Centre de Recherche en Informatique de Nancy

Campus Scientifique
Bâtiment LORIA
Boîte Postale 239
54506 Vandœuvre-lès-Nancy Cedex

Téléphone : (+ 33) 83.59.20.00
Télécopie : (+ 33) 83.41.30.79

Guide local (LA)TEX
du site LORIA

J.-M. Antoine, J.-M. Hufflen, D. Rœgel, K. Tombre

CRIN 94-R-256

décembre 1994

Table des matières

Avant-propos

> For each computer system, there is a short companion to this book entitled something like *Local Guide to LaTeX for the Kludge-499 Computer* containing information specific to that system. This companion will be called the *Local Guide.* It is distributed with the LaTeX software.
>
> [Lamport, 1985, page 2]

Depuis que LaTeX a été installé sur notre site, en 1987 si nos souvenirs sont bons, les utilisateurs de plus en plus nombreux de cet excellent outil ont régulièrement été bloqués dans leurs recherches de solutions par des phrases de ce genre, les renvoyant à un guide local pour des détails d'implantation qui sont souvent fondamentaux pour bien maîtriser un logiciel aussi complet que LaTeX. Or ce guide local n'existait pas ! Il y avait bien en ligne des documentations diverses et variées, des livres compléments du manuel de référence ont fait leur apparition au fil des ans, et depuis quelque temps nous avons même une archive assez exhaustive du monde TeX [1], puisque notre site `ftp` est un miroir des trois sites du réseau CTAN (cf. chapitre 10). Cette archive comprend en particulier plusieurs manuels et guides locaux mis au point sur différents sites. Mais même si beaucoup d'environnements se ressemblent, cela ne peut pas complètement remplacer une description précise de l'état des lieux local.

Partant de cette constatation, nous avons donc décidé en 1993 de former un petit groupe chargé de rédiger enfin ce fameux guide local. Le temps de gestation a été plus long que prévu, pour plusieurs raisons. D'abord, nous avions tous d'autres activités à mener en parallèle, et des activités aussi diverses que la rédaction d'une thèse, d'un mémoire CNAM, de publications scientifiques et même d'un livre de cours, la participation à l'administration système au jour le jour de l'ensemble CRIN et INRIA Lorraine, des charges d'enseignement, la responsabilité d'une équipe de recherche, etc. Mais nous souhaitions aussi que quitte à combler enfin une lacune, ce soit bien pensé, bien organisé, aussi utile que possible, sans faire double emploi avec la littérature existante ; cela nous a amenés à remettre plusieurs fois l'ouvrage sur le métier, avant d'être à peu près satisfaits du résultat. Enfin, le monde LaTeX lui-même a fortement évolué ces derniers temps, notamment avec la sortie de LaTeX 2_ε, et nous voulions faire coïncider la sortie de ce guide avec la transition définitive à LaTeX 2_ε sur notre site.

Le résultat est ce que vous êtes en train de lire. Nous devons avouer que nous sommes plutôt contents du résultat, mais notre vraie satisfaction sera de voir ce guide utilisé dans la vie de tous les jours de nos collègues. Malheureusement, malgré tous nos efforts, nous n'avons pas vraiment réussi à nous limiter à un guide qu'on puisse qualifier de « *short companion* », puisque nous en sommes à 100 pages à peu près. Mais nous espérons que ces pages n'auront pas été rédigées en vain...

Maintenant que ce guide existe, nous souhaitons bien sûr qu'il reste vivant, en s'adaptant en permanence aux besoins des utilisateurs et à l'évolution de notre installation. C'est pourquoi nous avons mis en place un mail latex@loria.fr (cf. chapitre 9) pour vous permettre de nous faire parvenir toutes suggestions d'amélioration de cet ouvrage.

Bon LaTeX, ou plutôt bon LaTeX 2_ε, bien que ce soit devenu équivalent [2]...

Les auteurs

1. Sous `/local/tex`.
2. Ce guide local a été totalement rédigé et compilé avec LaTeX 2_ε, et exploite un grand nombre de nouvelles fonctionnalités. Nous espérons qu'il prêche en faveur de cette nouvelle version.

Chapitre 1

LaTeX 2ε

LaTeX 2ε est la nouvelle version standard de LaTeX. Elle intègre entre autres NFSS (une nouvelle interface pour la gestion des polices), et réunit sous un format unique les fonctionnalités de LaTeX 2.09, SliTeX et AMS-LaTeX. **LaTeX 2ε est compatible avec LaTeX 2.09**. Plus précisément, LaTeX 2ε adhère autant que possible aux principes suivants :

1. **Les fichiers compilant sous la version 2.09 produiront la même sortie avec LaTeX 2ε qu'avec la version 2.09**.
2. Toutes les nouvelles caractéristiques de LaTeX 2ε se conforment aux conventions de la version 2.09, facilitant ainsi leur apprentissage aux utilisateurs actuels.

1.1 Comment lancer LaTeX 2ε

LaTeX 2ε est lancé par l'appel de '`latex`'. L'ancienne version (LaTeX 2.09) reste *temporairement* accessible sous '`latex.old`'.

Il est important de noter que les divers fichiers utilisés par LaTeX 2ε ne se trouvent actuellement pas dans le même répertoire que ceux utilisés par LaTeX2.09, ou TeX. L'organisation de ces « répertoires standard » n'est pas encore stabilisée et évoluera certainement prochainement. On évitera donc dans la mesure du possible de mentionner explicitement l'un des répertoires `/usr/local/lib/tex/inputs` ou `/usr/local/lib/tex/-inputs/latex2e` dans une variable d'environnement.

1.2 Le mode de compatibilité

Le mode de compatibilité est le mode particulier dans lequel se trouve LaTeX 2ε au moment où on l'applique à un ancien fichier LaTeX 2.09. Normalement, ces fichiers compilent sous LaTeX 2ε. Voici les principales (et rares) exceptions.

1.2.1 Incompatibilité dans la sélection des polices

Ceci peut se produire si l'ancien fichier (en-tête `\documentstyle`) accède à des commandes internes de gestion des polices. Dans ce cas, les macros correspondantes ou les styles utilisés doivent être modifiés ou mis à jour.

Un cas particulier est celui de l'accès à des commandes comme `\sevit`, `\tenbf`, `\elvtt`, `\twfvrm`, etc. C'est par exemple le cas avec les macros de PicTeX. Dans ce cas, il suffit d'ajouter le style mineur `rawfonts`. Si l'on choisit de convertir le fichier en LaTeX 2ε, et que ce type d'erreur persiste, on chargera le *package* rawfonts, éventuellement en ne demandant de charger que certaines définitions, comme dans l'exemple suivant où seul `\fivrm` est défini :

```
\usepackage[only,fivrm]{rawfonts}
```

1.2.2 Incompatibilité avec de vieux styles

Certaines erreurs sont dues à l'emploi de styles obsolètes. On essaiera alors de trouver des versions plus récentes fonctionnant avec LATEX 2ε.

1.2.3 Incompatibilité liée à l'utilisation de TEX sur des fichiers MlTEX

En utilisant un format LATEX 2ε construit sur TEX, il est nécessaire[3] de modifier les polices utilisées dans les vieux documents[4] écrits en français. Ceci est heureusement fait automatiquement pour les polices qui ne sont pas déclarées explicitement, c'est-à-dire pas avec `\font` ou `\newfont`.

Note pour les experts: *par défaut, le codage utilisé par LATEX 2ε est le codage OT1 ; cependant, lorsque LATEX 2ε est appliqué à un document LATEX 2.09, on force le codage T1,* sauf *pour les anciens fichiers* SLITEX *où le codage OT1 est maintenu.*

Il se peut cependant que des polices soient déclarées explicitement, avec `\newfont` ou `\font`. Si c'est le cas, et que *ces polices sont utilisées pour des caractères accentués*, il faut – si l'on souhaite avoir des divisions de mot optimales – remplacer ces polices par d'autres. Les substitutions à effectuer sont données dans le chapitre consacré à MlTEX. Une solution préférable consiste à convertir le document en un vrai document LATEX 2ε. Dans ce cas, des commandes comme

```
\newfont{\one}{cmbxti10}
\newfont{\two}{cmssi10 scaled 1440}
```

pourront être remplacées par

```
\DeclareFixedFont{\one}{\encodingdefault}{cmr}{bx}{it}{10}
\DeclareFixedFont{\two}{\encodingdefault}{cmss}{m}{it}{14.4}
```

On se renseignera pour les paramètres à donner à `\DeclareFixedFont` dans d'autres cas.

1.2.4 Incompatibilité avec les anciennes erreurs

LATEX 2ε s'efforce d'être compatible avec LATEX 2.09, mais cette compatibilité évite de s'appliquer aux erreurs. En effet, LATEX 2.09 acceptait un certain nombre de constructions que le manuel prohibait. La commande `\verb` était parfois tolérée dans une note de bas de page, alors que le manuel dit explicitement de ne pas utiliser `\verb` en argument d'une autre commande [Lamport, 1985, p. 66]. Un autre exemple est le chargement de styles mineurs *avant* la commande `\documentstyle`. Ce genre de construction n'est plus accepté avec LATEX 2ε. Il ne s'agit pas d'une incompatibilité, mais de la correction d'anomalies.

1.3 Comment convertir un fichier LATEX 2.09 en LATEX 2ε

Si l'on souhaite tirer profit des caractéristiques et nouvelles fonctionnalités de LATEX 2ε, il peut être intéressant de convertir certains fichiers LATEX 2.09 en LATEX 2ε. Cette opération dépend de la complexité du document. Nous supposerons que le document ne fait appel qu'à des fonctionnalités usuelles de LATEX 2.09. Dans ce cas, il faut procéder de la manière suivante :

1. Si le style majeur utilisé est un style standard (`article`, `book`, `letter`, `proc`, `report`, `slides`,...) il faut remplacer `\documentstyle[...]{`⟨*style*⟩`}` par `\documentclass[...]{`⟨*style*⟩`}`.
2. Les styles mineurs utilisés doivent tous être sortis du `\documentclass` *sauf ceux qui sont des options du style majeur.* On trouvera plus loin (§ 1.4.1) les options autorisées par les classes standard. Les autres styles mineurs doivent être chargés par la commande `\usepackage`, après le `\documentclass`.

   Par exemple :

   ```
   \documentstyle[12pt,array,french]{report}
   ```

3. *stricto sensu*, ce n'est nécessaire que pour les documents tirant profit de MlTEX, donc ceux qui utilisent des caractères accentués.

4. c'est-à-dire, commençant avec `\documentstyle`.

devient

```
\documentclass[12pt]{report}
   \usepackage{array}
   \usepackage{french}
```

On peut aussi écrire :

```
\documentclass[12pt]{report}
   \usepackage{array,french}
```

Le reste du document peut normalement rester inchangé.

3. Si le style majeur n'est pas un style standard, on peut essayer d'en faire une copie pour créer la classe correspondante, c'est-à-dire un fichier de même nom, mais avec le suffixe .cls. Ceci peut ne pas fonctionner.

Il est conseillé à chacun de convertir petit à petit les anciens fichiers LaTeX 2.09 qui sont susceptibles d'être modifiés par la suite et en particulier les documents en français, pour lesquels le mode de compatibilité de LaTeX 2ε est plus aléatoire.

1.4 Principales commandes propres à LaTeX 2ε

Nous passons ici en revue quelques commandes qui sont soit complètement nouvelles, soit ont été modifiées par rapport à la version 2.09. Cette énumération se base sur le document `usrguide.dvi`[5] et **n'est nullement exhaustive**.

1.4.1 Commandes du préambule

Afin de distinguer les anciens (2.09) documents de ceux qui utilisent des fonctions de la nouvelle version, les documents LaTeX 2ε utilisent une commande différente sur la première ligne.

`\documentclass` [⟨*liste d'options*⟩] {⟨*nom de la classe*⟩} [⟨*date de la version*⟩]

Cette commande remplace `\documentstyle` ; il est possible de spécifier une date requise pour la classe demandée. La classe est le type de document, par exemple, article, book, letter, proc, report, slides, ...

Les options disponibles avec les classes standard sont les suivantes :

- Classe article : options a4paper, a5paper, b5paper, letterpaper, legalpaper, executivepaper, landscape, 10pt, 11pt, 12pt, oneside, twoside, draft, final, titlepage, notitlepage, onecolumn, twocolumn, leqno, fleqn, openbib.
- Classes book et report : les mêmes options que la classe article plus openright et openany.
- Classe letter : options a4paper, a5paper, b5paper, letterpaper, legalpaper, executivepaper, 10pt, 11pt, 12pt, draft, final, oneside, twoside, leqno, fleqn.
- Classe proc (pour les articles destinés à des actes de conférences) : les mêmes options que la classe article, mais sans les options a5paper, b5paper, titlepage et onecolumn.
- Classe slides : options a4paper, a5paper, b5paper, letterpaper, legalpaper, executivepaper, landscape, clock, titlepage, notitlepage, draft, final, leqno.

D'autres classes sont disponibles. Parmi elles, signalons :

- seminar : classe correspondant à l'ancien style majeur `seminar` (cf. chapitre 6).

5. dans `/local/tex/doc/LaTeX2e`.

– TheseCRIN : classe de thèse locale (cf. chapitre 8).

\documentstyle [⟨*liste d'options*⟩] {⟨*nom de la classe*⟩} [⟨*date de la version*⟩]

Cette commande sélectionne le *mode de compatibilité* avec LaTeX 2.09. Notons que LaTeX 2ε est plus lent en mode de compatibilité.

\usepackage [⟨*liste d'options*⟩] {⟨*liste de packages*⟩} [⟨*date de la version*⟩]

Chargement d'un ou plusieurs *package(s)* (anciennement styles mineurs) et de leurs options (notion non présente dans LaTeX 2.09). L'inclusion d'un *package*[6] ajoute de nouvelles commandes ou redéfinit des commandes existantes afin de procurer des fonctionnalités complémentaires.

1.4.2 Définitions de commandes et d'environnements

\newcommand {⟨*commande*⟩} [⟨*nombre*⟩] [⟨*défaut*⟩] {⟨*définition*⟩}
\renewcommand {⟨*commande*⟩} [⟨*nombre*⟩] [⟨*défaut*⟩] {⟨*définition*⟩}
\newenvironment {⟨*commande*⟩} [⟨*nombre*⟩] [⟨*défaut*⟩] {⟨*déf-ini*⟩} {⟨*déf-fin*⟩}
\renewenvironment {⟨*commande*⟩} [⟨*nombre*⟩] [⟨*défaut*⟩] {⟨*déf-ini*⟩} {⟨*déf-fin*⟩}

Ces commandes admettent désormais un paramètre optionnel.

1.4.3 Les boîtes

\makebox [⟨*largeur*⟩] [⟨*position*⟩] {⟨*texte*⟩}
\framebox [⟨*largeur*⟩] [⟨*position*⟩] {⟨*texte*⟩}
\savebox {⟨*commande*⟩} [⟨*largeur*⟩] [⟨*position*⟩] {⟨*texte*⟩}

Ces commandes existaient déjà en LaTeX 2.09 mais il est désormais possible d'utiliser l'une des dimensions suivantes dans l'argument ⟨*largeur*⟩ :

\height la hauteur de ⟨*texte*⟩ ;

\depth la profondeur de ⟨*texte*⟩ ;

\totalheight la somme de \height et \depth ;

\width la largeur de ⟨*texte*⟩.

Ainsi, pour écrire « hello » au centre d'une boîte ayant deux fois sa largeur naturelle, on utilise :

```
\makebox[2\width]{hello}
```

\begin{lrbox} {⟨*commande*⟩}
⟨*texte*⟩
\end{lrbox}

Ce nouvel environnement permet en particulier de créer facilement un environnement « encadré ».

6. La traduction actuellement employée par l'association GUTenberg semble être « extension », mais nous avons choisi pour l'instant de parler de « *package* », « extension » n'étant pas encore entré dans les mœurs.

1.4.4 Mesurer des objets

```
\settowidth {⟨variable de dimension⟩} {⟨texte⟩}
\settoheight {⟨variable de dimension⟩} {⟨texte⟩}
\settodepth {⟨variable de dimension⟩} {⟨texte⟩}
```

La première de ces commandes était présente dans LaTeX 2.09. Les deux autres en sont les analogues.

1.4.5 Contrôle des sauts de page

```
\enlargethispage {⟨dimension⟩}
\enlargethispage* {⟨dimension⟩}
```

Ces deux commandes permettent d'accroître la taille de la page courante de ⟨*dimension*⟩, qui peut être négatif. Ceci permet d'ajouter une ligne ou au contraire d'en retirer une si nécessaire.

1.4.6 Flottants

La commande `\suppressfloats` et le paramètre '`!`' (qui s'utilise en conjonction avec `h`, `t` ou `b`) permettent de mieux contrôler le positionnement des figures et autres objets flottants (cf. [Goossens *et al.*, 1994, pp. 144–145]).

1.4.7 Gestion des polices

Toutes les commandes de NFSS sont disponibles. On se reportera donc à la partie correspondante (§ 1.7).

1.4.8 Forcer le mode mathématique

```
\ensuremath {⟨commandes mathématiques⟩}
```

Avec LaTeX 2.09, on devait écrire `\newcommand{\Gp}{\mbox{$G_p$}`. Maintenant, on peut employer `\newcommand{\Gp}{\ensuremath{G_p}}`, ce qui a l'avantage de fournir des symboles dont la taille varie correctement dans une formule mathématique.

1.4.9 Logos

```
\LaTeX
\LaTeXe
```

`\LaTeX` (produisant 'LaTeX') est toujours la « principale » commande de logo mais si vous avez besoin de faire référence aux nouvelles fonctions, vous pouvez dire `\LaTeXe` (produisant 'LaTeX 2_ε').

1.5 Les *packages* de LaTeX 2_ε

Les *packages* pouvant être utilisés avec LaTeX 2_ε sont très nombreux et ne peuvent pas tous être décrits ici. Cent-cinquante d'entre-eux sont décrits dans *The LaTeX Companion* [Goossens *et al.*, 1994]. En voici quelques-uns qu'il nous paraît utile de mentionner. On les chargera avec `\usepackage`.

- alltt : définition de l'environnement `alltt`, similaire à l'environnement `verbatim`, mais où l'on peut insérer des commandes au moyen de « \ ».
- array : ce *package* améliore les environnements `tabular` et `array`. Il permet par exemple de formatter une seule colonne d'un tableau en italiques, sans avoir à répéter `\textit`.

- calc : permet l'écriture de certaines expressions arithmétiques dans des commandes comme `\setlength`, etc.
- changebar : ce *package* permet de placer des barres de modification en marge de certains passages d'un document.
- enumerate : ajoute un paramètre optionnel à l'environnement `enumerate` pour modifier le « style » du compteur.
- fancybox : ce *package* fournit de nombreuses commandes pour faire des encadrements (en particulier de texte `verbatim`), des ombrages, etc.
- fancyheadings : permet de paramétrer facilement les en-têtes d'un document.
- flafter : ce *package* permet de forcer un objet « flottant » (figure ou table) à ne jamais apparaître dans le texte avant sa référence.
- french : typographie française (cf. chapitre 7).
- ifthen : implantation de structures de contrôle comme `\ifthenelse` ou `\whiledo`.
- inputenc : ce *package* permet de spécifier le codage des caractères en entrée. Si l'on souhaite utiliser le codage ISO Latin-1, on indiquera `\usepackage[latin1]{inputenc}`.
- latexsym : définition de quelques caractères de la fonte `lasy` (LATEX Symbols), comme `\Box` ($\Box$), `\Diamond` ($\Diamond$) ou `\rhd` ($\rhd$).
- longtable, supertabular : ces deux *packages* permettent de créer des tableaux s'étendant sur plusieurs pages.
- minitoc : mini-tables des matières par chapitres, par parties, etc.
- multicol : permet d'écrire du texte sur plusieurs colonnes, et en particulier de changer de nombre de colonnes au milieu d'une page.
- portland : définit les environnements `portrait` et `landscape`.
- rotating : fournit en particulier l'environnement `rotate` permettant de tourner un objet d'un angle donné, ainsi que des environnements de rotation de tables et de figures.
- setspace : permet de varier l'interligne et en particulier d'écrire un document en « double interligne » ; ce *package* améliore le *package* doublespace.
- syntonly : Pour utiliser cette option, il faut mettre `\syntaxonly` dans le préambule. LATEX 2ε ne teste que la syntaxe, imprimant tout « warning » ou message d'erreur pouvant se produire. Il n'y a pas de formatage ni de sortie .dvi. L'exécution est évidemment plus rapide.
- t1enc : définit le codage T1 (Cork) comme étant le codage par défaut. En l'absence d'autres directives, ceci sélectionne les polices DC. Ce *package* est quasiment indispensable si l'on écrit en français.
- tracefnt : rend compte du chargement des fontes et de leur changement. Ce *package* admet plusieurs options :
  - infoshow donne de nombreuses informations sur le chargement des polices.

    ```
    Font Info: Font shape `U/lasy/b/n' in size <7> not available.
    Font       Font shape `U/lasy/m/n' tried instead on input line 47.
    ```

  - errorshow : seules les erreurs sont affichées. En l'absence d'erreurs, cette option peut être utilisée pour supprimer les divers messages d'information.

– warningshow : les « warnings » sont transformés en condition d'erreur ; par exemple, un `\include` sur un fichier inexistant et non demandé par `\includeonly` donne :

```
! No such counter.
\@latexerr ...rcontextlines \m@ne \errmessage {#1}
                                                  \endgroup
l.539 \include{essai}
```

– loading permet de suivre le chargement des polices. Il en résulte des messages comme :

```
Font Info: Loading external cmr7 at7.0pt as OT1/cmr/m/n/7
Font        on input line 515.
```

– debugshow inclut l'option infoshow et donne des informations plus complètes sur les changements de police. Par exemple :

```
Font Note: Switching to \T1/cmr/m/sc/10  on input line 43.
Font Note: Changing size to 10/12.0pt on input line 43.
```

– vpage : ce *package* fournit des commandes pour redéfinir aisément les marges d'un document.

– wrapfig, picinpar : ces deux *packages* permettent d'« encastrer » une figure dans un paragraphe, sous certaines conditions.

– xspace : définit une commande `\xspace` supprimant l'absorbtion des espaces après une commande.

D'autre part, tous les *packages* de police *PostScript* de NFSS sont utilisables avec LaTeX 2ε (cf. § 1.7.6).

1.6 Faire des transparents avec LaTeX 2ε

La classe slides rend désormais le format SliTeX superflu. Grâce à l'option de compatibilité, LaTeX 2ε peut être appliqué aux anciens fichiers SliTeX 2.09. On se référera au chapitre 6 pour plus de détails.

1.7 NFSS

NFSS (**N**ew **F**ont **S**election **S**cheme, opposé à OFSS) est essentiellement une nouvelle version de l'ensemble des commandes gérant l'accès aux fontes dans LaTeX. Ce module fait partie de LaTeX 2ε. Grâce à cette nouvelle interface, AMS-LaTeX et SliTeX deviennent désormais des options de LaTeX et ne sont plus des formats distincts.

Les commandes de modification de fonte comportent maintenant un mécanisme d'*héritage*. Plus précisément, ce sont les caractéristiques non explicitement modifiées parmi le *type de codage*, la *famille*, la *série*, l'*aspect*, et la *taille* qui restent identiques. D'autre part, il est désormais possible de substituer une fonte à une fonte demandée, mais non existante.

Grâce à NFSS, il est possible de définir facilement des symboles mathématiques dont la taille varie en fonction du texte. Ceci était particulièrement lourd avec LaTeX 2.09 sans NFSS. Les fontes utilisables sont très facilement extensibles.

Au début d'une exécution LaTeX typique, les cinq attributs ont les valeurs :

encoding	`OT1`	Normal TeX encoding
family	`cmr`	Computer Modern Roman
shape	`n`	Normal (upright roman)
series	`m`	Medium weight
size	`10pt`	

1.7.1 Exemples

1. `\fontfamily{ccr}\fontencoding{OT1}\fontseries{c}\fontshape{sl}\selectfont`

   sélectionne la famille *Concrete*, codage OT1, série *condensed*, aspect *slanted*. La taille n'est pas modifiée. Les commandes utilisées ici sont des commandes de bas niveau, normalement non utilisées directement.

2. `\fontfamily{ccr}\fontencoding{OT1}\fontseries{c}\fontshape{sl}\fontsize{9}{11pt}%`

   `\selectfont`,même effet mais en plus, la fonte demandée est de 9 points et l'interligne (défini par `\baselineskip`) est de 11 points.

3. Écriture cyrillique :

   on utilisera le *package* crin-cyr en attendant que l'$\mathcal{AMS}$ sorte son propre *package* [7].

```
...
\usepackage{crin-cyr}
...

\newcommand{\tcyrm}{\fontencoding{OT2}\fontfamily{cmr}%
        \fontseries{m}\fontshape{n}\selectfont}
...
```

   Ceci a pour effet de charger les déclarations figurant dans `OT2cmr.fd`.

   puis `{\tcyrm Mikhail Gorbach\"ev}` pour obtenir Михаил Горбачёв.

   Un exemple minimal de fichier est :

```
\documentclass{article}
\usepackage{crin-cyr}
\newcommand{\tcyrm}{\usefont{OT2}{cmr}{m}{n}}
\begin{document}
{\tcyrm Mikhail Gorbach\"ev} Mikhail Gorbach\"ev
\end{document}
```

4. Création d'un nouvel alphabet mathématique :

```
\documentclass{article}
\DeclareMathAlphabet{\eusm}{U}{}{}{}  % Euler script math
\SetMathAlphabet\eusm{normal}{U}{eus}{m}{n}
\SetMathAlphabet\eusm{bold}{U}{eus}{b}{n}
\begin{document}
$\mathcal{ABCDEF}$

$\eusm{ABCDEF}$
\end{document}
```

   `\eusm` peut maintenant être utilisé en mode mathématique, tout comme `\mathcal` (remarquons qu'en OFSS-LaTeX, la syntaxe de `\cal` est légèrement différente : c'est une bascule).

