et `\(\sqrt[n]{n!}\)` donnent $\sqrt{b^2-4ac}$ et $\sqrt[n]{n!}$

6.9 — Flèches

`\(\overrightarrow{gradient}\)` donne $\overrightarrow{gradient}$

`\(\vec u\)`, `\(\vec i\)` et `\(\vec{\imath}\)` donnent \vec{u} , \vec{i} et $\vec{\imath}$
`\(\imath\)` et `\(\jmath\)` fournissent \imath et \jmath

`\(\uparrow\)` et `\(\downarrow\)` donnent \uparrow et \downarrow
`\(\leftarrow\)` et `\(\rightarrow\)` donnent \leftarrow et \rightarrow

`\(x\to2\)` donne $x \rightarrow 2$

6.10 — Délimiteurs

$()$, $[]$, $| |$, $|| ||$, $|| ||$, $\{ \}$, $\lfloor \rfloor$ et $\lceil \rceil$ sont des paires de *délimiteurs*. Leur taille peut être adaptée au contexte, avec `\left` et `\right`.

`\(\lceil\frac{a}{x}\rceil\)` donne $\lceil \frac{a}{x} \rceil$

`\(\left\lfloor\frac{a}{x}\right\rfloor\)` donne $\left\lfloor \frac{a}{x} \right\rfloor$

Variante: `\(\Bigl|\frac{a}{x}\Bigr|\)` donne $\left\| \frac{a}{x} \right\|$

On dispose de `big`, `Big`, `bigg` et `Bigg`.

6.11 — Matrices

Pour obtenir $\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & x/2 \end{pmatrix}$, saisir:

```
\(  
  \begin{pmatrix}  
    1 & -1 & 0 \\  
    1 & 0 & x/2 \\  
  \end{pmatrix}  
\)
```

La présentation donne une idée du résultat, et permet d'éviter certaines erreurs. Requiert le paquetage `amsmath`. Variantes: `vmatrix`, `Vmatrix`, `bmatrix` et `Bmatrix`.

6.12 — Lettres grecques

Elles ne sont accessibles que dans les formules!

Facile: il y a une balise par lettre (minuscule ou majuscule), sauf pour celles qui existent dans l'alphabet latin.

Exemple: `\(\alpha\beta\Delta\Omega\)` donne $\alpha\beta\Delta\Omega$; si vous n'avez pas appris l'alphabet grec pendant vos années de prépa, il n'est pas encore trop tard pour vous cultiver!

Quelques gadgets: `\(\phi\)` donne ϕ ; je préfère `\(\varphi\)`, qui donne φ ; de même, `\(\epsilon\)` donne ϵ ; je préfère `\(\varepsilon\)`, qui donne ε

6.13 — Autres caractères particuliers

\mathbb{N} peut être obtenu avec `\(\mathbb{N}\)`; requiert `amsmath`.

$\mathcal{C}^\infty(\mathbb{R})$ peut être obtenu avec `\(\mathcal{C}^\infty(\mathbb{R})\)`.

Pour obtenir $\text{grad}(f)$, saisir `\(\mathop{\mathbf{grad}}(f)\)`.

Pour obtenir A^* , saisir `\(\mathsf{A}^*\)` (caractères *sans sérifs*).

Montrer le fichier

exemple-N

6.14 — Équations numérotées

L'environnement `equation` permet de numéroté certaines des équations d'un texte; ainsi:

$$(\pi + \xi)^n = \sum_{k=0}^n \binom{n}{k} \pi^k \xi^{n-k} \quad (1)$$

a été obtenue avec:

```
\begin{equation}
(\pi+\xi)^n = \sum_{k=0}^n \binom{n}{k} \pi^k \xi^{n-k}
\end{equation}
```

6.15 — Équations alignées et numérotées

J'utilise l'environnement `eqnarray`; il permet d'aligner et (au besoin) de numéroté une suite d'équations:

$$\begin{aligned}\mu(s_1) &= \gamma_{x_1} \star n_2 + n_2 \circ \psi^{u_2} \\ \Phi(s_3 \wedge q) &= n_1 \oplus \delta_{x_1} + n_a^k\end{aligned}$$

```
\begin{eqnarray*}
\mu(s_1) &= & \gamma_{x_1} \star n_2 + n_2 \circ \psi^{u_2} \\
\Phi(s_3 \wedge q) &= & n_1 \oplus \delta_{x_1} + n_a^k \\
\end{eqnarray*}
```