5. `\sf\bf` conduit à l'utilisation de la fonte `cmbx` (« **police** », `\bf` écrase `\sf`, pas d'héritage). Par contre, on obtient la police `cmssbx` (« **police** », héritage) avec `\sffamily\bfseries`.

7. On notera cependant que ce codage cyrillique ne convient en pratique que pour de courtes citations ; des solutions plus appropriées existent pour les autochtones (voir par exemple `http ://sunsite.oit.unc.edu/sergei/cysoft.html`).

1.7.2 Le codage

Le codage d'une fonte indique la position des caractères. Lorsqu'on passe d'un codage à l'autre, certaines commandes doivent être modifiées. C'est en particulier le cas des commandes d'accentuation. Par exemple, avec les fontes classiques de TeX (codage dit OT1), la commande `\'` *superpose* un accent aigu au caractère qui suit. Dans les nouvelles fontes DC (codage dit T1 ou de Cork), les caractères accentués usuels sont présents. Par conséquent, on ne fait plus correspondre à `\'e` la *superposition* de « ´ » et « e », mais la *sélection* du caractère « é ». On pourra examiner avec profit les fichiers `OT1enc.def` et `T1enc.def`[8] qui montrent les commandes modifiées dans les deux codages OT1 et T1.

En résumé, les commandes d'accès aux caractères accentués ou spéciaux (comme 'æ') sont toujours les mêmes, mais leur définition est fonction du codage.

1.7.2.1 Codages possibles

Les codages actuellement reconnus sont les suivants :

OT1	old TeX text encoding
OT2	TeX text cyrillic UW fonts
T1	TeX text encoding
OML	old TeX math letters (italic) encoding
OMS	old TeX math symbols encoding
OMX	old TeX math extension symbols encoding
U	unknown encoding

À l'avenir ils seront complétés pour différentes classes d'alphabets, en particulier pour des extensions des fontes mathématiques.

1.7.2.2 Choix du codage : OT1 ou T1 (Cork)

En général, le codage est imposé par la langue que l'on utilise. Pour écrire en anglais, une police au codage OT1 convient. Par contre, pour écrire en français, il est préférable d'utiliser une police au codage T1, donc de charger le *package* fontenc avec l'option T1 : `\usepackage[T1]{fontenc}`.

Il arrive aussi que la police impose le codage à employer. Les polices Pandora n'existent par exemple qu'en OT1. Mais le *package* pandora sélectionne automatiquement le codage approprié.

Le codage par défaut de LaTeX est le codage OT1. Il peut être changé en redéfinissant la commande `\encodingdefault`. Par exemple : `\renewcommand{\encodingdefault}{T1}`.

1.7.3 Les tables de correspondances

Pour que les exemples donnés en introduction fonctionnent, il faut d'une part que les fontes nécessaires existent, d'autre part fournir à LaTeX 2ε une *table de correspondance.* Ces tables viennent en général avec les *packages* qui les nécessitent. Exemple : `OT1ccr.fd`[9] pour les fontes de la famille `ccr`.

Les correspondances usuelles sont connues de LaTeX 2ε et se trouvent dans des fichiers .fd (*font definitions*). Pour en ajouter d'autres, on s'inspirera d'un fichier comme `OT1ccr.fd`.

Montrons sur un exemple comment ceci peut être utilisé. Imaginons qu'une fonte de petites capitales grasses ait été créée avec METAFONT, et que son nom soit `mycmbxcsc10` (cf. § 3.4.1). On veut faire en sorte que `\bf\sc` soit équivalent à `\fontencoding{OT1} \fontfamily{cmr} \fontseries{bx} \fontshape{sc} \selectfont` et que ceci sélectionne la fonte `mycmbxcsc`. Il suffit alors de créer un fichier `boldsc.sty` dans lequel on écrira par exemple :

```
\DeclareFontShape{OT1}{cmr}{bx}{sc}{
   <5> <6> <7> <8> <9> <10> <10.95> <12> <14.4> <17.28> <20.74> <24.88>mycmbxcsc10
   }{}
```

8. tous deux dans `/usr/local/lib/tex/inputs/latex2e`.
9. dans `/usr/local/lib/tex/inputs/latex2e`.

Ensuite, on ajoutera `boldsc` à la liste des *packages* du document.

Cet exemple conduit à formuler quelques remarques :

- La série donnée est `bx` (*bold extended*) car c'est celle que sélectionne `\bf` (`\bfdefault`).
- Dans cet exemple, il n'y a qu'un seul fichier source (à savoir `mycmbxcsc10`), mais on pourrait très bien faire appel à des fichiers différents lorsqu'on est au-dessous d'une certaine taille par exemple.
- Il faut déclarer avec `\DeclareFontFamily` une famille de fontes avant de donner une table de correspondances.

  Pour les fontes *PostScript*, c'est beaucoup plus simple. Par exemple, pour définir les petites capitales Bookman (Bookman Small Caps), et sachant que `pbklcq` est le nom de la fonte virtuelle correspondante, il suffit d'écrire :

```
\DeclareFontShape{T1}{pbk}{m}{sc}{<->pbklcq}{}
```

1.7.3.1 Substitution de fontes

- `\DeclareFontShape{OT1}{panr}{m}{it}{ sub * panr/m/sl }{}` : cette commande fait correspondre l'italique à l'oblique (*slanted*).
- `\DeclareFontSubstitution{OT1}{panr}{m}{n}` : cette commande déclare une fonte par défaut à utiliser si une autre fonte n'est pas trouvée. Si on demande une fonte avec les attributs `panr/bx/it` et que ce n'est pas une combinaison connue, la définition `\DeclareFontSubstitution` conduit à remplacer la fonte demandée par `panr/m/m` si le codage OT1 est employé. Ce genre de substitution est intéressant pour la portabilité du fichier source.
- `\DeclareErrorFont` : si une fonte n'a pas pu être trouvée, ni aucune déclaration de substitution, on prend la fonte déclarée par `\DeclareErrorFont`.

1.7.3.2 Utilisation d'une fonte définie par ses attributs

Outre la sélection « cumulative » des attributs d'une fonte au moyen de `\fontfamily`, `\fontencoding`, `\fontseries`, `\fontshape` et `\selectfont`, on peut sélectionner la fonte en donnant la liste des attributs à la commande `\usefont` et écrire par exemple : `\newcommand\goth{\usefont{U}{ygoth}{m}{n}}`.

1.7.4 Les fontes « texte »

Les commandes classiques `\rm`, `\tt`, etc., sont toujours disponibles dans LaTeX 2ε, mais on dispose des nouvelles commandes suivantes, chacune s'appliquant à un argument : `\textrm`, `\textsf`, `\texttt`, `\textbf`, `\textit`, `\textsl`, `\textsc`, etc. On dispose aussi de `\emph`, analogue de `\em`. Ces commandes peuvent être composées.[10] Le gras italique s'obtient par exemple avec `\textit{\textbf{...}}`. D'autre part, ces commandes ont l'avantage d'intégrer la *correction italique*. Toutefois, elles sont réservées à de courts passages ne s'étendant pas sur plus d'un paragraphe. Si l'on veut mettre une grande partie du texte en `Computer Modern Typewriter Type`, on emploiera `\ttfamily` en écrivant :

```
{\ttfamily ....
     ...
}
```

Ce qui précède concerne aussi les commandes `\textrm` (`\rmfamily`), `\textsf` (`\sffamily`), `\textbf` (`\bfseries`), `\textit` (`\itshape`), `\textsl` (`\slshape`), `\textsc` (`\scshape`), etc.

10. Comme ces commandes sont cumulatives, il pourra être nécessaire — dans de rares cas — de « réinitialiser » la police courante avec `\normalfont`.

1.7.5 Les caractères mathématiques

Maintenant, en mode mathématique, on dispose des commandes suivantes, dont l'effet se limite au texte, c'est-à-dire ne s'applique pas aux symboles :

Exemple	*Effet*
`\mathnormal`	*style italique mathématique*
`\mathcal`	$\mathcal{STYLE\ CALLIGRAPHIQUE}$
`\mathrm`	style droit
`\mathbf`	**style gras**
`\mathsf`	style sans empattements
`\mathit`	*style italique*
`\mathtt`	`style machine à écrire`

Ces commandes ont toutes un argument (à la différence de l'ancien `\cal` par exemple). Il ne faut donc plus écrire `${\bf F}(x)$` mais `$\mathbf{F}(x)$` pour obtenir $\mathbf{F}(x)$. `\mathnormal` correspond à l'ancien `\mit`.

De nombreuses autres commandes sont disponibles. En voici quelques exemples :

- Déclaration d'un nouvel alphabet mathématique et de ses variétés :

  `\DeclareMathAlphabet{\mathtt}{OT1}{cmtt}{m}{n}`

  `\SetMathAlphabet\mathtt{bold}{OT1}{cmr}{bx}{n}`,

  Pour ces commandes et les suivantes, les associations du type `OT1/cmtt/m/n` doivent déjà être connues.
- Déclaration d'une nouvelle série de symboles mathématiques :

  `\DeclareSymbolFont{AMSa}{U}{msa}{m}{n}`

  (comme précédemment, on peut définir des variantes avec `\SetSymbolFont` :

  `\DeclareSymbolFont{letters}{OML}{cmm}{b}{it}`,

  `\SetSymbolFont{letters}{bold}{OML}{ccm}{m}{it}`).

  Une fois ces déclarations faites, `$\int_{a}^{b}x^2\,dx=\frac{1}{3}(b^3-a^3)$` donne

  $$\int_{\boldsymbol{a}}^{\boldsymbol{b}} \boldsymbol{x}^2\, \boldsymbol{dx} = \frac{1}{3}(\boldsymbol{b}^3 - \boldsymbol{a}^3)$$

  et `{\boldmath$\int_{a}^{b}x^2\,dx=\frac{1}{3}(b^3-a^3)$}` donne

  $$\int_{a}^{b} x^2\, dx = \frac{1}{3}(b^3 - a^3).$$

  L'exemple précédent illustre l'emploi de l'une des catégories mathématiques standard. Outre la famille « `letters` », on dispose aussi des familles « `operators` », « `symbols` » et « `largesymbols` ».
- Utilisation de ces familles de symboles :
  - déclaration d'un symbole particulier : `\DeclareMathSymbol\lozenge{\mathord}{AMSa}{"06}`
  - utilisation de ces symboles comme un alphabet mathématique : `\DeclareSymbolFontAlphabet{\bbold}{AMSb}`.
- Version mathématique : généralement, une police mathématique existe en deux versions : `normal` et `bold`. L'exemple suivant montre comment définir une nouvelle version, appelée `euler` et la sélectionner :

```
\DeclareMathVersion{euler}
\SetSymbolFont{operators}{euler}{OT1}{ccr}{m}{n}
\SetSymbolFont{letters}{euler}{OML}{ccm}{m}{it}
\SetSymbolFont{symbols}{euler}{OMS}{cmsy}{m}{n}
\mathversion{euler}
```

1.7.6 Polices *PostScript*

Ces polices sont disponibles dans les codages OT1 et T1.

Pour mettre l'écriture *romaine* (`\rmfamily`) d'un document en Times Roman, l'écriture *linéale* (`\sffamily`) en Helvetica et l'écriture *mécane* (`\ttfamily`) en `Courier`, on peut mettre dans le préambule :

```
\renewcommand{\rmdefault}{ptm}
\renewcommand{\sfdefault}{phv}
\renewcommand{\ttdefault}{pcr}
```

Comme le codage par défaut est OT1, ceci conduit à utiliser la fonte *PostScript* Times Roman en codage OT1.

Plus simplement, on peut utiliser l'un des *packages* suivants : avant (AvantGarde), bookman (Bookman), courier (Courier), helvet (Helvetica), newcent ou ncs (New Century Schoolbook), palatino (Palatino) ou times (Times) (cf. § 3.5.2).

À ces *packages* « classiques », il faut ajouter le *package* pifont pour les fontes *PostScript* « symboles » ; ce *package* définit en particulier les commandes `\Pisymbol`, `\Pifont`, `\ding`, etc., ainsi que les environnements `dinglist`, `dingautolist`, qui permettent d'utiliser les symboles des polices `Zapf Dingbats` ou `Symbol` (cf. [Goossens *et al.*, 1994, pp. 335–338]).

1.7.7 Tests de fontes

Pour effectuer des tests de fontes, c'est-à-dire pour examiner le « contenu » d'une fonte, on pourra utiliser le fichier `nfssfont.tex`, analogue de `testfont.tex` (cf. § 3.6).

1.8 Documentation sur LaTeX 2ε et NFSS

Les principales nouvelles caractéristiques de LaTeX 2ε sont décrites sommairement dans `usrguide.dvi`, `clsguide.dvi` et `fntguide.dvi`, `cfgguide.dvi`, tous dans `/local/tex/doc/LaTeX2e`. Ces fichiers sont mis à jour tous les six mois.

Les classes et *packages* de LaTeX 2ε sont documentés dans la distribution et on trouvera de nombreux pointeurs sur WWW.[11]

LaTeX est décrit dans une nouvelle édition de *LaTeX : A Document Preparation System* [Lamport, 1994] et dans *The LaTeX Companion* [Goossens *et al.*, 1994]. Ce dernier ouvrage contient aussi une description complète de NFSS, interface pour laquelle il sera aussi profitable de lire l'article de Sebastian Rahtz, paru dans TUGboat, `essnfss2.dvi`[12].

1.9 AMS-LaTeX

AMS-LaTeX est un ensemble de *packages* conférant à LaTeX les caractéristiques d'AMS-TeX[13]. Il y a deux classes principales de documents et leurs options :

- la classe « Article AMS » amsart.
- la classe « Livre AMS » amsbook.

Ces deux classes admettent un certain nombre d'options décrites dans la documentation. On dispose en particulier des options 8pt, 9pt, 11pt, et 12pt, la taille par défaut étant de 10pt.

Le *package* amstex adapte la majeure partie des commandes mathématiques d'AMS-TeX afin de les rendre disponibles dans LaTeX 2ε. Cette « option » diffère néanmoins un peu d'AMS-TeX 2.1 (cf. `amsldoc.dvi`[14],

11. `http://www.loria.fr/tex`.
12. dans `/local/tex/doc/NFSS/NFSS2`.
13. Notons cependant que certaines fonctionnalités de LaTeX sont altérées par l'emploi du *package* AMS-LaTeX. C'est le cas du caractère `@` qu'il est nécessaire de doubler, ou encore de la commande `\ldots` qui ne fonctionne qu'en mode mathématique.
14. dans `/local/tex/archive/macros/latex/packages/amslatex`.

appendice C). Ce *package* est automatiquement chargé par les classes de document amsart et amsbook. Avec la classe article, on écrira donc

```
\documentclass{article}
\usepackage{amstex}
```

Le *package* amstex admet en outre des options complémentaires[15] :

- noamsfonts : cette option permet de ne pas déclarer les polices AMS.
- intlimits : cette option déplace les limites des intégrales depuis la droite vers « au-dessus et au-dessous ».
- nointlimits* : cette option a l'effet inverse de la précédente.
- nosumlimits : cette option déplace vers le côté la position des limites des sommes et autres opérateurs cumulatifs.
- sumlimits* : cette option a l'effet inverse de la précédente.
- namelimits* : cette option définit un *switch* de telle sorte que les indices inférieurs et supérieurs des noms d'opérateurs seront du côté droit plutôt qu'au-dessus et au-dessous, dans une *displayed* équation.
- nonamelimits : cette option a l'effet inverse de la précédente.
- leqno, reqno : ces options permettent de placer les numéros des équations à leur gauche ou à leur droite.
- tbtags : cette option s'applique aux environnements `split` ; elle déplace le numéro de l'équation verticalement (*top-bottom tags*) par rapport à sa position par défaut.
- centertags : cette option centre verticalement le numéro d'une équation dans l'environnement `split`.
- fleqn : cette option place les équations à une distance constante de la marge et ne les centre donc pas.

Le *package* amstex charge aussi les *packages* amstext, amsfonts et amsbsy.

On trouve aussi des *packages* plus spécialisés, pouvant tous être utilisés indépendamment du *package* amstex :

- amscd qui adapte à LaTeX les commandes d'AMS-TeX pour les diagrammes commutatifs.
- amsbsy : ce *package* définit d'une part la commande `\boldsymbol` qui produit des symboles mathématiques gras lorsque les fontes adéquates existent, d'autre part une commande `\pmb` ('*poor man's bold*') pouvant être employée lorsqu'une fonte grasse mathématique n'existe pas. Ce *package* est automatiquement chargé par le *package* amstex.
- amsfonts : ce *package* fournit la commande `\newsymbol` pour définir des symboles mathématiques à partir des deux fontes mathématiques de la collection AMSfonts. On trouvera plus d'informations dans le manuel d'AMSfonts (`userdoc.tex`[16]) ; ce *package* définit aussi la commande `\mathbb` pour produire l'alphabet majuscule '*blackboard bold*'. Ce *package* est automatiquement chargé par l'option amstex.
- amsintsm : ce *package* fournit quelques variantes pour l'impression des sommes et des intégrales.
- amssymb : ce *package* définit des noms pour tous les symboles des fontes `msam` et `msbm` du paquetage AMSfonts.
- amstext : ce *package* définit la commande `\text` qui facilite l'incorporation de fragments de texte dans une équation ou des indices. Les tailles sont automatiquement agrandies ou diminuées dans les indices. Ce *package* est automatiquement chargé par le *package* amstex.
- amsthm : ce *package* est inspiré du *package* theorem et permet d'étendre les possibilités de l'environnement `theorem`.
- amsxtra : ce *package* définit quelques commandes utilisateurs très rarement utilisées.
- romref : ce *package* a pour effet de mettre les références dans une police romaine, même si le contexte est italique.

15. les options marquées '*' sont les options par défaut.
16. dans `/local/tex/archive/fonts/ams/amsfonts/doc`.

1.9.1 Documentation

On consultera le guide d'AMS-LaTeX (AMS-LaTeX User's Guide, version 1.2, fichier `amsldoc.dvi`[17]) et les deux fichiers d'exemple `testart.dvi` et `testbook.dvi` dans le même répertoire. Ce document donne en particulier des indications pour la compilation d'anciens fichiers AMS-LaTeX. Pour AMS-TeX, on se reportera à *The Joy of TeX* [Spivak, 1990].

1.10 LaTeX3

Pour avoir une idée de ce que sera LaTeX3, on lira avec profit "The LaTeX3 Project" de Frank Mittelbach et Chris Rowley (fichier `l3d001.dvi`[18]).

17. dans `/local/tex/archive/macros/latex/packages/amslatex`.
18. dans `/local/tex/doc/LaTeX3`.

Chapitre 2

TEX et MlTEX

Lorsque TEX a été conçu, l'anglais était sa langue privilégiée. Les polices étaient adaptées à l'anglais. Même s'il était possible d'écrire du texte dans une autre langue, les polices disponibles n'étaient pas dédiées à cette tâche. La principale limitation était l'incapacité de TEX à couper un mot comportant un ou plusieurs caractères accentués. Cette limitation n'en était pas réellement une car il « suffisait » de créer des polices spéciales. Malheureusement, avant 1991, ces polices n'existaient pas et une autre solution a été proposée : la variante MlTEX (Multi-Lingual TEX). MlTEX diffère de TEX par le fait qu'il est possible de couper des mots accentués, écrits avec les polices classiques de TEX.

Maintenant, avec l'avènement des polices dites DC, MlTEX n'a plus de raison d'être. Il est en fait souhaitable de passer de MlTEX à TEX. C'est d'autant plus souhaitable qu'il existe des incompatibilités entre MlTEX et certains *packages* de LATEX 2_ε, incompatibilités qui nécessitent une modification de MlTEX et/ou de certaines polices.

Nous avons profité de l'installation de LATEX 2_ε pour faire tourner ce format exclusivement sous le *moteur* TEX et non MlTEX. Le passage progressif de chacun à LATEX 2_ε sera donc aussi un passage progressif au *moteur* TEX. Ce passage pourra nécessiter quelques rares modifications de la part de l'utilisateur. Examinons les différents cas susceptibles de se produire. Nous avons soit affaire à des fichiers LATEX classiques, soit à de nouveaux fichiers LATEX 2_ε.

2.1 Les anciens fichiers LATEX

Ces fichiers ne nécessiteront aucune modification, sauf dans l'unique (et rare) cas où un paragraphe utilise une police classique (par exemple `cm...`) définie par `\font` ou `\newfont`. Dans ce cas, le paragraphe n'aura plus de divisions dans les mots accentués et les caractères accentués sortiront mal. Si les déclarations `\font` ou `\newfont` ne sont utilisées que pour quelques mots, sans accents, et où la division importe peu, la conversion n'est pas nécessaire. Si au contraire, une conversion est nécessaire, deux solutions peuvent être envisagées :

Première solution Pour rétablir les divisions et l'accentuation, une solution est de remplacer les déclarations `\font` ou `\newfont` par des déclarations similaires plus appropriées. Ainsi, à chaque police Computer Modern, il correspond une police DC où la division est possible. Par exemple, si on a écrit

```
\newfont{\GrasIt}{cmbxti10}
{\GrasIt Du texte en gras italique ...}
```

on peut remplacer ce passage par

```
\newfont{\GrasIt}{dcbxti10}
{\GrasIt Du texte en gras italique ...}
```

où *seule la déclaration a changé.*

Voici de manière détaillée les substitutions à réaliser, si ces polices sont utilisées *explicitement* pour des paragraphes français. On a encadré la substitution faite dans l'exemple précédent.

`cmb`	$\longrightarrow$	`dcb`	`cmsltt`	$\longrightarrow$	`dcsltt`
`cmbx`	$\longrightarrow$	`dcbx`	`cmss`	$\longrightarrow$	`dcss`
`cmbxsl`	$\longrightarrow$	`dcbxsl`	`cmssbx`	$\longrightarrow$	`dcssbx`
$\boxed{\texttt{cmbxti}}$	$\longrightarrow$	$\boxed{\texttt{dcbxti}}$	`cmssdc`	$\longrightarrow$	`dcssdc`
`cmcsc`	$\longrightarrow$	`dccsc`	`cmssi`	$\longrightarrow$	`dcssi`
`cmdunh`	$\longrightarrow$	`dcdunh`	`cmssq`	$\longrightarrow$	`dcssq`
`cmff`	$\longrightarrow$	`dcff`	`cmssqi`	$\longrightarrow$	`dcssqi`
`cmfi`	$\longrightarrow$	`dcfi`	`cmtcsc`	$\longrightarrow$	`dctcsc`
`cmfib`	$\longrightarrow$	`dcfib`	`cmti`	$\longrightarrow$	`dcti`
`cmitt`	$\longrightarrow$	`dcitt`	`cmtt`	$\longrightarrow$	`dctt`
`cmr`	$\longrightarrow$	`dcr`	`cmu`	$\longrightarrow$	`dcu`
`cmsl`	$\longrightarrow$	`dcsl`	`cmvtt`	$\longrightarrow$	`dcvtt`

Deuxième solution Une autre solution, qui permet de bénéficier de tous les avantages de LaTeX 2ε, est de convertir complètement le fichier, de façon à le faire commencer par `\documentclass`. Dans ce cas, il faut — comme on l'a dit — ajouter `\usepackage[T1]{fontenc}` (ou `\usepackage{t1enc}`). On se reportera au chapitre 1, pour plus de détails.

2.2 Les nouveaux fichiers LaTeX 2ε

Les nouveaux fichiers LaTeX 2ε sont ceux commençant par `\documentclass` et n'utilisant pas les commandes `\font` ou `\newfont` pour définir une police qui sera utilisée pour un paragraphe.

Si le document comporte des passages en français, ces passages devant être justifiés avec des divisions correctes, on chargera le *package* `fontenc` avec l'option `T1` au début du document, après `\documentclass`.

Chapitre 3

Les polices

Ce chapitre s'efforce de décrire l'environnement TEX du CRIN en matière de polices, à savoir celles qui sont installées et comment s'en servir. Nous y donnons de nombreux exemples ainsi que des pointeurs pour approfondir chaque question[19].

3.1 Généralités

3.1.1 Caractères utilisables dans TEX. METAFONT

À l'origine, les caractères utilisables étaient ceux décrits par un programme METAFONT, le langage de description de fontes associé à TEX. Voici, pour fixer les idées, un extrait de source METAFONT (tracé du guillemet français ouvrant (<<), tiré du fichier `guill.mf`) :

```
beginchar("A", .7em#, .7ht#, 0);
pickup pencircle;
penpos1(.75,0);penpos2(4,0);penpos3(.75,0);
penpos4(.75,0);penpos5(4,0);penpos6(.75,0);
rt x1 = rt x3 = .5w;rt x4 = rt x6 = w;
top y1 = top y4 = h-u;bot y3 = bot y6 = u;
lft x2= 0; y2 = y5 = .5h; lft x5= .5w;
filldraw z1l--z2l--z3l--z3r--z2r--z1r--cycle;
filldraw z4l--z5l--z6l--z6r--z5r--z4r--cycle;
endchar;
```

METAFONT est un programme qui engendre des *bitmaps* pouvant être utilisés par TEX, les visionneuses, les *drivers* d'imprimantes, etc. METAFONT, tout comme TEX, est un logiciel pouvant être utilisé soit interactivement, soit à partir d'un fichier. Depuis la version 3.0 de TEX, il est possible d'utiliser les fontes *PostScript* de manière très souple grâce au mécanisme des fontes virtuelles[20].

3.1.2 Quelles sont les informations nécessaires à TEX pour utiliser une fonte ?

Dans la mesure où TEX ne gère que le positionnement des caractères, il lui faut uniquement les informations de taille, les ligatures, le crénage (*pair kerning*) et les corrections italiques. Ces renseignements figurent dans

19. Dans ce document, nous nous efforçons d'employer les mots « police » et « fonte » dans le sens que leur attribue le dictionnaire « le Petit Robert » (1993), à savoir :

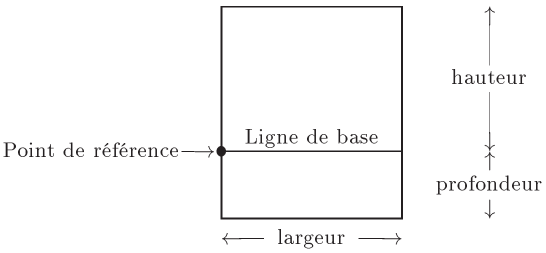
– En typographie, une « fonte » est un « ensemble de caractères d'un même type (fondus ensemble) ».

– Une « police » est « dans un corps déterminé, [un] assortiment complet [de] caractères de même graisse et de même famille ».

20. Il est toutefois inutile de connaître le principe de ces dernières pour utiliser des fontes *PostScript*.

le fichier ⟨*nom*⟩.tfm où ⟨*nom*⟩ est le nom de la fonte, par exemple, `cmr10.tfm` pour la fonte par défaut de (LA)TEX. TEX ne manipule que des boîtes noires et ne sait pas où ira l'encre.

Rappelons les notions de hauteur, profondeur, largeur et ligne de base pour un caractère :



(pour plus de détails, cf. The TEXbook [Knuth, 1984, p. 63])

- Les fichiers .tfm (TEX Font Metrics) standard se trouvent dans `/usr/local/lib/tex/fonts/tfm`.
- On peut en placer ailleurs, mais il faut dans ce cas positionner la variable `TEXFONTS` (cf. § 3.4.2).
- L'une des erreurs pouvant se produire est que TEX ne trouve pas un fichier .tfm. Le message d'erreur est similaire à :

  ```
  ! Font \titi=gogo not loadable: Metric (TFM) file not found.
  ```

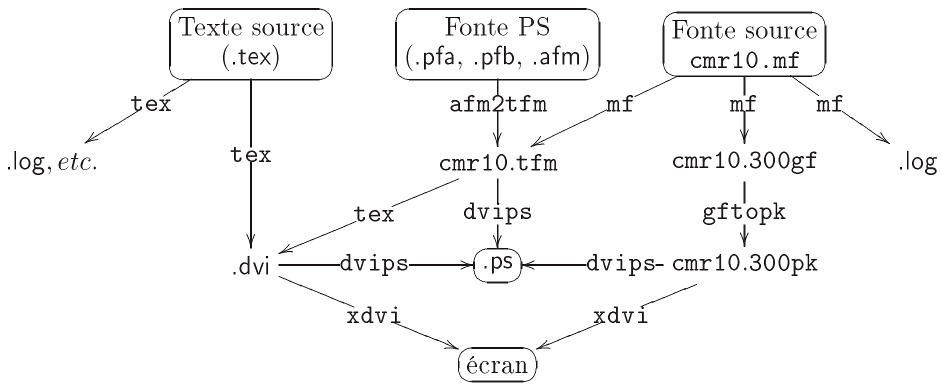
  Il y a essentiellement deux causes possibles :

  - la variable `TEXFONTS` est mal positionnée (cf. § 3.4.2),
  - le fichier .tfm n'existe pas : il faut alors l'engendrer à partir du source .mf, comme indiqué au § 3.4.1.

▷ Pour obtenir une description plus détaillée des fichiers .tfm, on se reportera à l'appendice F de The METAFONTbook [Knuth, 1986b].

3.1.3 Hiérarchie des fichiers

Le schéma suivant montre les principales interactions entre les fichiers. Certains de ces fichiers seront décrits ultérieurement. Les fontes virtuelles ne sont pas décrites ici.



Description des types de fichiers :

Extension	Description
.afm	Fichier de métrique des fontes *PostScript* (*Adobe Font Metric*)
.dvi	*Description Device Independent*
.xgf	*Generic Font file* (*bitmaps* des caractères) (ici, créés pour la résolution de 300 points par pouce)
.mf	Fichier source d'une fonte METAFONT
.pfa	Adobe Type 1 Format Ascii
.pfb	Adobe Type 1 Format Binaire
.xpk	*Packed bitmaps* (.xgf compressés)
.ps	*PostScript*
.tex	Source TEX, LATEX, etc.
.tfm	Fichier de métrique TEX contenant les tailles des caractères (*TEX Font Metric*)

3.1.4 Commandes tournant autour de METAFONT

Voici quelques pages de `man` qui comportent d'utiles renseignements : `mf`, `gftopk`, `dvips`, `xdvi`, `gftodvi`. Citons de manière plus générale les utilitaires `gftype`, `pktype`, `pktogf`, `tftopl`, `pltotf`. En particulier, les deux derniers permettent d'aller et venir entre un fichier .tfm et son contenu « en clair ».

3.2 Comment changer de fonte en TEX ou en LATEX

Il y a plusieurs solutions pour changer de fonte. On peut changer le « style » du document, appliquer une commande particulière, déclarer une nouvelle fonte, éventuellement en en modifiant la taille, etc.

3.2.1 Changement du style

Chaque association d'une classe, d'options et de *packages* LATEX détermine une « fonte standard ». Voici quelques associations typiques :

`\documentclass{article}` $\Longrightarrow$ `cmr10` (Computer Modern Roman 10pt)

`\documentclass[12pt]{article}` $\Longrightarrow$ `cmr12` (Computer Modern Roman 12pt)

`\documentclass{article}` $\Longrightarrow$ `Times Roman 10pt` (fonte *PostScript*)
`\usepackage{times}`

Comme les « styles » et options disponibles sont en nombre limité, on ne peut pas — à moins de créer un nouveau style — utiliser de cette manière une police *Computer Modern Roman 13pt* par exemple.

3.2.2 Modification de la fonte à l'intérieur du document

– taille (LATEX) : `\large`, `\Large`, `\Huge`, etc.

– taille, série, aspect (Plain TEX, PICTEX) : `\tenrm`, `\tenbf`, `\sevenrm`, etc.

– style : `\bf`, `\tt`, `\sc`, `\sl`, `\it`, etc. (commandes conservées pour la compatibilité avec LATEX 2.09)

– famille, série, aspect : `\rmfamily`, `\sffamily`, ..., `\bfseries`, ..., `\scshape`, ...

Attention : certaines combinaisons sont impossibles en standard (voir en § 1.7.3 et § 1.7.4 comment y remédier) ; par exemple `\bf\it` donne de l'italique sans conserver le gras.
Certains *packages* fournissent des commandes supplémentaires pour passer d'une fonte à l'autre.

3.2.3 Déclaration d'une nouvelle fonte

3.2.3.1 Commandes de bas niveau

Une nouvelle fonte est généralement déclarée lorsque la fonte en question n'est pas accessible par une commande prédéfinie. Par exemple, pour accéder à la fonte *Computer Modern Sans Serif 12pt*, on peut faire :

- en LaTeX : `\newfont{\toto}{cmss12}`.
- en TeX et LaTeX : `\font\toto=cmss12`

puis `{\toto un texte sans empattements}` qui donne un texte sans empattements en utilisant la fonte `cmss12`. NFSS fournit des commandes plus élaborées comme `\DeclareFixedFont` (cf. § 1.2.3) ou `\usefont` (cf. § 1.7.3.2).
Il est possible d'agrandir ou de réduire des fontes lors de la déclaration.[21] Voici quelques exemples :
`\font\toto=cmss12 scaled \magstep1` multiplie la taille de `cmss12` par 1,2
`\font\toto=cmss12 scaled 3141` multiplie la taille de `cmss12` par 3,141
`\font\titi=cmss12 scaled 900` multiplie la taille de `cmss12` par 0,9
`\font\lolita=cmr10 at 12pt` permet de spécifier la taille finale, plutôt que le facteur à appliquer.

Cette utilisation « brute » d'une fonte présente un inconvénient : la fonte ne s'adapte pas à la taille environnante. Par exemple, `{\Huge\lolita Nabokov}` et `{\small\lolita Nabokov}` donnent exactement le même résultat. Pour remédier à cela de manière simple, on peut faire appel à la nouvelle gestion des polices de LaTeX 2$_\varepsilon$ (cf. § 1.7.3.2).

Un certain nombre de tailles de fontes sont disponibles en standard, c'est-à-dire qu'il leur correspond un fichier. Pour les autres, c'est au moment de la génération du *PostScript* par `dvips` que METAFONT est activé pour créer les fichiers manquants. Il faut bien sûr que la fonte que l'on veut agrandir ou diminuer existe (cf. § 3.3 pour les fontes disponibles et § 3.4 pour en générer d'autres) !

3.2.3.2 Utilisation de *packages*

De nombreuses fontes viennent avec des *packages* effectuant toutes les déclarations nécessaires. Par exemple, pour pouvoir écrire 'METAFONT', il nous a suffi de charger le *package* mflogo et d'écrire `\MF`. On trouvera plus loin une liste des fontes disponibles et les *packages* correspondants, le cas échéant.

3.3 Polices METAFONT disponibles

Les fichiers sources se trouvent dans `/usr/local/lib/mf/inputs`. Leurs noms sont ici donnés sans extension. Il est sous-entendu que .mf est l'extension du fichier source et .tfm l'extension du fichier de métrique utilisé par TeX.

3.3.1 Tailles

Pour une fonte donnée, les corps disponibles forment en général un sous-ensemble de l'ensemble `5pt`, `6pt`, `7pt`, `8pt`, `9pt`, `10pt`, `12pt`, `14pt`, `17pt`. Mais rien n'empêche d'en créer de plus grandes ou de plus petites (cf. § 3.2.3.1).

3.3.2 Familles de fontes

Pour simplifier, on qualifiera une famille de fontes de « police ». Pour chaque fonte, une seule taille est citée ici. Le nom des fontes est donné de telle sorte qu'on puisse en déclarer une par `\font\essai=`⟨*nom*⟩ (par exemple).

21. Le nom des fontes ainsi déclarées ne doit comporter que des lettres. Pour plus de détails, cf. The TeXbook [Knuth, 1984, pp. 16–17].

3.3.2.1 Polices Computer Modern

On se reportera au TEXbook [Knuth, 1984] et au volume E de *Computers & Typesetting* [Knuth, 1986a], pour plus de détails sur la conception et l'utilisation de cette famille.
(fichiers `cmb10`, `cmbsy10`, `cmbx10`, `cmbxsl10`, `cmbxti10`, `cmcsc10`, `cmdunh10`, `cmex10`, `cmitt10`, `cmmi10`, `cmmib10`, `cmr10`, `cmsl10`, `cmsltt10`, `cmss10`, `cmssbx10`, `cmssdc10`, `cmssi10`, `cmssq8`, `cmssqi8`,[22] `cmsy10`, `cmtcsc10`, `cmtex10`, `cmti10`, `cmtt10`, `cmu10`, `cmvtt10`, `cmff10`, `cmfi10`, `cminch`, `cmfib8`).

Nous disposons aussi de certaines fontes ne faisant pas partie de la « distribution standard *Computer Modern* ». C'est le cas des petites capitales grasses `cmbcsc10`. En outre, la distribution AMSfonts complète les séries de tailles pour quelques unes de ces fontes (cf. § 3.3.2.6).

3.3.2.2 Polices DC

La famille Computer Modern et son codage n'ont jamais eu une prétention d'universalité. Ils étaient avant tout destinés à la langue anglaise. C'est pour cela que l'introduction des accents reste lourde. Suite à une réunion TEX à Cork en 1990, il a été décidé de concevoir un nouveau codage, plus adapté aux alphabets européens. C'est ce que l'on appelle actuellement les polices DC. Le standard futur devrait s'appeler EC (Extended Character Set). Les caractères accentués y correspondent à un code bien précis et ne sont pas formés comme jusqu'à présent de l'assemblage d'un caractère non-accentué et d'un accent. Ceci résout le problème des divisions dans les mots accentués, problème qui avait conduit Michael Ferguson à développer la variante MlTEX de TEX. Par conséquent, MlTEX est condamné à disparaître (cf. chapitres 1 et 2).
Les fontes DC sont des fontes 8 bits. Elles reprennent presque toute la famille Computer Modern (sauf les fontes mathématiques) mais les caractères accentués ne sont pour la plupart plus des caractères composés. Les fontes *Concrete* — qui ont aussi une variante DC — sont données plus loin.

Il y a une correspondance entre les noms des fontes CM et les versions DC : par exemple, à `cmbxsl10` correspond `dcbxsl10`, etc. (cf. chapitre 2) Pour utiliser les polices DC, on se servira du *package* `t1enc`.

3.3.2.3 Polices Concrete

Cette famille est une variante de Computer Modern introduite par Knuth pour l'ouvrage *Concrete Mathematics* [Graham *et al.*, 1989]. Elle existe aussi en 8 bits (fontes DC).
En voici un exemple : ceci est écrit en Concrete.

- Fichiers 7 bits : `cccsc10`, `ccmi10`, `ccr5`, `ccr6`, `ccr7`, `ccr8`, `ccr9`, `ccr10`, `ccsl10`, `ccslc9`, `ccti10`.
- Fichiers 8 bits : `dccr5`, `dccr6`, `dccr7`, `dccr8`, `dccr9`, `dccr10`, `dccr11`, `dccr12`, `dccr14`, `dccr17`, `dccsc10`, `dccsc12`, `dccsc17`.

Le *package* beton de LATEX 2_ε permet d'accéder facilement à ces fontes.

3.3.2.4 Polices Pandora

Exemple : La boîte de Pandore.
(fichiers `pnb10`, `pnr10`, `pnsl10`, `pnsltt9`, `pnss10`, `pnssb10`, `pnssi10`, `pntt9`)
Le *package* pandora de LATEX 2_ε permet d'accéder facilement à ces fontes.

3.3.2.5 Polices gothiques

Ces fontes sont destinées à représenter les différentes variantes de l'ancienne écriture allemande.

- `yfrak` Fraktur Die Niebelungen
- `ygoth` Gotisch Siegfrieds Tod

22. Pour accéder aux polices `cmssq`, c'est-à-dire les polices *Computer Modern Sans Serif Quotation*, on peut utiliser le *package* `ssqquote` de LATEX 2_ε. On trouvera plus de détails dans `ssqquote.dvi` (répertoire `/local/tex/archive/macros/latex/contrib/-supported/ssqquote`).

– `yswab` Schwabacher Kriemhilds Rache

– `yinit` Grandes initiales gothiques :

Le *package* oldgerm de LaTeX 2ε permet d'accéder facilement à ces fontes.

3.3.2.6 Polices AMS

La collection AMSfonts est un ensemble de polices réalisées par la société mathématique américaine (*American Mathematical Society*) pour ses publications.

Description :

Ces polices se subdivisent en quatre catégories :

– Famille Euler (gothique, script, etc.) : fichiers `euex10`, `eufb10`, `eufm10`, `eurb10`, `eurm10`, `eusb10`, `eusm10`. Cette famille a été dessinée par Hermann Zapf.

– Symboles mathématiques supplémentaires, y compris l'alphabet « Blackboard bold » pour les nombres réels ($\mathbb{R}$), naturels ($\mathbb{N}$), etc. : fichiers `msam10`, `msbm10`.

– Alphabet cyrillique : fichiers `wncyr10`, `wncyb10`, `wncyi10`, `wncysc10`, `wncyss10`.

– Autres tailles que celles dessinées par Knuth pour `cmmib`, `cmbsy`, `cmex`, `cmcsc`.

On pourra examiner plus précisément ces fontes en appliquant `tex` au fichier `userdoc.fnt` [23].

Utilisation :

– Si on sait dans quel fichier se trouve un caractère recherché, on peut écrire `\font\...` et y accéder par son code.

– On peut aussi écrire au début du fichier : `\input amssym.def` (en TeX ou LaTeX)
`\input amssym.tex`

– Une autre solution est d'utiliser le *package* amsfonts avec LaTeX 2ε (cf. § 1.9). De plus, les *packages* euscript et eufrak permettent d'accéder aux familles Euler script et fraktur.

Cette solution présente un avantage important : la taille des caractères *suit* la taille englobante, ce qui n'est pas le cas avec les deux autres solutions.

Les deux dernières possibilités définissent des commandes pour accéder aux caractères des polices `msam`, `msbm` et `eufm`. Par exemple, en écrivant `$\varnothing$`, on obtient '$\varnothing$'. Et `$\mathbb{Z}$` (`${\Bbb Z}$` avec l'ancienne syntaxe de LaTeX 2.09) donne '$\mathbb{Z}$'.

Installation d'une police mathématique :

Pour installer « proprement » une police mathématique comme `eufm` en TeX, on écrira par exemple :

```
\font\teneufm=eufm10
\font\seveneufm=eufm7
\font\fiveeufm=eufm5
\newfam\eufmfam
\textfont\eufmfam=\teneufm
\scriptfont\eufmfam=\seveneufm
\scriptscriptfont\eufmfam=\fiveeufm
\def\eufm#1{{\fam\eufmfam\relax#1}}
```

23. dans `/local/tex/archive/fonts/ams/amsfonts/doc`.

puis `\eufm{Texte en eufm}`.

L'installation d'une autre famille mathématique en TEX est similaire. L'installation d'une telle police en LATEX est encore plus simple (cf. chapitre 4).

Documentation :

▷ Le fichier `userdoc.tex`[24] décrit de manière détaillée cette collection. Il comporte en particulier la liste de tous les symboles définis dans `amssym.tex` et précise les conventions à observer pour l'écriture cyrillique.

3.3.2.7 Divers

1. ***Ralph Smith's formal script*** (RSFS)

   Cette police fournit une alternative à la police calligraphique usuelle donnée par `\mathcal`. On l'utilisera en général par l'entremise du *package* calrsfs qui redéfinit `\mathcal`. Par conséquent, `$\mathcal{F}$` ne donnera plus « $\mathcal{F}$ », mais « $\mathscr{F}$ ».

   (fichiers `rsfs*`)

2. **Bbold**

   Cette police est une alternative à l'utilisation des polices $\mathcal{AMS}$ pour représenter les ensembles de naturels, de réels, etc. En voici quelques exemples : $\mathbb{N}$, $\mathbb{R}$, $\mathbb{Z}$ (comparez la position des doubles barres avec les polices de l'$\mathcal{AMS}$).

   (fichiers `bbold*`)

   Cette police peut naturellement être utilisée avec LATEX 2ε et le *package* bbold. Celui-ci fournit les commandes `\textbb` et `\mathbb`. Ceci s'inscrit dans la lignée des commandes standard (cf. §1.7.4 et §1.7.5).

3. **Polices de XY-pic**

   Celles-ci sont utilisées par le *package* xypic (XY-pic) pour le tracé de diagrammes (cf. 5.1.2)

   (fichiers `xyatip10`, `xybsql10`, `xybtip10`, `xycirc10`, `xycmat10`, `xycmbt10`, `xydash10`)

4. **Logo METAFONT**

   Ce logo est disponible en graisse normale, en gras, en oblique et en demi-gras.

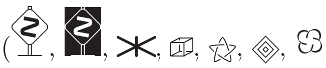
   (fichiers `logo10`, `logobf10`, `logosl10`, `logod10`)

   Le *package* mflogo de LATEX 2ε permet d'accéder de manière simple à cette police. En particulier, les commandes `\MF` et `\MP` y sont définies, pour respectivement METAFONT et METAPOST. Ce *package* est documenté dans `mflogo.ps`[25].

5. **Fichiers « propres » à METAFONT**

   Il s'agit des fichiers `slant`, `gray`, `black` qui sont utilisés par METAFONT pour certains modes particuliers (`proof mode`, `smoke mode`), ainsi que par `gftodvi`.

6. **Quelques caractères spéciaux utilisés dans le TEXbook et le METAFONTbook**

   (, etc.)

   Tous ces caractères font partie de la police `manfnt` (cf. [Knuth, 1984; Knuth, 1986b]).

7. **Fontes LATEX**

   (a) Fontes propres à LATEX : ces fontes sont essentiellement utilisées dans l'environnement `picture` et par la classe slides.
   (fichiers `circle10`, `circlew10`, `lasy10`, `lasyb10`, `lcircle10`, `lcirclew10`, `lcmss8`, `lcmssb8`, `lcmssi8`, `line10`, `linew10`)

24. dans `/local/tex/archive/fonts/ams/amsfonts/doc`.
25. dans `/local/tex/archive/macros/latex/contrib/supported/mflogo`.

(b) Fontes invisibles utilisées par la classe slides : ces fontes servent à la classe slides pour établir des transparents couleurs ou des *overlays* (cf. chapitre 6). Elles interviennent en particulier dans la commande `\invisible` de LaTeX 2.09 (cf. « *LaTeX—A document preparation system* » [Lamport, 1985, pp. 136–137]).

Dans LaTeX 2ε, ces polices sont encore utilisées, mais le texte est rendu invisible par le biais de `\textcolor{white}{...}` (*package* color) (cf. [Lamport, 1994, page 82]).

Rappelons que la classe slides remplace l'obsolète format `\SLiTeX`.

(fichiers `icmex10`, `icmmi8`, `icmsy8`, `icmtt8`, `ilasy8`, `ilcmss8`, `ilcmssb8`, `ilcmssi8`)

8. **Fonte** *calligraphique*.

   (fichier `callig15`)

9. **Alphabet phonétique international**

   (fichiers `wbxipa8`, `wbxipa9`, `wbxipa10`, `wbxipa11`, `wbxipa12`, `wbxipa17`,

   `wslipa8`, `wslipa9`, `wslipa10`, `wslipa11`, `wslipa12`, `wslipa17`,

   `wsuipa8`, `wsuipa9`, `wsuipa10`, `wsuipa11`, `wsuipa12`, `wsuipa17`)

   Pour utiliser ces fontes, on chargera les macros de `ipamacs.tex` par `\input ipamacs`, ou bien on utilisera LaTeX 2ε avec le package ipa (`\usepackage{ipa}`), ce qui a l'avantage de fournir des symboles variant en taille et en style.

   Exemple : la prononciation de « shampooing » est transcrite par [ʃɑ̃pwɛ̃]

   On trouvera de la documentation dans `lipaman.ps`[26] et `ipaman.dvi`[27].

10. **La police St Mary's Road.**

    (fichiers `stmary5`, `stmary6`, `stmary7`, `stmary8`, `stmary9`, `stmary10`)

    Cette police comporte de nombreux symboles intéressants comme : ⊗, ⊝, ⟆, ⇾, ↯, ⋎, etc. Elle est documentée dans `stmaryrd.dvi`[28].

    Le *package* stmaryrd définit des commandes pour accéder à ces symboles.

11. **Police Wasy.**

    (fichiers `wasy5`, `wasy6`, `wasy7`, `wasy8`, `wasy9`, `wasy10`, `wasyb10`)

    Cette police contient de nombreux symboles comme : ◷, ☎, ◯, ✡, ♪, ⌑, ☹, ♋, etc.

    Elle est documentée dans `wasydoc.ps`[29].

    Le *package* wasysym de LaTeX 2ε permet d'accéder facilement à ces caractères. Les noms des symboles peuvent être trouvés dans `wasysym.ps`[30].

De nombreux fichiers sources .mf existent de par le monde, en particulier pour les alphabets d'autres langues. Citons par exemple : caractères chinois, hébreux, japonais, sanskrits, pièces du jeu d'échecs, caractères pour écrire des partitions de musique, mains, etc. On en trouvera une liste très conséquente dans le fichier `metafonts.lst`[31]. Les adresses des sites `ftp` correspondants pourront être trouvées dans les FAQ (*Frequently Asked/Answered Questions*) pour TeX.

26. dans `/local/tex/archive/macros/text1/wsuipa`.
27. dans `/local/tex/archive/macros/latex/contrib/other/ipa`.
28. dans `/local/tex/archive/macros/latex/contrib/supported/stmaryrd`.
29. dans `/local/tex/archive/fonts/wasy2/wasy`.
30. dans `/local/tex/archive/macros/latex/contrib/supported/wasysym`.
31. dans `/local/tex/doc/Metafont`.

3.4 Comment générer et utiliser une nouvelle fonte avec METAFONT

La version 2.71 de METAFONT est installée. On y accède avec `mf`. Le principe d'utilisation est tout-à-fait similaire à TEX. Un fichier source METAFONT a normalement l'extension .mf. Tout comme un fichier source TEX, un fichier source METAFONT peut être découpé en plusieurs fichiers. Il y a un fichier *racine* et c'est à lui que l'on applique `mf`. C'est cette hiérarchie de fichiers et l'utilisation de descriptions paramétrées qui est à la base du concept de *méta-fonte*.

▷ Pour dessiner soi-même une police METAFONT, on se reportera à The METAFONTbook [Knuth, 1986b].

3.4.1 Un exemple

On peut aussi utiliser une *méta-fonte* et la modifier. Donnons un exemple très simple. Supposons que l'on veuille générer du gras *oblique* (bold slanted). Le plus rapide est de reprendre le fichier source (racine) du gras standard (`\bf`), à savoir `cmbx10.mf`, et de modifier son *paramètre d'inclinaison*. Plus précisément, l'une des lignes du fichier `cmbx10.mf` est `"slant:=0;"`. On remplacera par exemple `0` par `0.25`.

En renommant[32] ce fichier en `mycmbxsl10.mf`, en ajoutant la ligne `"mode=localfont;"` au début et en le donnant en argument à `mf`, on obtient `mycmbxsl10.tfm` et `mycmbxsl10.300gf`. Puis l'application de `gftopk` à `mycmbxsl10.300gf` donne `mycmbxsl10.300pk`, utilisable par `dvips`.