Partie 7

Tableaux & matrices

7.1 — Tableaux (1)

L'environnement `tabular` permet de définir un tableau; la structure générale est celle d'une matrice.

```
\begin{tabular}{|l|r|r|}  
  \hline  
  Article & Prix & Quantité \\  
  \hline  
  Assiette & 30 & 2.40 \\  
  \hline  
  Marmite & 8 & 12.80 \\  
  \hline  
\end{tabular}
```


7.2 — Tableaux (2)

Article	Prix	Quantité
Assiette	30	2.40
Marmite	8	12.80

Les éléments sont séparés par `&`; chaque ligne est terminée par `\\`. Le caractère `|` indique un filet vertical. `\hline` donne un filet horizontal.

On utilise les caractères `l`, `c` et `r` pour indiquer que les éléments de la colonne doivent être appuyés à gauche, centrés ou appuyés à droite.

7.3 — Tableaux (3)

L'environnement `array` donne les mêmes fonctionnalités, en mode mathématique.

Voici une matrice, avec un découpage en blocs.

$$\left(\begin{array}{cc|c} H_1 & a + b & 12.34 \\ \hline x_1 & c^2 & 2.40 \\ \frac{1}{k} & \sin(\alpha) & 12.80 \end{array} \right)$$

7.4 — Tableaux (4)

Voici comment obtenir la matrice bizarre de la vue précédente:

```
\(  
\left(  
  \begin{array}{lrr}  
    H_1 & a+b & 12.34 \\ \\  
    \hline  
    x_1 & c^2 & 2.40 \\ \\  
    \frac{1}{k} & \sin(\alpha) & 12.80 \\ \\  
  \end{array}  
\right)  
\)
```

7.5 — Cadres

`fbox` et `framebox` permettent d'encadrer des fragments de texte.

Exemple:

Je peux `\fbox{encadrer}` un mot. Je peux aussi indiquer la largeur de la `\framebox[35 mm][c]{boîte}`, et l'alignement.

Ce qui donne:

Je peux encadrer un mot. Je peux aussi indiquer la largeur de la boîte, et l'alignement (l, c ou r).

7.6 — Des cadres plus jolis

Avec le paquetage `fancybox`:

`\doublebox{double bordure}` donne `double bordure`

`\shadowbox{case ombrée}` donne `case ombrée`

`\ovalbox{bords arrondis}` donne `bords arrondis`

`\Ovalbox{bords arrondis épais}` donne `bords arrondis épais`

Partie 8

Figures et tables

8.1 — Éléments flottants

Les éléments flottants (figures et tables) ne doivent pas être coupés (répartis entre deux pages).

Le principe: s'il ne reste pas assez de place dans la page courante, ils seront placés au début de la page suivante.

On dispose des environnements `figure` et `table`. Ils ne peuvent pas être utilisés dans la classe `slides`. ?

On peut obtenir une liste des figures (fichier `LOF`) et une liste des tables (fichier `LOT`). ?

8.2 — Numérotation & légende

Les figures et tables sont automatiquement numérotées, sauf si l'on utilise les formes `figure*` et `table*`.

`\caption` permet de placer une légende sous la figure/table.

Montrer le fichier
exemple-flottants

Partie 9

Interface avec PostScript

9.1 — Parallèle $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}/\text{PostScript}$

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$: langage de description de document; conçu pour la rédaction de textes scientifique; excellent moteur typographique. Programmation par macros (expansion de textes).

PostScript: langage de description de pages; conçu pour les arts graphiques; opérateurs de haut niveau (détourage). Implantable dans le matériel (imprimantes laser, photocomposeuses). Devenu un standard mondial de fait. Programmation avec pile d'opérateurs.