Ces opérations peuvent aussi être réalisées en deux lignes de la manière suivante :

```
mf '\mode=localfont; input mycmbxsl10.mf'
gftopk mycmbxsl10.300gf
```

Désormais, on peut écrire en gras *oblique* (ce n'est pas de l'italique) en écrivant `\font\bfsl=mycmbxsl10` au début du fichier TEX, puis `{\bfsl un texte gras oblique}`.

3.4.2 Variables d'environnement

Les variables d'environnement suivantes doivent être positionnées si on accède à des fichiers ne figurant pas dans les chemins de recherche standard :

- `TEXFONTS` (.tfm) Si cette variable existe, elle doit au moins contenir les chemins standard et les nouveaux chemins. Pour inclure les chemins par défaut, il suffit de mettre '`:`' à la fin du `setenv`. Exemple : `setenv TEXFONTS ~dek/tfm:`

  À titre d'indication, par défaut, les chemins de recherche des métriques sont :

  `.:/usr/local/lib/tex/fonts/tfm` (pour `tex` ou `latex`).
- `MFINPUTS` (.mf) par défaut : `.:/usr/local/lib/mf/inputs` (pour `mf`).
- `TEXPKS` (.pk) par défaut : `.:/usr/local/lib/tex/fonts/pk/pk`⟨*résolution*⟩`:/tmp/pk/pk`⟨*résolution*⟩, où ⟨*résolution*⟩ est la résolution à laquelle `dvips` engendre le .ps (300 ou 600 dpi actuellement).
- `XDVIFONTS` (.gf ou .pk) par défaut :[33] `.:/usr/local/lib/tex/fonts/pk/pk300:`

  `/tmp/pk/pk300:/usr/local/lib/tex/fonts/pk/pk600:/tmp/pk/pk600`

  (pour `xdvi`).
- `VFFONTS` (.vf) par défaut : `.:/usr/local/lib/tex/fonts/vf` (pour `dvips`).

3.4.3 Documentation

▷ D'autres paramètres que l'inclinaison peuvent être modifiés. On en trouvera une description précise dans le volume E de la série *Computers & Typesetting* [Knuth, 1986a]. Des introductions élémentaires à METAFONT sont `metafont-for-beginners.tex` et `MF.ps`[34].

32. Il est bon de prendre l'habitude de commenter le fichier modifié et d'en indiquer la provenance.
33. Ces chemins sont justifiés pour le cas d'une fonte pour laquelle on ne dispose pas d'un fichier source `.mf` mais d'un fichier `.pk` dans une seule résolution ; ce fichier `.pk` pourra alors être trouvé même si la visualisation se fait à une autre résolution.
34. tous deux dans `/local/tex/doc/Metafont`.

3.5 Fontes *PostScript*

3.5.1 Fontes disponibles

Les fontes *PostScript* utilisables sont soit les 35 fontes *PostScript* résidentes, soit celles pour lesquelles on dispose d'un fichier de métrique .afm et d'un fichier .pfa (respectivement .pfb), c'est-à-dire d'un fichier ASCII (respectivement binaire) d'une fonte *PostScript* Type 1. LaTeX 2ε utilise deux séries de polices *PostScript*, pour chacun des codages OT1 et T1.

Au total, nous disposons actuellement des polices AvantGarde, Bookman, Courier, Helvetica, New Century Schoolbook, Palatino, Symbol, Times, Zapf Chancery et Zapf Dingbats.

3.5.2 Utilisation des polices *PostScript*

3.5.2.1 Commandes de bas niveau

On peut déclarer une telle fonte avec `\font` ou `\newfont`, par exemple `\newfont{\Courier}{pcrr}` puis l'utiliser simplement avec `{\Courier Le facteur}`. On peut aussi, de cette manière, modifier la taille des fontes *PostScript* avec `scaled` (cf. § 3.2.3.1).

Toutefois, comme on l'a déjà montré, l'emploi de ces commandes conduit à une certaine rigidité.

3.5.2.2 Utilisation de *packages*

Une meilleure solution est d'utiliser un des *packages* suivants (cf. § 1.7.6) :

Packages	**Effet**
avant	Texte `sf` en AvantGarde.
bookman	Texte en **Bookman**.
chancery	Texte en *Zapf Chancery*.
courier	Texte `tt` en `Courier`.
helvet	Texte `sf` en Helvetica.
newcent ou ncs	Texte en New Century Schoolbook.
palatino	Texte en Palatino.
times	Texte en Times.
pifont	Symboles et *Zapf Dingbats*

Pour avoir un document en Times, on pourra écrire :

```
\documentclass[11pt]{article}
\usepackage{times}
...
```

Ces *packages* ne redéfinissent en fait que les familles `\rmfamily`, `\ttfamily` et `\sffamily`, et toutes les autres combinaisons s'en déduisent. Par contre le texte italique mathématique – c'est-à-dire ce que l'on obtient avec `\mathnormal` – reste en Computer Modern. En cas d'urgence, on peut utiliser la police `Times` pour des formules mathématiques grâce au *package* mathptm, mais le résultat est très moyen car cette police n'est pas conçue pour la typographie mathématique.

3.5.3 Documentation

▷ Pour plus de détails sur les fontes *PostScript*, cf. *Métrique des fontes* PostScript [André et Bur, 1991], et sur les fontes *PostScript* virtuelles, cf. *Virtual Fonts : More Fun for Grand Wizards* [Knuth, 1990]. On consultera aussi avec profit le manuel de `dvips` (c'est-à-dire `dvips.tex`[35]). Enfin, on pourra accéder à un fond très riche de polices *PostScript* du domaine public sur WWW[36].

35. dans `/usr/local/lib/tex/inputs`.
36. `http ://jasper.ora.com/Internet-Font-Archive.html`.

3.6 Comment trouver un caractère particulier ?

▷ Bien souvent un caractère est accessible par une commande. On se réfèrera aux tables données dans « The TeXbook » [Knuth, 1984], « *LaTeX—A Document Preparation System* » [Lamport, 1994], ...

3.6.1 Création de tables

On peut aussi visualiser tous les caractères d'une fonte avec `testfont.tex`. Voici un exemple d'utilisation pour engendrer une table de la fonte `eurm10`, qui fait partie des polices $\mathcal{AMS}$:

```
tex /usr/local/include/tex/inputs/testfont.tex
This is MlTeX, C Version 3.141
(/usr/local/include/tex/inputs/testfont.tex
Name of the font to test = eurm10
Now type a test command (\help for help):)
*\help

\init switches to another font;
\end or \bye finishes the run;
\table prints the font layout in tabular format;
    ...
\help repeats this message;
and you can use ordinary TeX commands (e.g., to \input a file).
*\table

*\end
[1]
(see the transcript file for additional information)
Output written on testfont.dvi (1 page, 8316 bytes).
Transcript written on testfont.log.
```

Ce fichier de test n'est pas adapté à des fontes 8 bits comme les polices DC. On appliquera dans ce cas LaTeX 2_ε à `nfssfont.tex`.

3.6.2 Accès individuel à un caractère

On peut aussi accéder à un caractère particulier en donnant simplement son code. Ainsi, pour avoir , on peut écrire `\font\manual=manfnt` puis `{\manual\char127}`, `{\manual\char'177}` ou `{\manual\char"7F}`, suivant que la position du caractère dans la fonte est donnée en décimal, octal ou hexadécimal. On peut aussi écrire `{\manual\symbol{127}}`.

3.7 Outils divers

Un certain nombre d'outils de conversion peuvent être utilisés :

– `bm2font` : il s'agit d'un programme permettant de convertir un certain nombre de formats *bitmap* (`PCX`, `GIF`, `BMP`, `IFF` ou `LBM`, `TIFF`, `IMG`, `CUT`, ...) en polices utilisables par TeX, ce qui a en particulier l'avantage de pouvoir les visualiser avec `xdvi`. Par exemple :

  `bm2font -m30 duck.gif`

  transforme `duck.gif` en caractères de police TeX de telle sorte que la largeur de l'image résultante soit 30mm. Cette opération génère `duck.tex` qui doit être inclus dans le fichier source avec `\input`. `\setduck` sera utilisé pour placer l'image là où on souhaite la faire apparaître. Il existe de nombreux paramètres permettant d'agir sur les niveaux de gris, etc. Pour de plus amples informations, on se reportera au manuel (`bm2fmane.dvi`[37]).

37. dans `/local/tex/archive/graphics/bm2font/manual`.

- `fig2MF` : il s'agit d'un programme convertissant un fichier .fig en un fichier METAFONT utilisant les macros de `mfpic`. Exemple d'utilisation : si on a un fichier `avion.fig`, on écrit :

```
fig2MF avion.fig > avion.mf
mf '\mode=localfont;input avion'
```

  ce qui donne un fichier `avion.tfm` directement utilisable par TEX. Si la position du caractère créé est 33, on pourra utiliser le fichier TEX suivant :

```
\font\test=avion
{\test\char33}
\bye
```

  Pour de plus amples informations, on se reportera au manuel (`fig2MFman.dvi`[38]).

3.8 Polices et divisions des mots

Dans TEX, les mots comportant des caractères accentués *composés* ne peuvent pas être coupés. Cette limitation a été supprimée dans MlTEX. Toutefois, l'apparition de nouvelles polices où les caractères accentués sont monolithiques (les polices DC) permet enfin d'utiliser TEX pour le français. LATEX 2$_\varepsilon$ tourne sur un moteur TEX (cf. chapitres 1 et 2) et on doit donc presque obligatoirement utiliser les polices DC pour du français, etc. Rappelons que ces polices peuvent être sélectionnées avec `\usepackage[T1]{fontenc}` ou `\usepackage{t1enc}`.

38. dans `/local/tex/archive/graphics/fig2mf`.

Chapitre 4

BibTeX et les références bibliographiques

Ce chapitre donne quelques conseils sur l'utilisation conjuguée de BibTeX et de LaTeX et sur la manière de créer, d'éditer et de gérer correctement des fichiers .bib.

4.1 Quelques rappels

Les utilisateurs novices sont souvent étonnés, voire irrités, par le nombre de passages nécessaires avec LaTeX avant et après un passage de BibTeX. Cela s'explique pourtant aisément par le fait que LaTeX travaille en une seule passe sur le texte. Prenons un exemple : voici quelques extraits d'un source LaTeX ; appelons-le `toto.tex` :

```
\documentclass[twocolumn]{article}
\usepackage{french}
...
\begin{document}
...
\bibliographystyle{plain}
...
... ce qui est encore un problème ouvert~\cite{joseph92a}.
...
\bibliography{image}
\end{document}
```

Le fichier `image.bib` correspondant contient quelque part l'enregistrement suivant :

```
@ARTICLE{joseph92a,
  AUTHOR = {S.H. Joseph and T.P. Pridmore},
  TITLE = {{Knowledge-Directed Interpretation of Mechanical Engineering Drawings}},
  YEAR = {1992},
  JOURNAL = {IEEE Transactions on PAMI},
  PAGES = {928--940},
  NUMBER = {9},
  VOLUME = {14},
  MONTH = {September}
}
```

Au premier passage sur `toto.tex`, LaTeX note le style de présentation souhaité pour les références bibliographiques (`plain`) et le fait qu'il est fait appel à une référence de clé `joseph92a`. Celle-ci lui étant pour l'instant inconnue, vous avez un message d'erreur du genre :

```
LaTeX Warning: Citation ‘joseph92a’ on page 1 undefined on input line 138.
```

Les informations de ce type glanées au passage sont écrites dans le fichier `toto.aux`, où l'on retrouve donc entre autres les lignes suivantes :

```
\bibstyle{plain}
\citation{joseph92a}
...
\bibdata{image}
```

L'étape suivante est alors de lancer BibTeX sur le fichier par la commande `bibtex toto`. BibTeX va alors examiner le fichier `toto.aux` pour trouver les clés demandées, le nom du fichier bibliographique dans lequel on souhaite chercher les références correspondantes (en l'occurrence `image.bib`, qui doit se trouver dans un des chemins donné par la variable d'environnement `BIBINPUTS`) et le style de présentation choisi par l'auteur, qui lui permet de choisir le bon fichier de style bibliographique (qu'il recherche dans les répertoires indiqués par la variable d'environnement `BSTINPUTS`, ou `TEXINPUTS` si la précédente n'est pas définie), en l'occurrence `plain.bst`, comme indiqué par la commande `\bibliographystyle`. À partir de toutes ces informations, BibTeX construit le fichier `toto.bbl`, qui contiendra donc entre autres les lignes suivantes :

```
\bibitem{joseph92a}
S.H. Joseph and T.P. Pridmore.
\newblock {Knowledge-Directed Interpretation of Mechanical Engineering
  Drawings}.
\newblock {\em IEEE Transactions on PAMI\/}, 14(9):928--940, September 1992.
```

C'est ce fichier qui est inclus lors d'un passage ultérieur de LaTeX, au moment où celui-ci rencontre l'instruction `\bibliography{image}`. Or regardons de plus près ce qui se passe au deuxième passage de LaTeX sur `toto.tex`. Au moment où la citation `\cite{joseph92a}` est analysée, le fichier `toto.bbl` n'a pas encore été inclus et le fichier `toto.aux` ne contient donc aucune information permettant de remplacer la citation par le bon numéro. Cette information n'est écrite dans `toto.aux` qu'à la lecture de `toto.bbl`, déclenchée par l'instruction `\bibliography` :

```
...
\bibcite{joseph92a}{5}
...
```

Maintenant le tour est joué, et au troisième passage de LaTeX (le deuxième après le passage de BibTeX), la citation peut être remplacée par une clé numérique, en l'occurrence `[5]`.

Pour résumer ce mécanisme, le schéma suivant montre quels fichiers sont créés ou changés à chaque étape :

$$\xrightarrow{toto.tex} \texttt{latex} \xrightarrow{toto.aux} \texttt{bibtex} \xrightarrow{toto.bbl} \texttt{latex} \xrightarrow{toto.aux} \texttt{latex} \xrightarrow{toto.dvi}$$

$$\begin{matrix} image.bib \\ plain.bst \end{matrix} \uparrow$$

4.2 Compléments d'information sur BibTeX

Pour l'utilisation de BibTeX, nous renvoyons le lecteur en premier lieu au livre de Lamport [Lamport, 1994] et au *LaTeX Companion* [Goossens *et al.*, 1994, chapitre 13]. Pour ceux qui ne connaissent de BibTeX que ce qu'en dit la première édition de Lamport, il faut savoir que cet outil a évolué depuis ; par conséquent, il nous semble bon de donner ici quelques « trucs » sur BibTeX, tirés de la documentation « Bib*TeXing* », de Oren Patashnik, disponible en ligne dans `btxdoc.tex`[39] et dont la lecture est vivement conseillée.

- Si vous voulez citer toutes les références d'un fichier (typiquement pour avoir une sortie papier propre de votre fichier favori `biblio.bib`), il suffit d'écrire `\nocite{*}`.

39. dans `/usr/local/lib/tex/inputs`.

– Concaténation de chaînes :

```
@STRING(REC = "Le livre des records ")
...
@BOOK(records89,
  TITLE = REC # 1989,
  ...
@BOOK(records92,
  TITLE = REC # 1992,
  ...
```

– À l'heure actuelle, BibTeX n'accepte pas encore les caractères 8-bits, et il faut donc les accentuer avec les moyens classiques. Les caractères accentués dans un champ (par exemple `AUTHOR`), pour être correctement pris en compte dans la construction d'étiquettes, doivent être mis comme suit :

```
AUTHOR = {F. L{\'e}otard}
```

– Les types `@PHDTHESIS` et `@MASTERTHESIS` admettent maintenant un champ `TYPE`, ce qui permet d'enregistrer une thèse française sans utiliser `@TECHREPORT` ou `@MISC`. Il suffit de mettre « Thèse de doctorat » dans le champ `TYPE` :

```
TYPE = {Th{\`e}se de doctorat},
SCHOOL = {Universit{\'e} Henri Poincar{\'e} Nancy~I}
```

– S'il y a trop d'auteurs pour que l'on désire mettre la liste complète, on peut terminer une liste d'auteurs par « `and others` » ; les styles standard savent convertir ceci en « *et al.* » ou quelque chose de ce genre quand on cite une telle entrée.

Rappelons par ailleurs que dans tous les cas les auteurs doivent être séparés par le mot clé '`and`' et non par des virgules, des '`et`' ou d'autres séparateurs.

– Les caractères de commentaires LaTeX (%) ne sont **pas** des caractères de commentaire dans un fichier BibTeX. Toutefois, tout ce qui est compris entre deux enregistrements est considéré comme du commentaire par BibTeX.

– Cas des noms d'auteurs compliqués : les trois formes sous lesquelles on peut mettre un nom d'auteur sont :

  – Prénom particule Nom
  – particule Nom, Prénom
  – particule Nom, Jr, Prénom

BibTeX sachant distinguer une (ou des) particule(s) éventuelle(s) par le fait qu'elle(s) commence(nt) par une minuscule. La première forme convient en fait dans la grande majorité des cas.

4.3 Liste des styles bibliographiques disponibles

Tous les styles bibliographiques sont des fichiers `XXXX.bst`[40]. Pour chaque style, les version fXXXX correspondent à des versions francisées (« rédacteur » au lieu d'« *editor* », « et » au lieu de « and », etc.). Les styles sont écrits dans un langage à pile postfixé qui est documenté dans `btxhak.tex`[40], dont nous conseillons la lecture à ceux qui voudraient écrire leur propre style bibliographique ou même opérer des modifications non triviales à un style existant. Une autre possibilité est d'utiliser le programme TeX `makebst.tex`[41], avec documentation dans `makebst.dvi`[41]. Ce programme vous pose toute une série de questions et fabrique un style bibliographique sur mesure à partir de vos réponses à ces questions.

Nous passons en revue dans ce qui suit les principaux styles existants sur le site. Parfois, nous illustrons ce style en montrant le type de présentation qu'il produit sur un exemple. Il va sans dire qu'à moins d'allonger

40. dans `/usr/local/lib/tex/inputs`.
41. dans `/local/tex/archive/macros/latex/contrib/supported/custom-bib`.

démesurément ce chapitre, nous ne pouvons pas illustrer toutes les particularités de chaque style et que les exemples choisis n'en couvrent sûrement qu'une partie. Nous invitons donc le lecteur à faire ses propres expériences pour compléter les « pointeurs » donnés ici.

4.3.1 Famille *plain*

Les clés engendrées sont des numéros, de la forme [3]. Variantes :

– plain et fplain : version de base :

> [5] S.H. Joseph and T.P. Pridmore. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) :928–940, September 1992.

– abbrv et fabbrv : version avec abréviations standard définies pour les journaux d'informatique.

– acm : style *ACM Transactions*, proche de abbrv mais avec les noms en petites capitales et sous la forme « Nom, initiales », et certaines dates entre parenthèses :

> [5] JOSEPH, S., AND PRIDMORE, T. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI 14*, 9 (September 1992), 928–940.

– amsplain : une variante de plain par *American Mathematical Society* ; si plusieurs références qui se suivent ont les mêmes auteurs, ceux-ci ne sont pas répétés mais remplacés par un trait :

> [5] S.H. Joseph and T.P. Pridmore, *A System for the Interpretation of Images of Graphics*, Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA), 1990, pp. 154–164.
>
> [6] ______, *Knowledge-Directed Interpretation of Mechanical Engineering Drawings*, IEEE Transactions on PAMI **14** (1992), no. 9, 928–940.

– unsrt et funsrt : version de plain où les références ne sont pas triées par ordre alphabétique mais apparaissent dans l'ordre dans lequel elles sont citées.

– ieeetr : style correspondant aux *IEEE Transactions*, proche de unsrt :

> [1] S. Joseph and T. Pridmore, "Knowledge-Directed Interpretation of Mechanical Engineering Drawings," *IEEE Transactions on PAMI*, vol. 14, pp. 928–940, September 1992.

– siam : variante de plain, noms en petites capitales, titres en italiques :

> [5] S. JOSEPH AND T. PRIDMORE, *Knowledge-Directed Interpretation of Mechanical Engineering Drawings*, IEEE Transactions on PAMI, 14 (1992), pp. 928–940.

4.3.2 Famille *alpha*

Les clés sont de la forme [ABC92]. Variantes :

– alpha et falpha : un auteur donne [Tom91], deux auteurs [VT90], trois auteurs [ACT90], etc.

> [JP92] S.H. Joseph and T.P. Pridmore. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) :928–940, September 1992.

– alpha3 et falpha3 : on prend les trois premières lettres du premier auteur quel que soit le nombre d'auteurs :

> [Jos 92] S.H. Joseph and T.P. Pridmore. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) : 928–940, September 1992.

– amsalpha : une variante de alpha par *American Mathematical Society* :

> [JP92] S.H. Joseph and T.P. Pridmore, *Knowledge-Directed Interpretation of Mechanical Engineering Drawings*, IEEE Transactions on PAMI **14** (1992), no. 9, 928–940.

– beta3, fbeta3 et apbeta3 : une variante de alpha3 sans blanc entre lettres et chiffres ([Jos92]) ; apbeta3 est une toute petite variante supplémentaire (un espace blanc ajouté avant les numéros de pages).

4.3.3 Famille *long*

Les clés sont formées par un ou des noms complets d'auteurs. Variantes :

– long et flong : nom complet du premier auteur :

[Joseph 92] S.H. Joseph and T.P. Pridmore. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) :928–940, September 1992.

– realfullname et frealfullname : ajoutent **et al.** s'il y a deux auteurs ou plus. Présentent les auteurs et certains autres champs différemment de long :

[Joseph et al.92] Joseph (S.H.) et Pridmore (T.P.). – Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, vol. 14 (9), September 1992, pp. 928–940.

– apalike[42] : pas d'étiquettes dans la bibliographie mais une liste triée où chaque entrée apparaît de la manière suivante :

Joseph, S. and Pridmore, T. (1992). Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) :928–940.

À utiliser avec le *package* apalike pour que les commandes `\cite` engendrent des chaînes de la forme `(Joseph and Pridmore, 1992)`.

Un certain nombre d'autres styles existent dans cette même veine, avec plus ou moins de fonctionnalités suivant les cas, et des *packages* associés pour engendrer soit des citations du type précédent, soit des citations plus courtes, par exemple `(1992)`, `(Joseph)`, etc. grâce à des commandes telles que `\fullcite`, `\shortcite`, `\citeyear`, etc. Pour plus de précisions, nous vous conseillons de vous reporter au mode d'emploi de ces *packages*, qui est habituellement inclus en commentaires au début du fichier. Nous vous donnons ici les pointeurs uniquement :

- authordate{1,2,3,4}, quatre variantes mineures (voir commentaires dans les fichiers .bst correspondants) à accompagner du *package* authordate1-4.
- chicago et sa variante avec annotations chicagoa, à utiliser avec le *package* chicago.
- theapa et newapa avec les *packages* de même nom.

À notre avis, tous ces styles de présentation de bibliographie ne sont pas à conseiller dans l'absolu, car ils n'utilisent pas de clés aussi claires et non ambiguës que les autres, mais beaucoup de maisons d'édition exigent encore ce genre de présentation, probablement parce que c'est une survivance du passé, où n'existaient pas encore les moyens informatiques puissants qui nous permettent de renuméroter toutes les références sans perte de cohérence quand on ajoute ou qu'on enlève une entrée.

– named et fnamed : un peu comme long, avec des clés qui, suivant le nombre d'auteurs, prennent les formes : [Preiss, 1984], [Joseph and Pridmore, 1992] ou [Tsujio *et al.*, 1992] :

[Joseph and Pridmore, 1992] S.H. Joseph and T.P. Pridmore. Knowledge-Directed Interpretation of Mechanical Engineering Drawings. *IEEE Transactions on PAMI*, 14(9) :928–940, September 1992.

[Preiss, 1984] K. Preiss. Constructing the Solid Representation from Engineering Projections. *Computers and Graphics*, 8(4) :381–389, 1984.

[Tsujio *et al.*, 1992] S. Tsujio, T. Ono, and S.S. Lee. Computer-Aided Drawing Check for CAD Systems — A Method for the Checking of Dimensions in Multiview Mechanical Drawings. In *Proceedings of IEEE International Conference on Systems Engineering, Kobe (Japan)*, pages 234–237, September 1992.

À utiliser avec le *package* named, qui permet de varier les types de citations (cf. § 4.4).

4.3.4 Divers

Voici des styles que nous n'avons pas testés (bogués ou non documentés), mais qui existent sur notre site : fullname, newalpha. Il existe aussi beaucoup d'autres styles récupérables par `ftp` sur des sites archives, en particulier pour répondre aux présentations exigées par un certain nombre de revues.

42. *Like the American Psychology Association.*

4.4 Quelques options ou packages LATEX utiles pour les citations

Le choix d'un style bibliographique est bien entendu primordial pour la présentation des références bibliographiques. Mais vous pouvez aussi modifier les présentations de la bibliographie ou des références à celle-ci par un choix judicieux de *packages* ou d'options. En voici quelques-uns disponibles sur notre site ; comme d'habitude, nous ne donnons que quelques pointeurs, et une lecture attentive de la partie commentaire des fichiers correspondants est souvent conseillée pour une maîtrise complète de leur fonctionnement. Si nous ne précisons, on a affaire par défaut à un *package* (c'est le cas le plus fréquent) :

breakcites permet la division d'une citation dans le texte, ce qui est très pratique en particulier avec les styles engendrant des clés longues.

cite remplace une série de trois ou plus numéros consécutifs par l'intervalle correspondant (par exemple, `[4,5,6,7,9,8]` est remplacé par `[4-7,9,8]`).

citesort est une variante du précédent qui en plus trie les références (par conséquent, sur l'exemple précédent, le résultat serait `[4-9]`).

rangecite fait en gros la même chose que citesort.

overcite est une autre variante de cite qui remplace par des intervalles et met la citation en exposant, comme ici[4−7,8,9].

supercite met aussi les citations en exposant, mais sans créer d'intervalles.

draft est un *package* utile en cours de mise au point, et qui entre autres marque dans la marge les clés de citation utilisées par un `\cite`. Il ne sert d'ailleurs pas uniquement à la bibliographie et nous vous conseillons de lire son entête pour une documentation plus détaillée.

named s'utilise en conjonction avec le style bibliographique named ; il permet plusieurs types de citations dans le texte :

- `\cite{key}` a le comportement normal → `[Joseph and Pridmore, 1992]`,
- `\shortcite{key}` ne donne que l'année → `[1992]`,
- `\citeauthor{key}` donne seulement l'information auteur, sans crochets → `Joseph and Pridmore`,
- `\citeyear{key}` ne donne que l'année → `1992`

Comme indiqué précédemment, le même type de fonctionnalités existe pour un certain nombre d'autres styles bibliographiques de la famille *APA*.

spacecites ajoute des blancs entre les citations quand plusieurs d'entre elles sont incluses dans une même paire de crochets : `[Furu86,Furu87]` → `[Furu86, Furu87]`.

openbib est une option et non un *package* ; elle permet d'insérer un retour à la ligne après chaque bloc principal dans la bibliographie. Ainsi, avec le style bibliographique plain, l'exemple qui nous accompagne tout au long de ce chapitre donne avec l'option openbib :

[5] S.H. Joseph and T.P. Pridmore.
Knowledge-Directed Interpretation of Mechanical Engineering Drawings.
IEEE Transactions on PAMI, 14(9) :928–940, September 1992.

Attention : pour que openbib fonctionne, il faut que le style bibliographique que vous utilisez sépare les différents blocs principaux, dans votre fichier toto.bbl, par l'instruction `\newblock`, ce qui n'est pas le cas de tous les styles mentionnés au § 4.3.

chapterbib permet de créer plusieurs bibliographies dans un même ouvrage. Pour cela, chaque partie de votre source qui doit avoir sa propre bibliographie doit être incluse par une commande `\include` (pas `\input` !) et les commandes `\bibliography` et `\bibliographystyle` doivent être données dans ces fichiers inclus et pas dans le fichier maître.

bibunits permet aussi de créer plusieurs bibliographies séparées, une par *unité* de texte, l'étendue d'une unité pouvant soit être définie de manière globale par la commande `\bibliographyunit` (exemple : `\bibliographyunit{\chapter}` pour que chaque chapitre soit une unité) ou explicitement en ouvrant des environnements `bibunit`.

4.5 Manipuler un fichier BibTeX

Pour manipuler un fichier BibTeX, plusieurs outils sont disponibles sur notre site :

- Le *package* VORTEX sous *emacs* (cf. § 9.3) contient un utilitaire très puissant et convivial[43] pour manipuler des fichiers de références bibliographiques au format BibTeX.
- Il existe un certain nombre de logiciels du domaine public, sous X-Window, pour manipuler des fichiers BibTeX. Citons notamment `bibcard` et `bibview`.
- l'utilitaire `xbibgest`[44] développé et maintenu sur notre site en s'inspirant fortement de `bibcard`, fournit un moyen de gérer un fichier bibliographique, y compris des outils pour imprimer proprement un fichier ou une partie de fichier, des moyens de consulter suivant différentes clés, etc. Cet outil peut travailler en deux modes : un mode de gestion bibliographique standard et un mode réservé à la gestion des publications du Crin. Voir la documentation en ligne de ce logiciel pour plus d'information.
- `bibclean`[45] formatte de manière agréable un fichier BibTeX et signale les erreurs de syntaxe éventuelles.

43. Pour ceux qui n'ont pas peur de mémoriser les séquences de caractères de contrôle *emacs*.
44. © Crin/Cnrs.
45. Pas installé pour l'instant mais qu'on peut trouver dans `/local/tex/archive/biblio/bibtex/utils/bibclean`.

Chapitre 5

L'insertion des figures et des dessins

Cette partie présente les moyens disponibles sur le site loria pour créer et insérer une figure ou un dessin dans un document LaTeX. Pour plus d'informations, le lecteur est fortement invité à consulter les chapitres 10 (*Portable Graphics in LaTeX*) et 11 (*Using PostScript*) du *LaTeX Companion* [Goossens *et al.*, 1994], ou le chapitre 7 (*Pictures and Colors*) du manuel de référence [Lamport, 1994].

Rappelons qu'un document engendré à partir de TeX est transformé en un fichier Dvi (*device independent*) par les commandes `tex` ou `latex`, par exemple. Ensuite, pour que ce fichier puisse être imprimé, il doit être converti dans le langage de l'imprimante, en l'occurrence *PostScript* pour toutes les imprimantes de notre site. C'est la commande `dvips` qui réalise cette opération (cf. § 9.4.1 et § 9.4.2).

Tout d'abord, LaTeX lui-même permet de créer des figures directement, par ses primitives ou par l'intermédiaire de *packages* disponibles sur notre site. Nous donnerons quelques pointeurs sur ces possibilités, qui ont l'avantage principal de donner du graphique complètement portable, c'est-à-dire indépendant des moyens d'impression.

Par ailleurs, la commande `dvips` permet également d'insérer simplement des figures créées à l'aide des outils disponibles sur le réseau loria, tels que `tgif` ou `xfig`, ou quelque autre graphique ou image que ce soit disponible sous forme *Encapsulated PostScript*, ou même *PostScript* tout court, mais dans ce dernier cas il faudra parfois préciser les dimensions du graphique. Ces figures sont positionnées dans le source LaTeX grâce à la commande `\special`, qui peut être employée directement mais qui est habituellement « encapsulée » dans un *package*.

5.1 Écrire du graphique en LaTeX pur

5.1.1 L'environnement picture

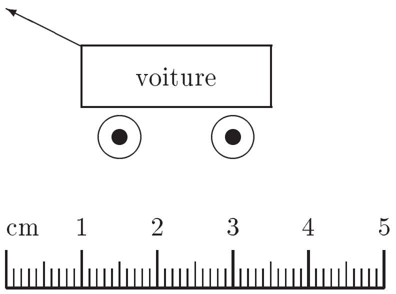
L'environnement `picture` est disponible en standard en LaTeX et permet de réaliser des dessins en traçant des traits, des flèches, des cercles et des boîtes. Voir le manuel de référence [Lamport, 1994, page 118] ou la section 10.2 de [Goossens *et al.*, 1994].

Les cercles ont un diamètre limité et les pentes des droites ne peuvent pas non plus avoir n'importe quelle pente (nombre rationnel dont le numérateur et le dénominateur sont en valeur absolue inférieurs ou égaux à 6, ou 4 pour les flèches). Il est bien sûr possible d'insérer librement du texte dans les dessins, ce texte utilisant par défaut la police courante. Cependant, les dessins sont limités (pas de courbes, pas de motifs complexes. . .) et assez fastidieux à définir.

La figure 5.1 donne un exemple de dessin et le code correspondant.

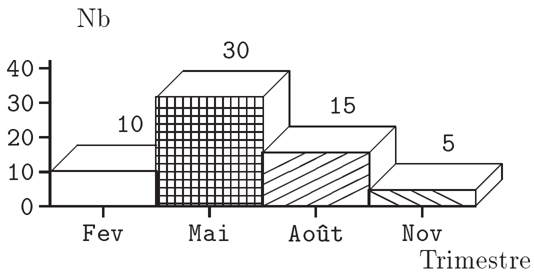
Un certain nombre de *packages* ont été développés au-dessus de `picture`. Citons :

– epic réduit le nombre de calculs nécessaires pour positionner correctement les objets graphiques qu'on définit, grâce à des commandes d'un peu plus haut niveau et à la syntaxe plus souple, pour tracer des traits de différents types. Son extension eepic ajoute des moyens de tracer des cercles, des ellipses et des *splines*, de donner des attributs d'épaisseur aux traits et de remplir des zones fermées, etc. Voir § 10.3 et § 10.4 de [Goossens *et al.*, 1994] pour plus de détails.




```
\newcounter{cms}
\setlength{\unitlength}{1mm}
\begin{picture}(50,39)
\put(15,20){\circle{6}}
\put(30,20){\circle{6}}
\put(15,20){\circle*{2}}
\put(30,20){\circle*{2}}
\put(10,24){\framebox(25,8){voiture}}
\put(10,32){\vector(-2,1){10}}
\multiput(10,7)(10,0){5}{\addtocounter{cms}{1}
\makebox(0,0)[b]{\arabic{cms}}}
\put(0,7){\makebox(0,0)[bl]{cm}}
\multiput(1,0)(1,0){49}{\line(0,1){2,5}}
\multiput(5,0)(10,0){5}{\line(0,1){3,5}}
\thicklines
\put(0,0){\line(1,0){50}}
\multiput(0,0)(10,0){6}{\line(0,1){5}}
\end{picture}
```

FIG. 5.1 - *Un exemple de dessin avec l'environnement* `picture` *et le code LATEX correspondant.*




```
\begin{barenv}
\setdepth{10}         % effet 3-D
\setstretch{1.4}      % éch. verticale
\setnumberpos{up}     % valeur
\setxvaluetyp{month}  % mois en X
\setxaxis{2}{12}{3}\setxname{Trimestre}
\setyaxis{0}{40}{10}\setyname{Nb}
\bar{10}{1}   \bar{30}{4}
\bar{15}{6}   \bar{5}{7}
\end{barenv}
```

FIG. 5.2 - *Exemple d'histogramme réalisé avec le* package bar.

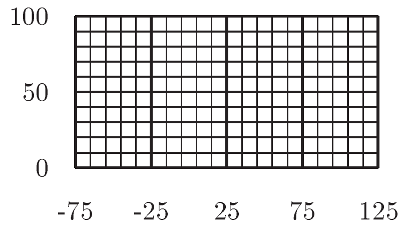
- bezier définit la commande `\bezier` pour tracer des courbes du même nom. Il faut noter que cette commande est incluse en standard dans LaTeX 2$_\varepsilon$ et que ce *package* n'est plus présent que pour des besoins de compatibilité.

- trees permet de définir des arbres binaires et ternaires.

- bar permet de réaliser des histogrammes, à l'aide de l'environnement `barenv`. La figure 5.2 en donne un exemple. Voir § 10.2.4 de [Goossens *et al.*, 1994] pour une liste complète des nombreuses fonctionnalités de ce *package*.

- curves permet de tracer un certain nombre de courbes. Voir § 10.2.6 de [Goossens *et al.*, 1994] pour une description.

- graphpap propose la commande `\graphpaper`, qui permet de dessiner une grille, bien utile pour la détermination des coordonnées des objets. Exemple :




```
\setlength{\unitlength}{0.2mm}
\graphpaper(-75,0)(200,100)
```

5.1.2 XY-pic

Le *package* xypic permet de réaliser des schémas de toute sorte. Un exemple d'utilisation de XY-pic est visible au § 3.1.3. Plutôt qu'un long discours, mieux vaut se reporter au fichier `xyrefer.dvi`[46], qui propose le manuel de référence de XY-pic (et qui comporte de nombreux exemples et des exercices), et à `xyguide.dvi`[46], qui est un manuel utilisateur de XY-pic.

5.1.3 Les diagrammes

Le *package* diagrams permet de réaliser des diagrammes commutatifs. Voici un exemple réalisé à l'aide de ce *package* :

$$\begin{tikzcd}
A*(B*(C*D)) \arrow[r, "\mathrm{assl}"] \arrow[d, "\mathrm{id}*\mathrm{assl}"'] & (A*B)*(C*D) \arrow[r, "\mathrm{assl}"] \arrow[d, phantom, "="] & ((A*B)*C)*D \\
A*((B*C)*D) \arrow[rr, "\mathrm{assl}"] & & (A*(B*C))*D \arrow[u, "\mathrm{assl}*\mathrm{id}"']
\end{tikzcd}$$

qui correspond à :

```
\def\Id{{\rm id}}\def\Assl{{\rm assl}}%
\begin{diagram}[width=6em]
A*(B*(C*D))&\relax\rTo^\Assl&(A*B)*(C*D)&\relax\rTo^\Assl&((A*B)*C)*D\\
\dTo^{\Id*\Assl} &       &   =   &      &\relax\uTo_{\Assl*\Id}\\
  A*((B*C)*D)  &       &\relax\rTo^\Assl &      &  (A*(B*C))*D\\
\end{diagram}
```

46. dans `/local/tex/archive/macros/generic/diagrams/xypic/doc`.

ou encore ce schéma :

$$\begin{tikzcd}
U \arrow[dr, "{(x,y)}"] \arrow[drr, "x"] \arrow[ddr, "y"'] & & \\
& X\times_Z Y \arrow[r, "p"] \arrow[d, "q"'] \arrow[dr, phantom, "\lrcorner", very near start] & X \arrow[d, "f"] \\
& Y \arrow[r, "g"] & Z
\end{tikzcd}$$

qui est réalisé par cette séquence de commandes :

```
\begin{diagram}
U \\
  &\relax\rdTo~{(x,y)}     \rdTo(4,2)^x
    \rdTo(2,4)_y  \\
  &         & X\times_Z \SEpbk Y & \rTo_p & X      \\
  &         & \dTo^q             &        & \dTo_f \\
  &         & Y                  & \rTo^g & Z      \\
\end{diagram}
```

Pour plus d'informations, on peut consulter le manuel `diagrams-manual.dvi`[47].

5.2 Inclure du graphique dans un document

Nous allons maintenant traiter du cas de l'insertion dans un document LaTeX d'un graphique produit par d'autres moyens. Au fil des années, un certain nombre de méthodes plus ou moins perfectionnées ont été mises au point et diffusées pour inclure des graphiques (principalement du *PostScript*) dans un fichier LaTeX. La sortie de LaTeX 2_ε a été l'occasion de « remettre à plat » ce problème et de proposer une solution générique, fournie par les *packages* graphics et graphicx.

Le format graphique utilisé à 99% sur notre site étant *PostScript*, il nous semble utile, avant d'étudier ce *package*, de préciser la notion de *BoundingBox*.

5.2.1 *BoundingBox*

Pour faciliter l'inclusion de figures *PostScript* dans des documents composés par des formatteurs de texte, Adobe a défini le format de fichier *Encapsulated PostScript* (Eps), ayant souvent une extension de fichier .eps. Ce format définit les règles standard d'importation de fichiers au format *PostScript* dans d'autres environnements.

Notre propos n'est pas de documenter complètement ce format Eps, mais d'en expliquer une caractéristique particulièrement utile pour l'inclusion de figures en LaTeX[48]. Le formattage d'une page par LaTeX consistant en un positionnement de boîtes les unes par rapport aux autres, il faut que LaTeX connaisse les dimensions de la boîte à réserver au graphique, pour que le texte englobant ne vienne pas se superposer sur celui-ci au moment de la conversion de Dvi en *PostScript*. Les *packages* d'inclusion de graphique ont justement pour rôle de trouver les dimensions de cette boîte, pour réserver l'espace nécessaire, et d'engendrer des commandes de type `\special` pour l'inclusion proprement dite par le *driver*.

Or, un fichier *PostScript* classique ne contient pas forcément les informations nécessaires au calcul de cette boîte. Le format Eps, en revanche, permet de définir les dimensions du graphique à inclure, grâce à la commande `BoundingBox`, que l'on utilise de la manière suivante :

```
%%BoundingBox: llx lly urx ury
```

47. dans `/local/tex/archive/macros/generic/diagrams/taylor`.
48. Attention : nous ne prétendons **absolument pas** qu'il suffit d'ajouter une *BoundingBox* pour qu'un fichier *PostScript* devienne un fichier Eps !

Cette ligne indique la taille du document *PostScript* à inclure, l'unité étant le point *PostScript* (1/72e par pouce). Les coordonnées du coin inférieur gauche (respectivement du coin supérieur droit) de la figure sont `llx lly` (respectivement `urx ury`).

Par exemple, pour une figure qui commence en $(100, 200)$, qui est un carré de 4 pouces de côtés (288 points *PostScript*), nous aurons :

```
%%BoundingBox: 100 200 388 488
```

Il est de bon goût d'ajouter systématiquement un ou deux points aux coordonnées pour être certain que l'ensemble de la figure est traité. Il faut noter que deux utilitaires locaux, `alltops` et `bbps`, créent la *BoundingBox* automatiquement (cf. § 9.6).

5.2.2 Les packages **graphics** et **graphicx**

Ces deux *packages* sont en fait issus de la même souche, mais offrent deux interfaces différentes. Ils fournissent un moyen générique d'inclure du graphique, même non *PostScript*. L'adaptation à toutes sortes de formats graphiques se fait grâce à des commandes `\DeclareGraphicsRule` attachant à un suffixe des actions particulières. Par exemple :

```
\DeclareGraphicsRule{ps}{eps}{ps.bb}{'epsfig #1}
\DeclareGraphicsRule{eps}{eps}{eps.bb}{'epsfig #1}
\DeclareGraphicsRule{ai}{eps}{ai.bb}{'epsfig #1}
\DeclareGraphicsRule{epsi}{eps}{epsi.bb}{'epsfig #1}
\DeclareGraphicsRule{psi}{eps}{psi.bb}{'epsfig #1}
```

dit que les fichiers de suffixe .ps, .eps, .ai, .epsi et .psi sont tous en fait du *PostScript* encapsulé et que le *driver* devra lancer la commande `epsfig` <*nom-du-fichier*>. Quand une extension n'est pas connue, le format pris par défaut est *PostScript*. Si vous manipulez régulièrement des fichiers de certains types, vous pouvez inclure les commandes `\DeclareGraphicsRule` dans un fichier `graphics.cfg`[49], qui sera chargé automatiquement par les *packages* graphics ou graphicx.

Au chargement de ces *packages*, on peut aussi spécifier en option quel *driver* on utilise. Par exemple, avec `dvips`, on utilisera l'option dvips :

```
\usepackage[dvips]{graphics}
```

Voici un exemple d'inclusion de graphique :

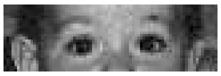
```
\includegraphics{floriane.eps}
```

Dans cet exemple la taille de la boîte est spécifiée dans le fichier `floriane.eps` lui-même, par la *BoundingBox* `%%BoundingBox : -65 688 68 846`. Si le fichier n'avait pas contenu de *BoundingBox*, LaTeX aurait produit un message d'erreur, et il aurait fallu spécifier les dimensions dans les options de la commande `\includegraphics`.

Il est également possible de spécifier dans la commande `\includegraphics` les dimensions à réserver pour la figure. Si celle-ci est plus grande que l'espace réservé, elle va déborder sur le texte avoisinant. Si on

49. Dans un des répertoires de `TEXINPUTS`.

veut au contraire qu'elle soit « découpée » pour tenir dans l'espace spécifié, il faut utiliser la commande `\includegraphics*`, comme dans cet exemple :



```
\includegraphics*[-40,785][40,810]{floriane.eps}
```

Le package graphics permet également de modifier l'échelle d'un texte grâce aux commandes `\scalebox` et `\resizebox`.

`\scalebox` permet d'élargir ou de rétrécir un texte par un facteur d'échelle constant :

- `\scalebox{2}{Il est...}` → Il est doublement grand ce guide.
- Il est normalement grand ce guide.
- `\scalebox{.5}{Il est...}` → Il est minable ce guide.

Un second argument, optionnel, permet d'appliquer un facteur particulier à l'échelle verticale.

`\scalebox{0.75}[2]{Il est...}` → Il est surtout en hauteur ce guide.

La commande `\resizebox` permet de fixer la taille d'un texte. Les deux arguments donnent la largeur et la hauteur du texte. L'utilisation de `!` comme argument permet de ne pas modifier les proportions :

- `\resizebox{7cm}{.4cm}{Il est...}` → Il est super ce guide.
- `\resizebox{2cm}{!}{Il est...}` → Il est super ce guide.

Pour plus de précisions sur graphics et graphicx, en particulier sur leurs interfaces respectives, on se reportera à `grfguide.ps`[50].

Epsfig

Les utilisateurs locaux ont probablement été habitués dans le passé à utiliser des *packages* tels que psfig, epsf ou epsfig. Il n'y a plus de raison d'utiliser ces anciens *packages*, sauf dans le cas particulier de la classe seminar, qui est expliqué plus loin. Pour ceux qui ont pris l'habitude de l'interface conviviale des anciens *packages* psfig et de epsfig, un petit *package* appelé epsfig est disponible dans la nouvelle distribution ; il se contente d'ajouter à graphicx l'interface à laquelle vous êtes habitués, à savoir :

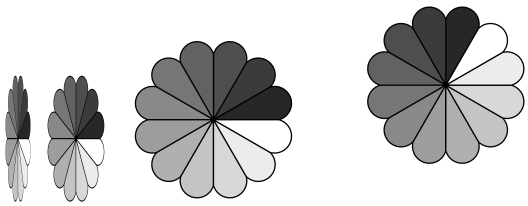
```
\begin{figure}
\epsfig{figure=dessin.ps}
\caption{Titre de la figure.}
\end{figure}
```

les commandes `\psfig` et `\epsfig` étant en l'occurrence équivalentes et correspondant à une simple redéfinition de la commande `\includegraphics`.

Par exemple, la séquence suivante :

```
\centerline{\hbox{
\epsfig{figure=rosette,height=2cm,width=4mm}
\epsfig{figure=rosette,height=2cm,width=9mm}
\epsfig{figure=rosette,height=2.5cm}
\epsfig{figure=rosette,height=2.5cm,angle=60}
}}
```

donne :



50. dans `/local/tex/doc/LaTeX2e`.

5.2.3 Un cas particulier : la classe seminar

La classe seminar (cf. § 6.2) permet de préparer à partir du même fichier source des transparents et un support de cours, l'ensemble des transparents étant dans ce dernier cas réduits d'une certaine échelle, qui peut être définie par l'utilisateur. Pour que les graphiques inclus suivent ces réductions, le *package* d'inclusion de figures doit gérer correctement le changement d'échelle (*scaling*). Au départ, epsf était préconisé par l'auteur de seminar. En fait, l'ancien style mineur de nom epsfig mariait l'interface de psfig avec les capacités de changement d'échelle de epsf.

C'est là qu'est le piège quand on passe à LaTeX 2ε. Comme nous l'avons vu, le *package* nommé epsfig n'est plus maintenant qu'une redéfinition des commandes de graphicx, et celles-ci ne gèrent pas pour l'instant le changement d'échelle, du moins pas de la manière prévue par seminar[51].

Quand vous préparez des transparents avec seminar, vous pouvez donner les ordres suivants pour indiquer les différences d'échelle entre l'option article et l'option slidesonly de seminar :

```
\slidesmag{5}
\articlemag{1}
```

Si en plus vous écrivez :

```
\epsfslidesize
```

vous aurez un comportement convivial avec l'ancien epsfig ou avec epsf. En effet, il suffit alors d'écrire :

```
\centerline{\epsfig{figure=mafigure.eps,height=10cm}}
```

et les 10 centimètres sont la hauteur *sur le transparent* (en `\slidesmag{5}`). Quand on formatte ensuite avec l'option article, les figures sont bien réduites pour rester à la bonne dimension... Cela donne donc une manière très naturelle de donner ses tailles de figure.

Cette réduction d'échelle ne fonctionnant plus avec le nouveau *package* epsfig, nous avons décidé de garder **pour cet usage** l'ancien *package*, sous le nom de oldepsfig. L'utilitaire local `skeltex` (cf. § 9.3) tient compte de cet état de fait et propose par défaut oldepsfig pour un document de classe seminar.

Ceci est bien sûr une solution transitoire, et nous passerons à une solution unique pour toutes classes dès que possible.

5.2.4 Le package color

Il est possible de réaliser des documents contenant du texte coloré à l'aide de commandes telles que `\color` et `\textcolor`, fournies par le *package* color. Exemples :

- `... UNE \textcolor{red}{PIPE} ROUGE.` → CECI N'EST PAS UNE PIPE ROUGE.
- `{\color{red} CECI ... {\color{black}PIPE} NOIRE.}` → CECI N'EST PAS UNE PIPE NOIRE.

Il est également possible d'écrire un texte dans une boîte de couleur grâce à la commande `\colorbox` :

`\colorbox{red}{UNE PIPE ROUGE}` → UNE PIPE ROUGE

On peut aussi mettre un cadre de couleur autour d'une boîte d'une autre couleur qui contient elle-même un texte :

`\fcolorbox{blue}{yellow}{UN EXEMPLE}` → UN EXEMPLE

Il est même possible de mettre un fond de couleur à toute une page par la commande `\pagecolor{`*couleur*`}`.

La commande `\definecolor` permet de définir ses propres couleurs, en plus de celles qui sont définies par défaut, et ceci selon différents modèles de couleurs : cmyk, rgb ou gray.

Cela étant, on peut vouloir sortir des versions intermédiaires sur une imprimante ne traitant pas la couleur. Pour cela, il faut utiliser le modèle de couleurs `gray` à la place de `rgb` :

Une boîte gris sombre. Une boîte gris clair.

51. L'auteur de cet ensemble de *packages*, David P. Carlisle, et celui de seminar, Timothy Van Zandt, ont promis qu'ils allaient s'accorder sur une solution à ce problème, mais pour l'instant, ce n'est pas fait.

Pour plus de précisions sur le *package* color, voir `grfguide.ps`[52] ou [Lamport, 1994, chapitre 7]. Un exemple d'utilisation pour les transparents est également donné en § 6.1. Nous conseillons aussi vivement la lecture d'un article paru dans les *Cahiers GUTenberg* [Goossens et Rahtz, 1994a], bien que le *package* color à proprement parler n'était pas encore disponible au moment où cet article a été écrit.