9.2 — En-tête d'un fichier PostScript

La partie intéressante est la *BoundingBox*, qui donne les dimensions (en points typographiques: 72 au pouce) du rectangle englobant le dessin.

```
%!PS-Adobe-3.0 EPSF-3.0
%%Creator: GraphicConverter V3.8.2
%%Title: logo2.epsf
%%CreationDate: 2003-10-01
%%Pages: 1
%%BoundingBox: 0 0 152 93
%%EndComments
```

Montrer le fichier

logo2.epsf

9.3 — Inclusion d'un fichier EPSF (1)

Un fichier EPSF contient du *PostScript encapsulé*. Prêt à l'emploi, en quelque sorte.

Utiliser le paquetage `epsfig`. Exemple:

```
\documentclass{article}
\usepackage{epsfig}
\begin{document}
\epsfig{file=logo2.epsf}
\end{document}
```

9.4 — Inclusion d'un fichier EPSF (2)

Préparez le texte source selon l'exemple précédent.

Compilez; vous pouvez visualiser le fichier DVI, mais vous ne verrez pas le dessin: son emplacement a été réservé, conformément à la *BoundingBox*.

Utilisez `dvips` pour produire un fichier PS, qui contiendra tout le document: texte, formule et fichier EPSF. Visualisez avec `gv`.

Montrer le fichier
exemple-epsfig

9.5 — Ouverture de TeX (balise `special`)

La balise `\special` permet d'ouvrir T_EX; son fonctionnement est simple: `\special{turlututu\truc-\bidule}` écrit dans le fichier DVI un *special*, accompagné du texte (après expansion des balises). Ensuite, c'est à la visionneuse de l'exploiter!

Applications: inclusion de fichiers graphiques; inclusion de fragments de PostScript (voir PStricks); gestion de la couleur; liens hypertexte, avec le paquetage `hyperref`; passage instantané du texte source à la prévisualisation et inversement; `ActiveDVI`, développé par l'INRIA, permet de préparer des transparents avec effets, interactivité, lancement de programmes extérieurs.

Partie 10

Divers

10.1 — Modes verbatim (1)

Pour citer un très court fragment de \LaTeX , utiliser `\verb`. Exemple:

`\verb+(\mu\)+` et `\verb+(\pi\)+` nous donnent `\(\mu\)` et `\(\pi\)`.

Nous obtenons:

`\(\mu\)` et `\(\pi\)` nous donnent μ et π .

La portée de `\verb` est limitée à une ligne.

10.2 — Modes verbatim (2)

L'environnement `verbatim` permet d'insérer une longue citation:

```
\begin{verbatim}
```

```
Les constructions \verb+(\mu\)+ et \verb+(\pi\)+  
nous donnent \(\mu\) et \(\pi\).
```

```
\end{verbatim}
```

L'environnement `verbatim*` signale explicitement les espaces:

```
\verb+(\mu\)+ et \verb+(\pi\)+ nous donnent \(\mu\) et \(\pi\).
```

10.3 — Modes verbatim (3)

Enfin, on peut inclure un fichier en mode verbatim:

```
\verbatiminput{exemple-array.tex}
```

Le paquetage `verbatim` est nécessaire. La variante `\verbatiminput*` signale explicitement les espaces.

Remarque: dans la diapo précédente, il y a trois niveaux de verbatim imbriqués!

10.4 — Modes verbatim (4)

Voici le résultat:

```
\(  
\left(  
  \begin{array}{lr|r}  
    H_1 & a+b & 12.34 \\ \hline  
    x_1 & c^2 & 2.40 \\ \hline  
    \frac{1}{k} & \sin(\alpha) & 12.80 \\ \hline  
  \end{array}  
\right)  
\)
```

10.5 — L'environnement `picture` (1)

Permet de réaliser des dessins techniques, sans moyens supplémentaires.