5.2.5 *PSTricks* : les astuces *PostScript*

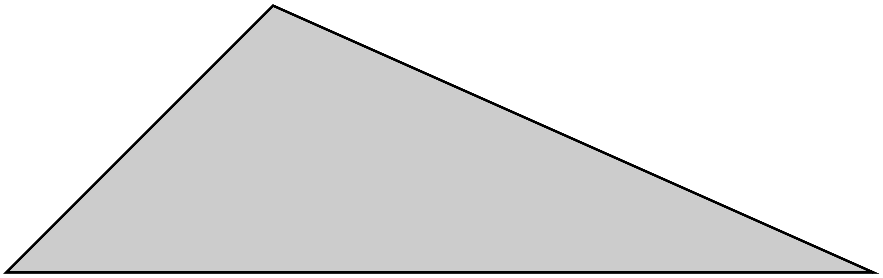
Le *package* pstricks est une collection des macros *PostScript* pour TeX, LaTeX, AMS-TeX, et AMS-LaTeX. Il a été développé par Timothy Van Zandt. On peut trouver une documentation dans le répertoire `/local/-tex/archive/graphics/pstricks/docs`. Nous conseillons aussi la lecture d'un article fort intéressant et très bien illustré dans les *Cahiers GUTenberg*, consacré à pstricks [Girou, 1994].

Les commandes de pstricks permettent de manipuler des couleurs[53] des graphiques, des arbres et des recouvrements, et de faire des rotations. L'utilisation de ces commandes nécessite l'inclusion d'un ou plusieurs des *packages* suivants : pst-node, pst-plot, pst2eps. Le *package* principal, pstricks, doit être inclus dans tous les cas.

Mieux qu'une longue énumération, la figure 5.3 donne quelque morceaux choisis réalisés à l'aide de pstricks, juste pour vous mettre l'eau à la bouche.

5.2.6 Le *PostScript* litéral

Notons pour terminer qu'il est possible d'écrire en *PostScript* litéral au sein d'un document LaTeX. Pour cela, il suffit d'utiliser une commande `\special` et de la faire débuter par une double quote (`"`). Par exemple le petit triangle suivant :



est obtenu par la commande :

```
\vbox to 100bp{\vss % insertion directe en \PS{}
\special{" newpath 0 0 moveto 100 100 lineto 325 0
lineto closepath gsave 0.8 setgray fill grestore stroke}}
```

5.3 Quelques utilitaires associés

- `tgif` crée des fichiers qui peuvent être insérés dans un texte à condition de bien les convertir en EPS. Pour cela, il faut sauvegarder le fichier objet (.obj) par la commande classique de sauvegarde (`^S`), et ensuite créer le fichier EPS à l'aide de la commande d'impression (`^P`), qui va incorporer la *BoundingBox* nécessaire (`^` représente la touche CONTROL).

  À noter que pour produire des caractères accentués avec `tgif`, on peut utiliser la touche ESCAPE suivie d'un autre caractère. Quelques exemples :

Pour produire :	&	é	à	è	ê
taper ESCAPE puis :	&	i	‘	h	j

52. dans `/local/tex/doc/LaTeX2e`.
53. Pour utiliser conjointement *PSTricks* et la couleur, il faut pour l'instant charger le *package* pstcol à la place de pstricks, en attendant que ce dernier soit modifié pour s'aligner sur le *package* color.

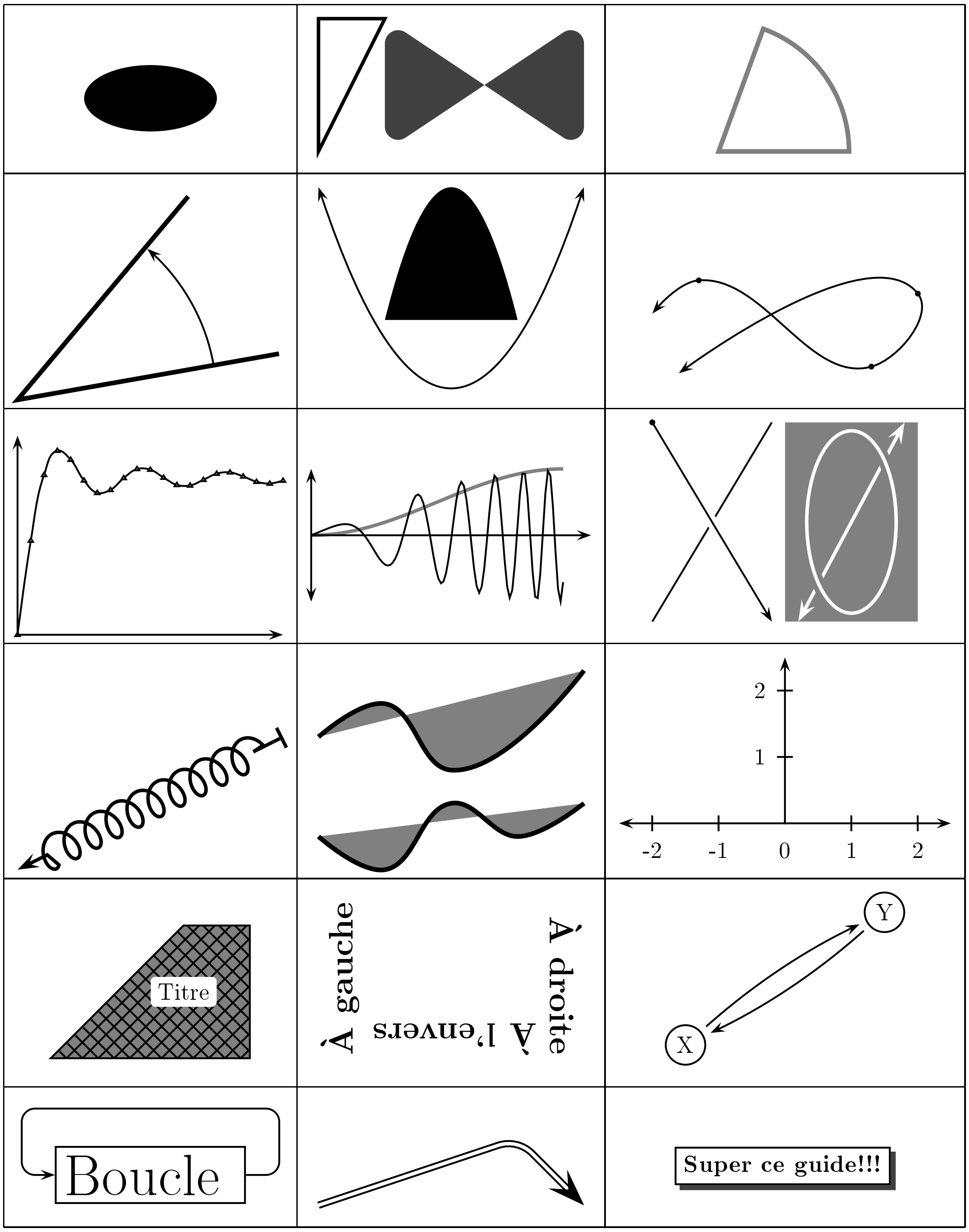


FIG. 5.3 - *Morceaux choisis en PSTricks.*

- `xfig` est un utilitaire qui permet à un utilisateur de dessiner et manipuler des objets interactivement à travers un environnement graphique. Les résultats du dessin peuvent être inclus dans un document LaTeX. Pour cela, il y a trois solutions possibles :
  - sélectionner un langage adéquat dans le menu `export` de `xfig`, le meilleur choix étant bien sûr le *PostScript* encapsulé ;
  - utiliser `fig2MF`, qui convertit un fichier .fig en un fichier METAFONT (cf. § 3.7) ;
  - utiliser l'outil `transfig`, qui permet de créer un `Makefile` automatisant la mise à jour de figures en utilisant les outils `fig2dev` (équivalent à `fig2ps`) et `fig2ps2tex`. La commande `transfig` a la syntaxe suivante :

    ```
    transfig [options]...[[control]...filename]...
    ```

    l'option la plus utile étant `-L` <*langage*>, le langage cible pouvant être `pictex` (pris par défaut), `epic`, `eepic`, `eepicemu`, `latex`, `ps`, `psfig`, `pstex`, `textyl`, `tpic`, ou `null`. Pour plus de précisions, voir la documentation dans `manual.dvi` [54].
- `gnuplot` permet d'engender divers formats, dont *PostScript*, des environnements `picture` LaTeX, ou des commandes *PSTricks*.
- METAPOST est une variante de METAFONT qui engendre du *PostScript*. La documentation se trouve dans `mpman.ps` [55].
- `bm2font` convertit un certain nombre de formats *bitmap* en polices utilisables par TeX (cf. § 3.7).
- mfpic est un ensemble de macros METAFONT permettant d'engendrer des graphiques sous forme de polices [56].

54. dans `/local/tex/archive/graphics/transfig/doc/manual`.
55. dans `/local//tex/loria/MetaPost/doc`.
56. Voir dans `/local/tex/archive/graphics/mfpic`.

Chapitre 6

Les transparents avec LaTeX

Il existe plusieurs possibilités pour réaliser des transparents avec LaTeX. Avant LaTeX 2ε, on disposait de manière standard du format SliTeX. Cette solution est toujours possible mais il s'agit maintenant de la classe slides. Une autre solution très en vogue est l'emploi du *package* seminar, lequel est fortement lié au *package* pstricks. Bien d'autres systèmes existent, comme foiltex, mais nous ne les détaillons pas ici.

6.1 La classe slides

C'est la classe standard pour réaliser des transparents. On notera qu'il y a un certain nombre de différences par rapport à SliTeX :

- LaTeX est utilisé pour compiler les anciens fichiers SliTeX ; auparavant, ceux-ci devaient être compilés par SliTeX.
- Dans les nouveaux fichiers utilisant la classe slides (commençant donc par `\documentclass{slides}`), on n'utilise plus qu'un fichier maître (la commande `\blackandwhite` a été supprimée).

Cette classe est détaillée dans la seconde édition du manuel de référence [Lamport, 1994]. En voici deux exemples d'utilisation très simples :

6.1.1 Transparents colorés

On utilise le *package* color (cf. § 5.2.4).

```
\documentclass{slides}
\usepackage[dvips]{color}
\begin{document}
\definecolor{light-blue}{rgb}{0.8,0.85,1}
\begin{slide}
{\color{red}This is red...}\\
{\color{green}This is green...}\\
{\color{blue}This is blue...}\\
{\color{light-blue}This is light blue...}\\
{\color[rgb]{0.5,0.8,0.2}What is this ?}\\
{\color[cmyk]{0.5,0.8,0.2,0.6}And what is this ?}\\
{\color[gray]{0.5}Some gray ?}\\
{\color{light-blue}This is light blue (not \colorbox{red}{blue})...}\\
{\color{yellow}Even yellow exists...}\\
Magenta looks like \textcolor{magenta}{this}.
\end{slide}
\end{document}
```

6.1.2 Exemple d'*overlay*

L'environnement `overlay` permet de superposer son contenu au transparent précédent. Si l'on souhaite cacher un mot alternativement, on pourra le mettre en blanc (après avoir chargé le *package* color). Exemple :

```
\begin{slide}
 ...
 bleu et \textcolor{white}{rouge}
 ...
\end{slide}

\begin{overlay}
 ...
 \textcolor{white}{bleu} et rouge
 ...
\end{overlay}
```

6.1.3 Compatibilité avec SLITEX

Pour des raisons d'existence de polices et de moteur TEX (le moteur MlTEX n'est plus utilisé avec LATEX 2ε), la compatibilité avec SLITEX n'est pas totale. Il n'est plus possible d'obtenir *simplement* de coupures dans des mots comportant des caractères accentués. Mais bien sûr, il est déconseillé — pour des questions de lisibilité — de couper des mots sur un transparent. De la même façon, il est déconseillé d'utiliser des polices à empattements. La police par défaut de la classe slides est en revanche réputée pour sa grande lisibilité.

Note pour les experts: *comme cela a déjà été dit, le codage utilisé par LATEX 2ε est par défaut le codage OT1 ; cependant, lorsque LATEX 2ε est appliqué à un document LATEX2.09, on force le codage T1,* sauf *pour les anciens fichiers* SLITEX *où le codage OT1 est maintenu.*

6.2 La classe seminar

La classe seminar est une alternative très séduisante à slides pour réaliser des transparents. Écrite par Timothy Van Zandt, également l'auteur de *packages* tels que fancybox ou pstricks (cf. § 5.2.5), elle offre un grand nombre de possibilités intéressantes.

Cette classe étant fournie avec un manuel d'utilisation très bien fait et plusieurs exemples de transparents[57], nous n'en dirons pas grand chose ici. Il est cependant bon de savoir que l'utilitaire local `skeltex` (cf. § 9.3) propose un « squelette » assez complet de fichier de classe seminar quand on choisit l'option adéquate.

Notons cependant quelques caractéristiques de seminar :

- Par défaut, les transparents sont en mode *landscape* quand ils sont créés avec un environnement `slide` et en mode portrait avec l'environnement `slide*`. L'option portrait permet de faire du mode portrait le mode par défaut.

- N'est pris en compte, quand on formatte les transparents, que ce qui se trouve dans un environnement `slide` ou `slide*`. Mais avec l'option article, les transparents sont réduits par réduction d'échelle (*scaling*, cf. § 5.2.3) et ce qui se trouve en-dehors des environnements de type `slide` est formatté comme un article. Cela permet donc de préparer un seul fichier avec les transparents et des commentaires supplémentaires, utiles pour faire un support de cours par exemple.

- L'interfaçage avec fancybox permet de mettre des cadres divers autour des transparents. Comme dans slides, la gestion des couleurs est aussi prise en compte. Pour l'instant, elle utilise les mécanismes définis dans *PSTricks* (cf. § 5.2.5), mais il est prévu de la faire bientôt reposer sur le *package* color (cf. § 5.2.4).

- La classe seminar est également bien interfacée avec les inclusions de graphique (le mieux étant un utilitaire contrôlant correctement la réduction d'échelle, dite *scaling*). Voir à ce sujet la discussion en § 5.2.3.

57. Voir dans `/local/tex/archive/macros/latex/contrib/other/seminar/doc`.

– La classe fournit une option semhelv permettant de faire de Helvetica la police utilisée dans les transparents. Nous ne pouvons que recommander l'emploi de cette option, une police sans empattements[58] comme Helvetica ou CMSS (option semlcmss) passant beaucoup mieux à la rétro-projection. En effet, les polices avec empattements (comme Times, etc.) ne laissent pas apparaître leurs détails à grande distance.

– L'option a4 a remplacé l'option sem-a4 décrite dans certaines versions de la documentation.

En plus du manuel cité précédemment, nous invitons le lecteur à se reporter à un article paru dans les *Cahiers GUTenberg* [Goossens et Rahtz, 1994b], qui décrit bien les possibilités offertes par seminar.

6.3 Foiltex

Actuellement, `foiltex` est un format TEX séparé, mais une classe foiltex pour LATEX 2$_\varepsilon$ est annoncée. L'ancienne version de `foiltex` est documentée dans `/local/tex/archive/macros/foiltex`.

58. *Sans serif* en anglais.

Chapitre 7

Principales règles typographiques

Dans ce petit aide-mémoire pour la rédaction de documents, en particulier de documents à caractère scientifique, nous n'avons pas voulu nous livrer à une étude comparative des conventions typographiques des langues — français et anglais — utilisées dans notre communauté, mais avons préféré proposer des mémentos *faciles à consulter*, au prix de certaines redites en ce qui concerne les conventions communes aux deux langues. Aussi, le lecteur trouvera en § 7.1 (resp. § 7.2) les conventions du français (resp. de l'anglais). Le plan est à chaque fois identique :

1° règles pour l'espacement, et règles liées à la coupure des lignes ;

2° règles de ponctuation (incluant les citations) et emploi des polices ;

3° emploi des lettres capitales[59].

À noter que les règles d'écriture de capitales pour certains mots se trouvent à la langue concernée. Ensuite, nous donnons au § 7.3 des tableaux des principales abréviations utilisées. Quant aux facilités offertes par le *package* french[60], elles sont signalées tout au long des paragraphes relatifs aux conventions françaises.

En outre, les règles qui ne font pas l'unanimité parmi les typographes sont repérées par une barre verticale noire en marge.

Au § 7.4, nous expliquons comment mélanger, au sein d'un même document, des fragments de textes écrits dans diverses langues. On y découvrira en particulier le futur *package* multilingue de LaTeX 2_ε. Enfin, nous donnons dans le § 7.5 des références à l'intention des lecteurs qui désireraient en savoir plus.

7.1 Typographie française

7.1.1 Espacement et coupure de lignes

Les signes de ponctuation suivants :

`. , : ; ! ?`

sont suivis d'une espace[61]. Il est d'usage, en français, d'insérer une espace légère insécable[62] avant les signes de ponctuation doubles : deux-points, points-virgules, points d'exclamation et d'interrogation. Ce qui peut être effectué par la commande `\,` comme suit[63] :

```
Par Jupiter\,!
```

qui produira :

Par Jupiter !

59. *Capitale* est le terme technique, en typographie, pour « lettre majuscule ». Les lettres minuscules forment le *bas de casse*.
60. Le *package* french a été développé par Bernard Gaulle, de l'association GUT*enberg* (**G**roupe francophone des **U**tilisateurs de TeX) et certaines de ses caractéristiques sont décrites dans [Rolland, 1993, § 8.2] et [Goossens *et al.*, 1994, § 9.3.2].
61. Le mot *espace* est féminin en français lorsqu'il désigne le caractère d'espacement.
62. C'est-à-dire que cette espace ne peut pas être remplacée par un passage à la ligne suivante.
63. Dans tout ce qui va suivre, les commandes LaTeX et les exemples de sources seront écrits en utilisant la police de caractères « `machine à écrire` ».

À noter que l'utilisation du *package* french rend inutile ce *modus operandi* : il est alors possible d'écrire plus simplement :

```
Par Jupiter !
```

pour obtenir le résultat précédent. En outre, si la commande `\untypedspaces` a été activée, l'entrée :

```
Par Jupiter!
```

produit elle aussi ce résultat.

L'espace légère insécable est également nécessaire après les guillemets ouvrants et avant les guillemets fermants :

> « Par Jupiter ! »

exemple que l'on peut produire comme suit en utilisant le *package* french :

```
<< Par Jupiter ! >>
```

Les parenthèses sont quant à elles « serrées » sur la portion de texte qu'elles délimitent :

> (une aventure d'Astérix le Gaulois)

Par contre une espace normale doit être laissée avant et après les tirets :

> Les aventures d'Astérix — écrites par René Goscinny et dessinées par Albert Uderzo — connaissent un franc succès.

En ce qui concerne la division en fin de ligne des mots français, il n'est point besoin d'en reproduire les règles ici, dans la mesure où elles sont presque toujours correctement appliquées lorsqu'on utilise le *package* french[64]. Mentionnons toutefois qu'en typographie « pure », on évite autant que possible de couper devant une syllabe muette, c'est-à-dire — par exemple — de couper :

> sylla-be

mais *Le bon usage* [Grévisse, 1988] est le premier à noter qu'il est difficile de suivre scrupuleusement cette règle[65].

Outre les usages des espaces insécables (« ~ » en LaTeX) signalés plus haut, la coupure des lignes est indésirable :

- entre une mesure et son unité :

  ```
  300~000~km/s
  ```

  (remarquez aussi comment séparer la colonne des mille de celle des centaines) ;
- entre l'avant-dernier prénom et le dernier, entre les éléments d'un nom de famille, et entre une particule et le mot qui la suit ; [Knuth, 1984, Chapter 14] illustre tous ces usages par l'exemple suivant :

  ```
  Charles Louis Xavier~Joseph de~la Vall\'{e}e~Poussin
  ```
- entre un prénom et un ordre :

  ```
  Louis~XVII
  ```

64. Les lecteurs intéressés par le *modus operandi* de la division des mots en TeX pourront trouver dans [Désarménien, 1986] une description des principes suivis pour établir les fichiers de coupures de TeX. La représentation de ces fichiers de coupure est expliquée dans [Knuth, 1984, Appendix H].

65. Une autre remarque générale — qui s'applique également à la langue anglaise — est qu'en typographie soignée, on évite autant que possible de terminer plus de deux lignes consécutives par un mot coupé. Ce point est une des différences entre les commandes `\fussy` (activée par défaut) et `\sloppy`, après l'activation de laquelle cette règle n'est pas suivie.

7.1.2 Ponctuation et emploi des polices

La « sémantique » des divers signes de ponctuation étant comprise, les seules règles devant lesquelles on peut hésiter[66] sont l'emploi du trait d'union pour relier plusieurs mots et le « télescopage » des divers signes de ponctuation en fin de citation ou de morceaux de phrase placés entre parenthèses.

En principe, le trait d'union sépare les éléments d'un mot composé. Mentionnons en outre que l'adverbe « non » est suivi d'un trait d'union lorsqu'il précède un nom, mais *pas* lorsqu'il précède un adjectif :

> un non-sens
> une relation non réflexive

En particulier, le trait d'union sépare les éléments d'un prénom composé :

> Jean-Jacques Rousseau

et est également utilisé entre les diverses composantes d'un nom de rue[67] :

> rue Henri-Poincaré
> rue Charles-III
> rue des Quatre-Églises

Les caractères *italiques* sont réservés aux mots que l'on désire mettre en valeur :

> Un *tapis-franc*, en argot de vol et de meurtre, signifie un estaminet ou un cabaret du plus bas étage[68].

aux titres d'œuvres :

> *Les nuits fauves*

et aux mots étrangers, cet usage incluant *ipso facto* les citations latines. Les guillemets sont en principe utilisés pour les mots qui ont un sens très imagé à l'intérieur de la phrase. Comme au début de ce § 7.1.2 :

> La « sémantique » des divers signes de ponctuation. . .

Dans ce cas, la frontière entre l'utilisation des caractères italiques et des guillemets reste néanmoins assez floue.

Selon l'usage le plus courant, les répliques s'indiquent à l'aide de tirets :

> — Vous êtes un fat personnage, Monsieur.
> — Ventrebleu, vous en êtes un autre, Monsieur.

quoique des guillemets puissent être utilisés :

> « Personne ne m'a jusqu'à ce jour adressé ainsi la parole, Monsieur. »
> « J'ai grand'peur que cela ne fasse que commencer, Monsieur. »

ou réservés aux seuls début et fin de la transcription d'une conversation :

> «Mes témoins attendront les vôtres, Monsieur.
> — Soit, puisque vous semblez las de vivre, Monsieur.»

Notons au passage que l'utilisation — avec le *package* `french` — des assemblages `<<` et `>>` pour les guillemets *exige* que lesdits guillemets soient appariés à l'intérieur du *même* paragraphe. Si ce n'est pas le cas, il convient d'utiliser les commandes `\leftguillemets` et `\rightguillemets`. Ainsi le passage précédent a été tapé :

```
\leftguillemets Mes t\'{e}moins...

... Monsieur.\rightguillemets
```

66. Surtout dans la mesure où elles sont différentes des règles anglaises (cf. § 7.2.2).
67. [Grévisse, 1988] précise que cet usage n'est pas suivi en Belgique.
68. Cette phrase est la première des *Mystères de Paris*, par Eugène Sue.

Lorsque des délimiteurs (parenthèses ou guillemets) ouvrants et fermants sont utilisés, le signe de ponctuation terminant le passage inclus est placé *avant* le délimiteur fermant si c'est *toute la phrase* qui est incluse à l'intérieur des délimiteurs (voyez les deux derniers échanges d'invectives plus haut). Ce cas inclut le fait que la citation peut avoir commencé à l'intérieur d'une phrase :

> Voici un exemple de syllogisme : « Ce qui est rare est cher. Un cheval bon marché est rare. Donc, un cheval bon marché est cher. »

Lorsque la citation est toute entière incluse dans une phrase, le signe de ponctuation qui termine la citation :

– est conservé si c'est un point d'interrogation ou d'exclamation :

> « Vertufichtre ! » s'exclama-t-il.

— et dans ce cas aucun signe de ponctuation ne suit les doubles guillemets fermants[69] — ;

– supprimé, dans les autres cas, parfois au détriment d'un point ou d'une virgule qui suit les doubles guillemets fermants :

> « Je bois au succès du guide local de LaTeX », dit-il et il joignit le geste à la parole.
> Je me souviens avoir dit : « Je bois au succès du guide local de LaTeX » et j'ai immédiatement joint le geste à la parole.

À noter que dans l'exemple suivant :

> Vous ne pourrez plus dire : « Je m'y perds dans tous ces TeX, LaTeX, LaTeX 2_ε, etc. »

le point fait partie intégrante de l'abréviation « etc. » (cf. table 7.1).

À propos des tirets, mentionnons que l'on peut omettre le « tiret fermant » en fin de phrase — du reste, il n'est pas indispensable. Dans l'écriture française, le tiret est également employé :

– comme début des différentes rubriques d'une énumération ;

– chacune de ces rubriques devant en principe se terminer par un point-virgule ;

– sauf bien entendu la dernière qui est close par un point.

Signalons que le tiret « classique » est produit en LaTeX par « `---` », mais que certains typographes actuels lui préfèrent le tiret court, produit par « `--` ».

Enfin, l'usage français commence tous les paragraphes par un retrait à droite, *y compris* celui qui est placé immédiatement après un titre de chapitre ou de subdivision de chapitre, mais ce n'est pas une règle absolue : cela dépend en fait des habitudes de la maison d'édition. Cette « indentation supplémentaire » du premier paragraphe peut être obtenue avec le *package* indentfirst.

7.1.3 Emploi des lettres capitales

Portent une capitale :

– les noms propres de personnes et de lieux ;

– certains titres :

> le Président de la République Française

– les *noms* désignant une nationalité ou les habitants d'un pays, mais *pas* les adjectifs[70] :

> Les Français sont attachés aux usages français.

69. C'est la règle la plus courante, mais elle n'est pas absolue en cas de conflit entre points d'interrogation et d'exclamation :

> — Pourquoi avez-vous crié : « Au voleur » ? Aurait-on volé les murs de votre bureau ?

l'important est de ne pas mettre deux signes de ponctuation, l'un avant les doubles guillemets fermants, l'autre après.

70. À moins que l'adjectif apparaisse dans une expression officielle, comme dans « la République Française » ou « l'État Français ».

Les *petites* capitales sont utilisées dans les énoncés de bibliographies :

Philippe DJIAN : *37° 2 le matin.* Barrault, éditeur.

et pour désigner les siècles, comme le XX^{e}. En revanche, les ordres des rois, empereurs ou papes sont en capitales :

Napoléon III

En ce qui concerne les titres d'œuvres, il n'y a en principe pas de règle introduisant des capitales supplémentaires. Cependant, on peut oberver que certaines maisons d'édition mettent une capitale au(x) mot(s) le(s) plus significatifs d'un titre. Il est toutefois déconseillé d'abuser de cette pratique [André et Grimault, 1990; André, 1993].

En principe, les accents, trémas et cédilles perdurent sur les capitales.

ÉVIDEMMENT, ÇA FAIT PLUS DE CARACTÈRES À TAPER SOUS L'ÉDITEUR.

A franchement parler, la seule exception concerne la préposition « à », car certains typographes considèrent le « À » inesthétique[71].

7.2 Typographie anglaise

7.2.1 Espacement et coupure de lignes

En anglais, les virgules, points, points-virgules, deux-points, points d'exclamation et d'interrogation ne sont *jamais* précédés d'une espace. Les parenthèses sont « serrées » sur la portion de texte qu'elles délimitent.

Quant aux tirets, aucune espace n'est laissée, ni à gauche, ni à droite :

Frank's reflection—distorted, fragmented—stated across the laquer.

La présence d'un tiret l'emporte sur un autre signe de ponctuation. Et de même qu'en français, le « tiret fermant » peut être omis à la fin d'une phrase :

Its voice, unlike that of its companion, was light and breathy—a voice of an excited girl.

En ce qui concerne les espaces insécables, il est d'usage d'en insérer devant un nombre que l'on interprète comme un numéro :

```
Lemma~10
```

et entre un *first name* et un *middle name* :

```
Donald~E. Knuth
Edgar~Rice Burroughs
H.~Rider Haggard
```

7.2.2 Ponctuation et emploi des polices

À la différence de l'usage français, le préfixe « *non* » est toujours suivi d'un trait d'union, ou — c'est ce que l'on remarque de plus en plus souvent — directement accolé au mot qu'il précède :

a non-generic procedure
a nonempty set

Les caractères italiques s'emploient pour mettre en valeur une phrase ou un morceau de phrase. Ils s'emploient aussi pour les titres d'œuvres. En ce qui concerne les citations, trois règles principales sont à retenir :

– les citations et répliques en style direct s'écrivent entre doubles guillemets ouvrants (‘‘) et fermants (’’) ;

71. Ce n'est toutefois pas une obligation. À preuve, cette note.

– si un signe de ponctuation termine la citation, il doit être placé *avant* les doubles guillemets fermants :

> "I asked you a question," it said.

– s'il y a conflit entre plusieurs signes, c'est le plus fort qui l'emporte : c'est-à-dire que les points d'interrogation et d'exclamation l'emportent sur les autres signes :

> "Do you understand?" the figure beside the first speaker demanded.

Lorsqu'une citation se termine par un point, il est d'usage de l'omettre avant le délimiteur fermant. En revanche, on n'omet pas le dernier point d'une abréviation devant un symbole de ponctuation autre que le point :

> John F. Kennedy, the president of the U.S.A., ...

Les locutions telles que *that is*, *namely*, *i.e.*, *e.g.* sont en principe suivies d'une virgule[72].

7.2.3 Emploi des lettres capitales

Portent une capitale, dans tous les cas :

– les noms propres de personnes et de lieux,

– les titres suivis du nom de la personne[73] :

> President John F. Kennedy
> the president of the U.S.A.

de même pour l'indication de la subvision d'un texte :

> Chapter 7
> the chapter about typographic rules

– les noms et adjectifs désignant une nationalité ou les habitants d'un pays,

– les notes de musique, ainsi que les modes (majeur et mineur) :

> Gustav Mahler's Fifth Symphony is in C sharp Minor.

Dans les titres d'œuvres, portent des capitales[74] :

– le premier et le dernier mot,

– tous les noms, pronoms (personnels, possessifs ou relatifs), adjectifs (qualificatifs ou possessifs), verbes, adverbes et conjonctions de subordination.

N'en portent pas :

– les articles, conjonctions de coordinations, prépositions, et le « *to* » des infinitifs ;

à moins, bien entendu, que ces mots soient à la première ou à la dernière place.

> *A River Runs through It*
> *The Man Who Mistook His Wife for a Hat*

En ce qui concerne les mots composés, le second élément porte une capitale si ce n'est pas un participe, ou si le premier élément est un préfixe courant :

> *Land of Always-Night*

72. La présence de cette virgule explique pourquoi l'usage de telles abréviations n'interfère pas avec la règle de l'espacement après le point en LaTeX, règle selon laquelle un point est suivi d'une espace plus large (car il est censé terminer une phrase), sauf si ce point suit une lettre capitale (car il est dans ce cas censé terminer l'abréviation d'un *middle name*). Cf. [Lamport, 1985, § 2.2.1] & [Lamport, 1994, § 2.2.1].

73. Auquel cas l'article défini ne doit pas être utilisé.

74. Les règles suivantes ne semblent pas être communément suivies en Grande-Bretagne.

cf.	latin *confer*, compare.
etc.	latin *et cetera*, et le reste.
ex.	exemple.
fig.	figure.
ibid.	*ibidem*, à la même place.
id.	*idem*, la même chose.
p.	page.
pp.	pages.
vol.	volume.

TAB. 7.1 - *Abréviations utilisées aussi bien en français qu'en anglais.*

7.3 Écriture des abréviations et principaux signes usuels

Nous proposons ci-après trois tables :

- la table 7.1 regroupe les abréviations dont l'emploi est commun au français et à l'anglais ;
- la table 7.2 regroupe les principales abréviations et signes particuliers spécifiques au français, ainsi que les commandes LaTeX produisant lesdits signes ;
- la table 7.3 propose un choix des principales abréviations anglaises.

À noter que :

- l'usage du signe « § » pour désigner un paragraphe semble beaucoup plus courant en français qu'en anglais, d'où sa mention dans la table 7.2 ;
- à part `\S`, toutes les commandes mentionnées dans la table 7.2 ne peuvent être utilisées qu'avec le *package* `french` ;
- si bon nombre des abréviations de la table 7.3 ont une origine latine, ces abréviations sont toutefois inusitées voire inconnues dans les documents français, c'est pourquoi nous les avons placées dans cette table[75].

Le *package* `french` permet en outre l'utilisation :

- de la commande `\quando`, dont l'argument doit être un entier naturel : ainsi, `\quando={10}` produit « 10^{o} » ;
- de l'environnement `order` : le texte source suivant :

```
\begin{order}
 \item Les aventuriers de l'arche perdue
 \item Indiana Jones et le temple maudit
\end{order}
```

  produit :

  1^{o} Les aventuriers de l'arche perdue
  2^{o} Indiana Jones et le temple maudit

En ce qui concerne les abréviations de prénoms, l'usage français conseille de ne pas couper les digrammes[76]. C'est-à-dire que — par exemple — :

Charles devrait s'abréger en « Ch. »,
Philippe .... « Ph. ».

Le trait d'union perdure dans une abréviation. Exemples :

John Fitzgerald Kennedy $\Longrightarrow$ J. F. Kennedy[77]
Jean-Jacques Rousseau $\Longrightarrow$ J.-J. Rousseau

75. En fait, par « abréviations anglaises », nous sous-entendons « abréviations utilisés dans des documents écrits en anglais ».
76. Il est dommage que BibTeX ne suive pas cette règle.
77. L'espace entre « J. » et « F. » étant insécable.

1^{er}, 1^{re}, 2^{e}, ...	adjectifs ordinaux : premier, première, ...	`1\ier`, `1\iere`, `2\ieme`, ...
1^{ers}, 2^{es}, ...	adjectifs ordinaux au pluriel.	`1\iers`, `2\iemes`, ...
1^{o}, 2^{o}	Les deux premiers...	`\primo`, `\secundo`
3^{o}, 4^{o}	... et les deux suivants.	`\tertio`, `\quatro`
ap.J.-C.	après Jésus-Christ.	
av.J.-C.	avant Jésus-Christ.	
c.-à-d.	c'est-à-dire.	
CEDEX	courrier d'entreprise à distribution exceptionnelle.	
C.E.E.	Communauté Économique Européenne.	
cf.	latin *confer*, compare.	
C.Q.F.D.	ce qu'il fallait démontrer.	
env.	environ.	
M.	Monsieur.	
M^{lle}	Mademoiselle.	`M\fup{lle}`
M^{lles}	Mesdemoiselles.	`M\fup{lles}`
MM.	Messieurs.	
M^{me}	Madame.	`M\fup{me}`
M^{mes}	Mesdames.	`M\fup{mes}`
M^{e}	Maître (avocat ou notaire).	`M\fup{e}`
n^{o}, N^{o}	numéro.	`\numero`, `\Numero`
n^{os}, N^{os}	numéros.	`\numeros`, `\Numeros`
p.ex.	par exemple.	
P.J.	pièce jointe [a].	
sqq.	latin *sequentes*, et suivant(e)s.	
S.V.P.	s'il vous plaît.	
T.S.V.P.	tournez, s'il vous plaît.	
§	paragraphe.	`\S`

[a] Désigne aussi « police judiciaire » pour les amateurs de polars.

TAB. 7.2 - *Abréviations et symboles propres au français.*

a.d.	latin *anno domini*, littéralement « année du Seigneur », après Jésus-Christ.
a.m.	latin *ante meridiem*, avant midi.
b.c.	*before Christ.*
ca.	latin *circa*, environ.
E.E.C.	*European Economic Community.*
e.g.	latin *exempli gratia*, par exemple.
esp.	*especially.*
et al.	latin *et alii*, et d'autres.
i.e.	latin *id est*, c'est-à-dire.
Mr.	*Mister.*
Mrs.	*Mistress.*
Ms.	(prononcé [miz]) : Madame ou Mademoiselle.
no.	numéro.
Ph.D.	latin *philosophiae doctor*, Docteur en Philosophie.
p.m.	latin *post meridiem*, après midi.
sec.	*section.*
ser.	*series.*
Q.E.D.	latin *quod erat demonstrandum*, ce qu'il fallait démontrer.
viz	d'origine latine (*videlicet*), équivalent de *namely* — et souvent lu ainsi.
w.r.t.	*with respect to.*
#	numéro

TAB. 7.3 - *Abréviations et symboles propres à l'anglais.*

```
\usepackage[english,french,catalan,slovak]{babel}

\selectlanguage{slovak}

Chcete ochutna\v{t} gul\'{a}\v{s}?

\selectlanguage{french}

= Voudriez-vous go\^{u}ter un peu de goulash ?
```

FIG. 7.4 - *Exemple d'utilisation du* package babel.

7.4 Être polyglotte...

7.4.1 ... avec le *package* french

La commande `\english` du *package* french permet de repasser dans la configuration initiale de LaTeX, comme si nous n'étions pas en mode français. En particulier, les règles d'espacement autour des signes de ponctuation tels que les deux-points ou le point-virgule (cf. § 7.1.1) n'ont plus cours et suivent les conventions « de base » de LaTeX. Le mode français peut être restauré par la commande `\french`. Voyez l'extrait suivant :

> *Every inch of its head had been tattoed with an intricate grid, and at every intersection of horizontal and vertical axes a jewelled pin driven through to the bone*[78].

ainsi que la note en bas de page qui s'y rattache. Ils ont été produits par :

```
\english
\emph{Every inch... through to the bone\footnotemark.}
\french
\footnotetext{Cette citation...}
```

7.4.2 ... avec le futur *package* babel

Le *package* babel [Goossens *et al.*, 1994, § 9.2] permet :

- l'adaptation d'un style de document à une langue particulière : ainsi les commandes `\chaptername`, `\contentsname`, ... y sont redéfinies pour chaque langue de ce *package* ;
- l'utilisation d'un fichier de *coupures possibles* qui soit *ad hoc* pour la langue choisie[79] ;
- l'écriture multilingue à l'intérieur d'un même document, comme vous le montre la figure 7.4 : la sélection d'une langue est effectuée par la commande `\selectlanguage`.

Actuellement, les *packages* babel et french ne sont pas compatibles : en attendant, il vaut mieux, pour l'écriture d'un texte français, utiliser le *package* french, qui devrait par la suite devenir une option de babel. On peut toutefois charger « à la main » les trois fichiers de coupures installés sur notre site :

- `\language=0` (par défaut) sélectionne le fichier des coupures anglaises ;
- `\language=1` (utilisée par le *package* french) sélectionne les coupures françaises ;
- `\language=2` sélectionne les coupures allemandes.

---

78. Cette citation — ainsi que quatre exemples du § 7.2 — est extraite de *The Hellbound Heart*, par Clive Barker (éd. par Fontana). *Of course*, elle apparaît en caractères italiques à l'intérieur d'un texte français.
79. À condition toutefois que ce fichier soit connu lors de l'installation du *package* babel.

7.5 Compléments

Il est clair que l'énoncé de règles grammaticales n'a pas sa place dans ce petit mémento. De bons ouvrages traitant des règles françaises sont :

[Grévisse, 1988] qui inclut en outre les principales règles typographiques françaises ;

[Mauger, 1968] plus spécifiquement destiné à des étrangers en phase d'apprentissage de la langue de Victor Hugo.

D'un point de vue « écriture de documents en LaTeX », on ajoutera à cette liste le chapitre 8 de [Rolland, 1993] qui est une présentation du *package* french plus fouillée que celle que nous en avons donnée (à noter toutefois que ce livre a été écrit à une époque où LaTeX 2_ε n'était pas encore diffusé), sans oublier la documentation de french elle-même[80].

En ce qui concerne la langue anglaise :

[Swan, 1980] est complet et facilement consultable ;

[Strunk Jr. et White, 1979] est peut-être plus orienté vers les questions de style que vers la justesse grammaticale ;

[Thomson et Martinet, 1980] est une grammaire très complète — à noter que l'achat de sa traduction en français [Thomson et Martinet, 1984] est un bon investissement : tous les exemples de l'original y sont traduits, avec des indications sur les diverses nuances.

Pour finir par le côté typographique « pur et dur », la référence incontournable pour la langue française est bien sûr [CT, 1989]. Sur des points techniques plus précis, de bons conseils pour l'emploi des capitales et la composition des titres peuvent être trouvés respectivement dans [André et Grimault, 1990] et [André, 1993]. Nous recommandons également [Chicago, 1982], qui n'est d'ailleurs pas limité à la langue anglaise et donne en sus de précieuses indications d'ordre général : comment annoter une épreuve, comment représenter un *copyright*, ...

80. dans `/local/tex/archive/language/french`.

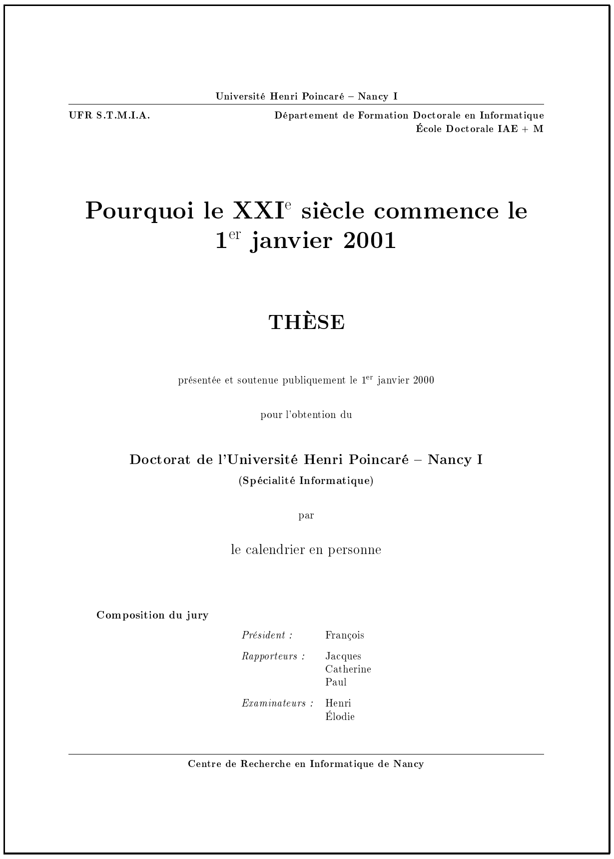
Chapitre 8

La classe **TheseCRIN**

Ce chapitre décrit la version v0.63b de la classe TheseCRIN et est disponible sous forme séparée dans `/local/tex/loria/TheseCRIN`.

La classe TheseCRIN *propose* un standard pour la présentation des thèses du CRIN et ne fonctionne qu'avec LaTeX 2_ε. Elle se caractérise par une certaine souplesse, en particulier pour la page de titre et la composition de la table des matières.

8.1 Exemple introductif



Université Henri Poincaré – Nancy I

UFR S.T.M.I.A.

Département de Formation Doctorale en Informatique
École Doctorale IAE + M

Pourquoi le XXI^e siècle commence le 1^er janvier 2001

THÈSE

présentée et soutenue publiquement le 1^er janvier 2000

pour l'obtention du

Doctorat de l'Université Henri Poincaré – Nancy I
(Spécialité Informatique)

par

le calendrier en personne

Composition du jury

Président :	François
Rapporteurs :	Jacques Catherine Paul
Examinateurs :	Henri Élodie

Centre de Recherche en Informatique de Nancy

La figure de la page précédente illustre le début d'une thèse composée avec TheseCRIN. Elle correspond aux commandes suivantes :

```
\documentclass[version1]{TheseCRIN} % version1 pour indiquer la version
                                    % que l'on utilise.
\usepackage{amssymb} % pour utiliser les symboles de l'AMS
\begin{document}

\ThesisTitle{Pourquoi le XXI$^{\mathrm{e}}$ si\`ecle commence
             le 1$^{\mathrm{er}}$ janvier 2001}
\ThesisDate{1$^{\mathrm{er}}$ janvier 2000} % 1\ier  avec french.sty
\ThesisAuthor{le calendrier en personne}
\ThesisNancyI
\President    = {Fran\c{c}ois}
\Rapporteurs  = {Jacques\\
                 Catherine\\
                 Paul}
\Examinateurs = {Henri\\
                 \'Elodie}
\MakeThesisTitlePage

\begin{ThesisAcknowledgments}
Les remerciements.
\end{ThesisAcknowledgments}

\begin{ThesisDedication}
Je d\'edie cette th\`ese\\
aux saints du calendrier.\\
\end{ThesisDedication}

\tableofcontents

\DontWriteThisInToc
\listoffigures

\mainmatter
\chapter*{Introduction}
...
\begin{ThesisAbstract}
  \begin{FrenchAbstract}
    Le r\'esum\'e.
    \KeyWords{chat, chien, puces.}
  \end{FrenchAbstract}
  \begin{EnglishAbstract}
    The abstract.
    \KeyWords{cat, dog, flees.}
  \end{EnglishAbstract}
\end{ThesisAbstract}
\end{document}
```

8.2 Fonctionnalités offertes par cette classe

La classe TheseCRIN hérite de toutes les commandes disponibles dans la classe `book`, ainsi que dans le *package* `french`, *sans cependant reposer sur ces dernières.* On lira les documentations correspondantes, par exemple :

- Leslie Lamport : **LaTeX – A document preparation system**, Addison-Wesley, seconde édition, 1994.
- Bernard Gaulle : **Notice d'utilisation du style french multilingue V 3.35** – 19 décembre 1994 (disponible dans `/local/tex/archive/language/french`).

Un squelette de document TheseCRIN peut-être créé au moyen du programme interactif `skeltex` ou bien à partir du fichier d'exemple distribué avec la classe dans `/local/tex/loria/TheseCRIN`.
Voyons les caractéristiques de TheseCRIN.

8.2.1 Options de la classe

Le document doit commencer par

`\documentclass[version1` [`,nofrench`] [`,minfrench`] [`,mltex`] [`,`⟨*autres options*⟩] `]{TheseCRIN}`

- '`version1`' est une option obligatoire qui permettra de gérer d'éventuelles incompatibilités futures (par exemple, le retrait de certaines commandes), de telle sorte qu'une thèse compilée en 1995 avec TheseCRIN.cls devrait encore compiler en l'an 2000, bien que TheseCRIN.cls puisse être différent (dans le futur, il pourra y avoir des options `version2`, `version3`, etc.).
  S'il y a plusieurs versions, avant de passer à une version plus récente, c'est-à-dire, avant de remplacer '`version1`' par '`version2`' ou plus, l'utilisateur devrait trouver ce qui a été modifié.
- '`nofrench`' est normalement une *option de test* à n'utiliser que lorsqu'on suspecte le *package* `french` d'être à l'origine d'une erreur. Cette option supprime les coupures de mots françaises et les commandes propres à `french`, comme `<< ... >>`, ne sont plus reconnues.
  Dans certains cas, cependant, cette option permet de contourner des problèmes liés à l'ordre de chargement des *packages*. Comme `french` est préchargé par TheseCRIN, les *packages* demandés par l'utilisateur sont en général chargés ultérieurement. Si l'on souhaite par exemple charger le *package* `oz` avant le *package* `french`, on écrira :

```
\documentclass[...,nofrench,...]{TheseCRIN}
\usepackage{oz}
\usepackage{french}
     ...
```

- '`minfrench`' est une option qui a pour effet de charger le minimum nécessaire pour la typographie française. Pour le moment, ce minimum est réduit aux règles de coupures des mots et à la traduction de « Table of contents », « chapter », etc. Si cette option est choisie, le *package* `french` n'est pas chargé.
- '`mltex`' est une option qui *peut* être utilisée si le moteur est MlTeX et non TeX. Cette option évite le chargement du *package* `t1enc`. La présence de cette option déclenche un avertissement si le moteur est TeX, et le *package* `t1enc` est alors tout de même chargé. Il vous est déconseillé d'utiliser cette option car elle peut conduire à des problèmes de portabilité. Il vous est même déconseillé d'utiliser MlTeX (cf. chapitre 1).
- ⟨*autres options*⟩ est une liste d'options reconnues par la classe `book`. L'option `a4paper` est sélectionnée par défaut.

8.2.2 Généralités sur la mise en page

Le format d'impression prévu est le format A4. D'autres formats de page n'ont pas (encore) été implantés. Par défaut, `\textheight` vaut approximativement 23 cm (ceci dépend légèrement de la police utilisée, car c'est un multiple entier de `\baselineskip`) et `\textwidth` vaut 160 mm.

8.2.2.1 Marges

L'impression recto-verso est le mode par défaut et les marges sont calculées pour obtenir une superposition par transparence.

Les marges gauches et droites devraient être changées, si nécessaire, avec `\SetRealMargins`. Les deux arguments de cette commande sont les valeurs des marges gauche et droite *réelles* sur les pages impaires. Sur les pages paires, les marges seront symétriques. `\SetRealMargins{26mm}{16mm}` définit par exemple une marge gauche de 26 mm et une marge droite de 16 mm, ce qui – avec du papier A4 – donne à `\textwidth` la valeur 21 cm − 26 mm − 16 mm = 168 mm. `\SetRealMargins{30mm}{20mm}` fixe les valeurs par défaut. L'utilisateur veillera à ne pas modifier `\textwidth` directement.

Les différents paramètres verticaux sont laissés à la discrétion de l'utilisateur. Toutefois, lorsqu'un paramètre de l'en-tête est modifié (par exemple `\headsep`), on peut vouloir ajuster d'autant la hauteur de la page. Ceci est réalisé avec la commande `\AdjustTextheight`, immédiatement après la modification d'un paramètre de mise en page verticale. Par exemple :

```
\setlength{\headsep}{30pt}
\AdjustTextheight
```

8.2.2.2 Décalages horizontaux et verticaux des imprimantes

Les quatre commandes `\Shift{Odd|Even}Pages{Right|Down}` peuvent être utilisés pour corriger des défauts de superposition dans les imprimantes recto-verso. Par exemple, si les pages impaires doivent être décalées à droite de 1 mm et vers le bas de 2 mm afin de correspondre aux pages paires, on pourra écrire :

```
\ShiftOddPagesRight{1mm}
\ShiftOddPagesDown{2mm}
```

Remarques :

- Les valeurs par défaut sont toutes nulles, ce qui revient à considérer l'imprimante parfaite.
- Les décalages peuvent être négatifs.
- `\ShiftOddPagesRight` et `\ShiftEvenPagesRight` doivent être réappliqués après chaque utilisation de `\SetRealMargins`.
- Pour le bon fonctionnement de `\ShiftOddPagesDown` et `\ShiftEvenPagesDown`, les pages numérotées impaires doivent toujours être placées sur le même côté.

8.2.2.3 Pages blanches

L'impression étant supposée *recto-verso*, des pages blanches sont ajoutées en certains endroits afin de conserver les pages impaires sur le côté *recto*.

8.2.2.4 Marque « brouillon »

`\ThesisDraft` écrit – sous forme encadrée – « Brouillon », la date et le nom du fichier source dans les « en-têtes ». Les en-têtes normaux restent présents.

Remarques :

- Les éventuels `\pagestyle{...}` suivant la commande `\ThesisDraft` sont ignorés.
- `\ThesisDraft` utilise la commande `\WeekDay` qui donne le jour de la semaine pour une date donnée du calendrier grégorien. Par exemple, `\WeekDay{1}{1}{2101}` donne 'samedi'. Cette commande peut être utilisée indépendamment.