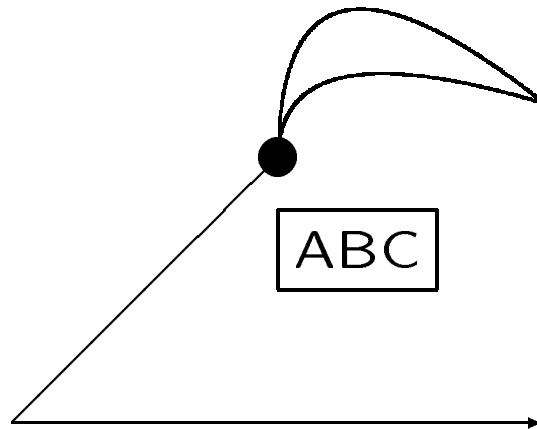
Exemple:

```
\setlength{\unitlength}{10pt}
\begin{picture}(20,20)(0,0) \thicklines
\put(0,0){\line(1,1){10}} \put(10,10){\circle*{2}}
\put(0,0){\vector(1,0){20}}
\put(10,5){\framebox(6,3)[c]{ABC}}
\qbezier(10,10)(10,15)(20,12) \qbezier(10,10)(10,20)(20,12)
\end{picture}
```



10.6 — L'environnement `picture` (2)

Ce que ça donne:



Usage vite pénible.

10.7 — Outils de dessin plus performants

Paquetages `epic`, `eepic`: on reste dans la même zone.

Adobe Illustrator: exporte en PostScript.

Xfig: dessin technique sous X, interfacé avec $\text{T}_\text{E}\text{X}$.

MetaPost: très puissant, résultats remarquables.

PStricks: très puissant également.

Montrer le fichier

`exemple-epic`

10.8 — Références croisées (1): le principe

On pose des *signets*, avec la balise `\label`.

On fait référence à un signet avec l'une des balises `\ref` ou `\pageref`.

`\ref` fait référence à la structure numérotée (chapitre, section, figure, équation, etc.) où a été défini le signet.

`\pageref` fait référence à la page.

Montrer le fichier

`exemple-ref`

10.9 — Index (1)

Dans le préambule, placer:

```
\usepackage{makeidx}  
\makeindex
```

Disposer des clés d'index aux endroits opportuns:

Les mathématiciens parlent d'anneaux`\index{anneau}`
et d'algèbres`\index{algebre@algèbre}`.

?

Placer `\printindex` à l'endroit où vous voulez que l'index soit placé; en général, c'est en fin du texte.

10.10 — Index (2)

Votre fichier principal est `toto.tex`. Lancer la compilation; celle-ci crée le fichier `toto.idx`; soumettre ensuite ce fichier à `makeindex` pour obtenir le fichier `toto.ind`. Ce dernier est l'index remis en forme (entrées collationées et triées).

Compiler une nouvelle fois et admirer le résultat.

La clé `\index{algebre@algèbre}` précise que, dans l'index, c'est le mot `algèbre` qui apparaîtra; mais que la clé de tri est `algebre`. Remarque: il est possible que les versions actuelles de `makeindex` règlent ce problème.

Montrer le fichier
exemple-index

Partie 11

Personnalisation

11.1 — Définir un nouvel environnement (1)

Nous définissons un environnement `attention`:

```
\newenvironment{attention}  
  {\noindent\(\blacktriangleright\)\kern 2pt}  
  {\removeelastskip\smallskip}
```



11.2 — Définir un nouvel environnement (2)

Utilisons cet environnement:

```
\begin{attention}  
Passer chez le teinturier!  
\end{attention}
```

Voici ce que ça donne:

► Passer chez le teinturier!

11.3 — Définir une nouvelle macro

Nous voulons une balise `\IntervalleDiscret` à deux arguments, pour construire des *intervalles discrets*: ainsi, `\(\IntervalleDiscret{1}{n}\)` nous donnera $\llbracket 1, n \rrbracket$.

```
\newcommand{\crog}{[\! [}  
\newcommand{\crod}{]\! ]}  
\newcommand{\IntervalleDiscret}[2]{\crog#1,#2\crod}
```

Commentaire: `\!` est un *espace négatif*; il permet de rapprocher deux éléments typographiques.

Partie 12

Typographie fine

12.1 — Coupure des mots

L'algorithme de découpage d'un paragraphe en lignes considère les espaces entre mots comme des ressorts, extensibles ou compressibles (dans une certaine mesure). Il tente une optimisation globale.

Lorsque celle-ci est impossible, l'algorithme s'autorise des coupures de mots (ou *césures*). Celles-ci suivent des règles adaptées au langage utilisé.

12.2 — Changer de langue

Commencez par consulter le début du TeX log, pour vous assurer que la table est disponible:

```
Hyphenation patterns for english, francais, loaded.
```

Ensuite, `\selectlanguage{english}` vous permet de passer en anglais* et `\selectlanguage{francais}` de revenir au français.

Remarquer sur l'exemple: la modification des coupures; le comportement différent du *point final*.

*Ce qui ne signifie pas que L^AT_EX traduit votre texte!

Montrer le fichier
`exemple-cesure.tex`

12.3 — Forcer une coupure

Dans certains cas, on devra forcer à la main une coupure:

Voici comment forcer la coupure du mot `anti\-constitu\-tionnel`.

Vous pouvez également définir la coupure de certaines mots, pour lesquels les règles semblent mal adaptées:

```
\hyphenation{bara-tineur bar-bou-zard}
```


12.4 — Ligatures

affaire donne `affaire`, tandis que `af{}faire` donne `affaire`.

Explication: les deux `f` consécutifs sont *ligaturés*.

Autres exemples: `fi` et `fi`; `ffi` et `ffi`. Le point sur le `i` est fondu dans le `f`.

Remarque: ces ligatures sont absentes de certaines fontes.

Et pour les hispanisants: `!‘?’` donne `i¿`.

Partie 13

Ressources

13.1 — Livres en français

L^AT_EX, *guide pratique*, de Christian Rolland (O'Reilly)

13.2 — Livres en anglais

L^AT_EX, de Leslie Lamport (Addison-Wesley): une première approche

The L^AT_EX companion, de Goossens, Mittelbach et Samarin (Addison-Wesley); la référence pour tout ce qui concerne les extensions (classes, paquetages, personnalisation ...)

The L^AT_EX graphic companion, de Goossens, Mittelbach et Samarin (Addison-Wesley); la référence pour tout ce qui concerne l'inclusion de graphiques de toutes sortes, et leur préparation

13.3 — Web

CTAN (*Comprehensive T_EX Archive Network*): une énorme archive contenant tout ce qui concerne T_EX et son environnement. Nombreux miroirs, dont un au LORIA.

Des implémentations pour toutes les plate-formes.

La FAQ L^AT_EX, de Marie-Paule Kluth.

Des guides pratiques, des forums de discussion.

Sur mon site Web: les transparents que vous venez de voir, et les exemples; bruno.maitresdumonde.com/stages.

13.4 — Implémentations

Pratiquement toutes les implémentations utilisent la même base:

- moteur `texmf`, écrit en `cweb`
- arborescence TDS
- gestionnaire d'arborescence `kpathsea`

La dernière distribution que j'ai reçue (février 2005) occupe deux CD et un DVD. Elle est le fruit d'un effort collectif, regroupant 28 groupes nationaux d'utilisateurs à la surface de la planète: enfin un effet positif de la mondialisation!.

13.5 — Implémentations PC

Linux: fourni de base (tetex) avec tout l'environnement nécessaire; les experts le piloteront depuis Emacs, avec le mode tex/latex.

Windows: MikTeX, avec YAP et WinEdt: très bien; basé sur tetex. Vient de sortir: proT_EXt, basé sur le précédent ...

13.6 — Implémentations Mac/OS 9

Textures (Blue Sky Research): très ergonomique, mais toujours pas porté sous Mac OS X (et payant, mais vu la qualité ce n'est pas anormal).

CMacTeX: gratuit, pilotable depuis l'éditeur Alpha (programmable en TCL).

OzTeX: avec lequel Eric Weisstein a rédigé sa *Concise Encyclopaedia of Mathematics...*

13.7 — Implémentations Mac/OS X

texmf a été porté sous OS/X, évidemment.

TeXshop est un environnement de travail conçu en Allemagne.

iTeXmax, par Jérôme Laurens. Très bien, en évolution constante.

À chacun d'eux, Baldabiou avait dévoilé, sans difficultés, les secrets du métier. C'était bien plus amusant pour lui que de faire de l'argent à la pelle. Enseigner. Et avoir des secrets à raconter. Il était comme ça, cet homme.

Alessandro Baricco, *Soie*