8.2.2.5 En-têtes et pieds de page

• Styles d'en-têtes

`\FormatHeadingsWith` permet de formatter les « en-têtes » avec une commande particulière. Ceci ne s'applique pas aux numéros de page que l'on pourra modifier en changeant `\thepage`. L'exemple

```
\newcommand\bfheadings[1]{\textbf{#1}}
\FormatHeadingsWith{\bfheadings}
```

a pour effet de mettre dans les en-têtes les titres courants (chapitres, sections) en gras. L'argument de `\FormatHeadingsWith` doit être un unique nom de commande. On peut réutiliser des commandes déjà existantes :

```
\FormatHeadingsWith{\uppercase}
```

ou combiner plusieurs formats :

```
\newcommand\upun[1]{\uppercase{\underline{#1}}}
\FormatHeadingsWith\upun
```

Les commandes suivantes sont prédéfinies :

- `\UppercaseHeadings` : les hauts de page sont mis en majuscules. Cette commande est équivalente à `\FormatHeadingsWith{\uppercase}`.
- `\NormalHeadings` : les hauts de page sont inchangés (c'est l'option par défaut)
- `\UnderlineHeadings` : les hauts de page sont inchangés et soulignés. Cette commande est équivalente à `\FormatHeadingsWith{\underline}`.

• Styles de pages

- Par défaut, les en-têtes suivent le style « `ThesisHeadings` » : les numéros de page sont placés au bas et au milieu de chaque page et les autres « en-têtes » sont placés comme avec `\pagestyle{headings}`.
- `\pagestyle{headings}`, au contraire, place les numéros de page à droite sur les pages impaires et à gauche sur les pages paires.
- `\pagestyle{plain}` place simplement les numéros de page en bas et au milieu. Aucune information n'est placée en haut de la page. Ce style peut être intéressant, localement, par exemple pour la table des matières. On procédera ainsi :

```
...
\newpage % pour que le style 'plain' n'agisse pas plus tot
\pagestyle{plain}
\tableofcontents
\newpage % remarque analogue pour 'ThesisHeadings'
\pagestyle{ThesisHeadings}
...
```

- `\pagestyle{ThesisHeadings}` permet de revenir aux « en-têtes » par défaut, après un passage au style `headings` par exemple.

• Lignes de rappel

Il est possible d'obtenir une ligne soulignant tout l'en-tête. On pourra agir sur `\HeadRuleWidth` et `\HeadRuleSep`. La première variable indique l'épaisseur de la ligne et la seconde est la distance entre l'en-tête et la ligne. Les valeurs par défaut sont 0 pt et 2 mm. Ces valeurs peuvent être changées avec `\setlength` (l'épaisseur usuelle d'une ligne étant de 0,4 pt). Exemple :

```
\setlength{\HeadRuleWidth}{0.4pt}
```

Cette ligne ne sera visible que si un nom de chapitre ou de section apparaît au-dessus d'elle. L'utilisateur veillera à modifier `\headsep` si nécessaire.

8.2.2.6 Numérotation des pages

Par défaut, les pages sont numérotées dès le début avec des chiffres romains. En général, l'utilisateur changera cette numérotation après l'introduction de sa thèse, avec `\mainmatter` qui est une nouvelle commande de LaTeX 2ε. (`\frontmatter` et `\backmatter` ne peuvent pas être utilisés dans TheseCRIN) Cette commande est quasiment équivalente à `\pagenumbering{arabic}`, mais si la dernière page numérotée en chiffres romains est impaire, une page blanche supplémentaire est ajoutée. Si le document est structuré en plusieurs fichiers, il faut veiller à ce que `\mainmatter` figure dans le fichier principal.

La position des numéros de page peut être modifiée en changeant le style de la page (`\pagestyle`). On peut ne modifier que le numéro de page en redéfinissant `\thepage`.

8.2.3 Page de titre

La génération la plus simple de la page de titre n'implique que quelques commandes. En voici un exemple :

```
\ThesisTitle{Pourquoi les oiseaux ne peuvent pas
             voler aussi haut que les jumbo-jets}
\ThesisDate{11 ao\^ut 1999}
\ThesisAuthor{La mouche}
\ThesisNancyI
% Jury:
\President    = {Le pr\'esident}
\Rapporteurs  = {Le rapporteur 1\\
                 Le rapporteur 2\\
                 Le rapporteur 3}
\Examinateurs = {L'examinateur 1\\
                 L'examinateur 2}
\MakeThesisTitlePage
```

8.2.3.1 Commandes générales

Ces commandes n'appellent, pour la plupart, pas de commentaires :

- Titre : `\ThesisTitle`
- Date : `\ThesisDate`
- Auteur : `\ThesisAuthor`
- Université de la thèse : `\ThesisNancyI`, `\ThesisNancyII` ou `\ThesisINPL`
- Projet : `\InriaProject` (ceci ajoute la mention « INRIA–Lorraine » sur la page de titre)

8.2.3.2 Catégories dans le jury

Le jury est divisé en catégories. Celles prédéfinies dans TheseCRIN sont `\President`, `\Rapporteurs`, `\Examinateurs` et `\Invites`. Par défaut, les catégories sont vides.

8.2.3.3 Nouvelles catégories

De nouvelles catégories peuvent être ajoutées par le truchement de la commande `\NewJuryCategory`. Par exemple :

```
\NewJuryCategory{family}{\textit{Membre de la famille :}}%
                        {\textit{Membres de la famille :}}
```

Remarques :

- Le premier argument de `\NewJuryCategory` peut être utilisé pour définir les éléments de la nouvelle catégorie. Après cette définition, on pourrait écrire : `\family={Mon fr\`ere\\Ma s\oe ur}`

- Les second et troisième arguments de `\NewJuryCategory` sont respectivement les formes singulière et plurielle du label de la catégorie.
- Une nouvelle catégorie est ajoutée à la fin de la liste des catégories existantes.
- L'ordre des catégories peut être changé en effaçant le jury (avec `\ClearJury`) et en donnant un nouvel ensemble de catégories avec `\NewJuryCategory`. L'exemple suivant montre comment échanger l'ordre d'apparition des rapporteurs et des examinateurs :

```
\ClearJury
\NewJuryCategory{President}{\textit{Pr\'esident :}}{\textit{Pr\'esidents :}}
\NewJuryCategory{Examinateurs}{\textit{Examinateur :}}{\textit{Examinateurs :}}
\NewJuryCategory{Rapporteurs}{\textit{Rapporteur :}}{\textit{Rapporteurs :}}
\NewJuryCategory{Invites}{\textit{Invit\'e :}}{\textit{Invit\'es :}}
```

- Le label faisant face à une catégorie est au singulier ou au pluriel suivant qu'il y a une ou plusieurs personnes dans cette catégorie.
- On notera qu'il est en particulier possible de créer une (voire plusieurs) catégories sans label. Par exemple : `\NewJuryCategory{nothing}{}{}`.

8.2.3.4 La page de titre

La page de titre elle-même est construite avec `\MakeThesisTitlePage`. Si certains champs essentiels n'ont pas été remplis, des valeurs par défaut sont choisies et dans certains cas, il apparaît une case indiquant la commande à utiliser pour la remplir.

La page de titre est indépendante des options de taille (`11pt`, `12pt`) ou de police (par exemple, utilisation du *package* `times`). D'autre part, la commande `\MakeThesisTitlePage` peut apparaître plusieurs fois dans le document, éventuellement en changeant certains paramètres.

8.2.3.5 La reliure

Normalement, la page de titre est centrée sur la page physique. Toutefois, s'il y a une reliure, on peut vouloir décaler la page de titre un peu vers la droite. Ceci est réalisé avec `\SetBounding{`⟨*décalage*⟩`}`. Par exemple : `\SetBounding{5mm}` spécifie une reliure dont la largeur est de 5mm. La page de titre est alors centrée sur l'espace restant. Cette commande doit précéder `\MakeThesisTitlePage`. La valeur par défaut de la largeur de reliure est 0 mm.

8.2.4 Remerciements

L'environnement `ThesisAcknowledgments` permet de placer des remerciements sur une page qui leur est propre. Un exemple simple est :

```
\begin{ThesisAcknowledgments}
Je vous remercie tous.
\end{ThesisAcknowledgments}
```

Remarques :

- La position du titre peut être changée avec `\AlignTitlesLeft`, etc. (cf. § 8.2.9)
- Le titre peut être modifié en redéfinissant `\AcknowledgmentName`. Exemple :

  ```
  \renewcommand{\AcknowledgmentName}{Ceux que je remercie:}
  ```

  Cette commande est donc similaire à `\partname`, `\chaptername`, etc.
- Par défaut, il n'y a pas d'entrée pour les remerciements dans la table des matières, mais la commande `\WriteThisInToc` placée avant l'environnement de remerciements force l'entrée. L'entrée peut aussi être encadrée (si elle ne l'est pas par défaut par `\FrameChaptersInToc`) en plaçant `\FrameThisInToc` avant l'environnement.

8.2.5 Résumés

L'environnement `ThesisAbstract` crée une page pour le(s) résumé(s). L'utilisateur peut y employer les sous-environnements `FrenchAbstract` et `EnglishAbstract` et donner des mots clés par l'intermédiaire de la commande `\KeyWords`. Par exemple :

```
\begin{ThesisAbstract}
   \begin{FrenchAbstract}
      Le r\'esum\'e.
      \KeyWords{chat, chien, puces.}
   \end{FrenchAbstract}
   \begin{EnglishAbstract}
      The abstract.
      \KeyWords{cat, dog, flees.}
   \end{EnglishAbstract}
\end{ThesisAbstract}
```

Remarques :

1. `\KeyWords` ne peut être utilisé que dans les environnements `FrenchAbstract` ou `EnglishAbstract`.
2. Au sein de l'environnement `EnglishAbstract`, les règles typographiques anglo-saxonnes sont observées, et en particulier les motifs de division de mots anglo-saxons sont utilisés.
3. La position des titres peut être changée avec `\AlignTitlesLeft`, etc. (cf. § 8.2.9)
4. L'environnement crée deux pages sans numéro de page, de telle sorte que la page de résumé puisse figurer sur la quatrième page de couverture de la thèse.
5. La commande `\NumberAbstractPages` placée devant l'environnement `ThesisAbstract` entraîne la numérotation des pages du résumé. La commande `\DontNumberAbstractPages` annule la précédente.
6. Par défaut, il n'y a pas d'entrée dans la table des matières, mais `\WriteThisInToc` placé avant l'environnement de remerciements force cette entrée.
7. L'environnement compte dans la numérotation des pages, même si les numéros n'apparaissent pas sur la page. Ceci permet d'avoir un comportement correct avec `\WriteThisInToc`. On placera généralement le(s) résumé(s) à la fin de la thèse.
8. Cet environnement peut être réutilisé à plusieurs reprises, par exemple au début et à la fin de la thèse. Le plus simple est d'écrire quelque chose comme :

   ```
   \newcommand{\MonResumeFrancais}{Le r\'esum\'e.
                   \KeyWords{chat, chien, puces.}}
   \newcommand{\MonResumeAnglais}{The abstract.
                  \KeyWords{cat, dog, flees.}}
   ...
   \begin{ThesisAbstract}
      \begin{FrenchAbstract}
         \MonResumeFrancais
      \end{FrenchAbstract}
      \begin{EnglishAbstract}
         \MonResumeAnglais
      \end{EnglishAbstract}
   \end{ThesisAbstract}
   ...
   ```

   En procédant ainsi, on peut éventuellement mettre des résumés différents en plusieurs endroits. On peut aussi placer facilement les résumés français et anglais sur des pages différentes en n'utilisant à chaque fois que l'un des sous-environnements `FrenchAbstract` ou `EnglishAbstract`.
9. Les entrées de résumé, lorsqu'elles sont présentes dans la table des matières, ne sont jamais encadrées (elles n'obéissent pas à `\FrameChaptersInToc` ou `\FrameThisInToc`).

8.2.6 Dédicace

L'environnement `ThesisDedication` place la dédicace sur une page isolée, alignée sur le bord droit, à environ un tiers du haut de la page. Par défaut, c'est la famille de fontes cmss (Computer Modern Sans Serif), en *oblique* qui est utilisée. Exemple :

```
\begin{ThesisDedication}
Je d\'edie ce travail\\
\`a ma machine.
\end{ThesisDedication}
```

8.2.7 Table des matières

8.2.7.1 Généralités

- Il n'y a jamais d'entrée « table des matières » dans la table des matières.
- Le rapport entre l'interligne de la table des matières et l'interligne normal est de 1,3. Cette valeur peut être changée avec `\SetTocSpacing`, avant `\tableofcontents`.
  Par exemple : `\SetTocSpacing{2}`.
- Une ligne peut être ajouté entre deux entrées de la table des matières en écrivant `\PutLineInToc` à l'*endroit du texte* où doit se situer la ligne.
- La position du titre « Table des matières » peut être changée avec `\AlignTitlesLeft`, etc. (cf. § 8.2.9).
- La pagination des entrées des chapitres et des parties peut être modifiée localement avec les commandes `\NumberThisInToc` et `\DontNumberThisInToc` qui doivent précéder la commande de sectionnement.

8.2.7.2 Entrées des parties

- Une entrée est écrite dans la table des matières, à la fois avec `\part`, et avec `\part*`.
- `\WriteThisInToc` et `\DontWriteThisInToc` n'agissent pas sur les entrées `\part` ou `\part*`.
- `\WritePartLabelInToc` : ceci ajoute 'Partie' devant le numéro de partie dans la table des matières.
- `\FramePartsInToc` : ceci encadre l'entrée des parties (`\part` et `\part*`) dans la table des matières, met le numéro de partie sur une ligne propre et le nom de la partie sur une autre ligne. Cette commande fonctionne le mieux en conjonction avec `\WritePartLabelInToc`. Cette commande (tout comme la suivante) doit être placée *après* `\begin{document}`.
  `\DontFramePartsInToc` annule l'effet de `\FramePartsInToc` pour les parties subséquentes (ou jusqu'au prochain `\FramePartsInToc`).
- Lorsqu'une entrée n'est normalement pas encadrée, on peut demander un encadrement uniquement pour cette entrée en plaçant `\FrameThisInToc` devant `\part...`. Cette commande ne prend pas d'argument. Exemple :

```
\FrameThisInToc
\part{Une entr\'ee exceptionnellement encadr\'ee}
```

- Similairement, `\DontFrameThisInToc` peut être utilisé pour l'effet opposé.

8.2.7.3 Entrées des chapitres

- Une entrée est écrite dans la table des matières, à la fois avec `\chapter`, et avec `\chapter*`. Dans ce dernier cas, cela peut être évité en utilisant `\DontWriteStarChaptersInToc`. La bibliographie, l'index et le glossaire sont créés par des commandes `\chapter*`.
- Les entrées `\chapter*` peuvent être modifiées localement en utilisant soit `\WriteThisInToc`, soit `\DontWriteThisInToc` avant le chapitre.
- Les entrées de `\chapter` ne peuvent pas être retirées de la table des matières.
- `\WriteChapterLabelInToc` : ceci ajoute 'Chapitre' devant le numéro de chapitre dans la table des matières.

– `\FrameChaptersInToc` : ceci encadre l'entrée des chapitres (`\chapter` ou `\chapter*`) dans la table des matières, met le numéro de chapitre sur une ligne propre et le nom du chapitre sur une autre ligne. Ceci fonctionne le mieux en conjonction avec `\WriteChapterLabelInToc`. Cette commande (tout comme la suivante) doit être placée *après* `\begin{document}`.
  Attention : ceci s'applique aussi aux entrées de remerciements, bien que ce soit une section.
  `\DontFrameChaptersInToc` annule l'effet de `\FrameChaptersInToc` pour les chapitres subséquents. (ou jusqu'au prochain `\FrameChaptersInToc`)
  La commande `\DontFrameThisInToc` permet de supprimer l'encadré localement et ce pour l'unique `\chapter` ou `\chapter*` qui suit la commande. Le fonctionnement est en tous points analogue à celui des parties.
– `\NoChapterPrefix` supprime le numéro du chapitre devant le numéro de section, etc. (aussi dans les chapitres eux-mêmes). Les labels et les références ne sont pas affectés, sauf si `\NoChapterNumberInRef` est utilisé conjointement.

8.2.8 Têtes de parties

La commande `\NoNewPageAfterParts` supprime le saut de page immédiatement après les titres des parties. Ceci permet alors d'insérer une introduction immédiatement après le titre.

8.2.9 Têtes de chapitres

– `\NoChapterHead` :
  Avec cette commande, le mot « Chapitre » n'apparaît pas au début d'un chapitre (c'est-à-dire « 3 » au lieu de « Chapitre 3 »).
– `\AlignTitlesLeft`, `\CenterTitles`, `\AlignTitlesRight` :
  Le numéro et le nom des chapitres est justifié en rapport avec cette commande. Par défaut, on centre. Ceci s'applique aux titres des environnements de résumé, de remerciements, d'index, de glossaire, de bibliographie, aux listes des tables et des figures et à la table des matières.
– S'il y a des parties, les chapitres sont normalement numérotés consécutivement mais peuvent être remis à 1 systématiquement en plaçant `\ResetChaptersAtParts`, immédiatement après `\begin{document}`.

8.2.10 Index

L'environnement d'index se présente comme dans la classe `book`, mais crée une entrée dans la table des matières. Cet environnement peut être utilisé soit directement, comme :

```
\begin{theindex}
...
\item divide and conquer, 15
\item double
  \subitem constants, \see{constants}{111}
  \subitem data type, 11
...
\end{theindex}
```

soit par l'intermédiaire de `\PrintIndex`, après application de `makeindex` (la commande `\printindex` ne doit pas être utilisé car étant redéfinie par le *package* french, elle risque d'interagir facheusement avec la classe TheseCRIN).

Remarques :

– L'entrée dans la table des matières peut être supprimée si `\DontWriteStarChaptersInToc` est donné, ou si `\DontWriteThisInToc` est utilisé immédiatement avant `\PrintIndex`.
– L'entrée peut aussi être encadrée (si elle ne l'est pas par défaut par `\FrameChaptersInToc`) en plaçant `\FrameThisInToc` avant l'environnement.

– On peut ajouter un texte après le titre « Index » et avant la première entrée d'index en utilisant la commande `\BeginIndWith` avant `\PrintIndex`. Exemple :

```
\BeginIndWith{Dans cet index ne figurent que les pages les
plus importantes.}
\PrintIndex
```

8.2.11 Glossaire

L'environnement de glossaire est analogue à l'environnement d'index et crée aussi une entrée dans la table des matières. Cet environnement peut être soit utilisé directement, comme :

```
\begin{TheGlossary}
...
\item \textbf{Chat} : Animal \`a poils
\item \textbf{Souris} : Blanche ou grise, selon les go\^uts.
...
\end{TheGlossary}
```

soit par l'intermédiaire de `\PrintGlossary`, après application de :

```
makeindex -s TCglo.ist -o <fichier>.gls <fichier>.glo
```

On écrirait :

```
...
\makeglossary
...
\begin{document}
...
\Glossary{Chat}{Animal \`a poils.}
...
\Glossary{Souris}{Blanche ou grise, selon les go\^uts.}
...
\PrintGlossary
...
```

Remarques :

– Les commandes `\Glossary` peuvent se trouver n'importe où dans la thèse. Par défaut, les numéros de page où apparaissent les définitions ne sont pas indiqués. Pour les indiquer, on pourra modifier `\Glossary` (avec `\renewcommand`). Exemple : pour mettre la définition en italiques, ainsi que les numéros de page, on écrirait :
`\renewcommand{\Glossary}[2]{\glossary{#1@\textbf{#1} : \textit{#2}}}`

– Le glossaire peut être modifié en changeant certaines commandes dans `TCglo.ist` (qui fait partie de la distribution de la classe TheseCRIN). On lira à cet effet la documentation de `makeindex`.

– L'entrée dans la table des matières peut être supprimée si `\DontWriteStarChaptersInToc` est donné, ou si `\DontWriteThisInToc` est utilisé immédiatement avant `\PrintGlossary`.

– L'entrée peut aussi être encadrée (si elle ne l'est pas par défaut par `\FrameChaptersInToc`) en plaçant `\FrameThisInToc` avant l'environnement.

– On peut ajouter un texte après le titre « Glossaire » et avant la première entrée du glossaire en utilisant la commande `\BeginGloWith` avant `\PrintGlossary`. Exemple :

```
\BeginGloWith{Dans ce glossaire ne figurent que les d\'efinitions
les plus importantes.}
\PrintGlossary
```

8.2.12 Bibliographie

L'environnement de bibliographie se présente comme dans la classe `book`, mais crée une entrée dans la table des matières. Cet environnement peut être soit utilisé directement, comme :

```
\begin{thebibliography}
...
\bibitem{backus78}
John Backus.
\newblock {Can Programming Be Liberated from the von Neumann Style~? A
  Functional Style and Its Algebra of Programs}.
\newblock \emph{CACM}, 21(8):613--641, 1978.
\bibitem{dijkstra72}
Edsger~W. Dijkstra.
\newblock {The Humble Programmer}.
\newblock \emph{CACM}, 15(10):859--866, October 1972.
\bibitem{hoare69}
C.A.R. Hoare.
\newblock {An Axiomatic Basis for Computer Programming}.
\newblock \emph{CACM}, 12(10):576--583, October 1969.
...
\end{thebibliography}
```

soit par l'intermédiaire de `\bibliography`, après application de `bibtex` (cf. chapitre 4).

Remarques :

- L'entrée dans la table des matières peut être supprimée si `\DontWriteStarChaptersInToc` est donné, ou si `\DontWriteThisInToc` est utilisé immédiatement avant `\bibliography`.
- L'entrée peut aussi être encadrée (si elle ne l'est pas par défaut par `\FrameChaptersInToc`) en plaçant `\FrameThisInToc` avant l'environnement `thebibliography` ou la commande `\bibliography`.
- On peut ajouter un texte après le titre « Bibliographie » et avant la première référence en utilisant la commande `\BeginBibWith` avant `\bibliography`. Exemple :

```
\BeginBibWith{La partie \`a gauche de chaque entr\'ee est
ce qui apparait dans le texte lorsque l'on fait une citation.}
\bibliography{mabiblio}
```

8.2.13 Liste des figures et des tables

On peut ajouter un texte après le titre « Liste des figures » et avant la première entrée de la liste en utilisant la commande `\BeginLofWith` avant `\listoffigures`. Exemple :

```
\BeginLofWith{Le num\'ero de page est indiqu\'e \`a droite
de chaque ligne.}
\listoffigures
```

- La position du titre peut être changée avec `\AlignTitlesLeft`, etc. (cf. § 8.2.9).
- `\BeginLotWith` est la commande analogue pour la liste des tables.

8.2.14 Notes de bas de page

Par défaut, les notes de bas de page sont numérotées de manière consécutive tout au long du document. On peut cependant les faire repartir à un à chaque chapitre en écrivant `\ResetFootnotesAtChapters` dans le préambule.

8.2.15 Divers

- Le codage des caractères en entrée est le code ISO Latin-1.
- La commande `\TC` donne 'TheseCRIN'

8.3 Comment étendre **TheseCRIN**?

Comme toute classe, comme tout *package*, comme tout ensemble de commandes TEX ou LATEX, le comportement de TheseCRIN peut-être modifié et étendu.

Cependant, il y a de bonnes et de mauvaises manières de « personnaliser » ou compléter TheseCRIN. Une mauvaise manière serait de faire chez soi une copie du fichier `TheseCRIN.cls` et d'y apporter directement des modifications. C'est une mauvaise démarche parce qu'elle risque de corrompre la classe elle-même et de rendre obscure l'origine d'erreurs qui peuvent en résulter. Pour cette raison, l'auteur de TheseCRIN se refuse à considérer un *quelconque problème* apparaissant avec une version modifiée de cette classe.

La bonne manière consiste à *amender* la classe TheseCRIN au sein du fichier maître qui l'utilise, ou bien dans un *package* qui serait chargé par le document principal. C'est typiquement de cette manière que l'on personnaliserait l'apparence de la première page pour réaliser autre chose qu'un mémoire de thèse. Un certain nombre de commandes de personnalisation sont d'ailleurs disponibles et on se reportera directement au fichier `TheseCRIN.cls` pour en savoir plus. Ce n'est pas ici le lieu de les détailler.

L'avantage de conserver le fichier `TheseCRIN.cls` inchangé est aussi de rendre le passage à une version plus récente de TheseCRIN le plus simple possible.

8.4 En cas de problème

Si un problème apparaît, la première chose à faire est de classer l'erreur dans l'une des catégories suivantes :

1. erreur dans TheseCRIN.cls ;
2. erreur dans `book.cls` ;
3. erreur dans `french.sty` ;
4. erreur dans un autre fichier ;
5. erreur dans LATEX 2_ε.

On avisera ensuite le responsable présumé. On notera que le *package* `french` peut être innocenté si l'erreur persiste lorsqu'on ajoute l'option `nofrench` dans `\documentclass[...]{TheseCRIN}`.

8.5 Index des commandes de **TheseCRIN**

Chapitre 9

L'environnement de travail LaTeX

Mise en garde : *Les chemins et les détails d'installation donnés ici sont ceux qui existaient au moment où ce guide a été rédigé. Il se peut fort bien que l'installation évolue (en particulier au fur et à mesure que l'ensemble des utilisateurs passera à LaTeX 2ε) et que certaines informations contenues ici ne soient plus vraies.*

Ce chapitre présente les variables d'environnement à connaître pour utiliser correctement LaTeX, et décrit les principaux outils associés. Ceci est valable pour toute les machines du réseau loria qui sont des HP sous HP-UX quelle que soit la version, des SUN à architecture SPARC (sun4) fonctionnant sous Sun-OS 4.x (Solaris 1.x) ou sous Solaris 2.x (Sun-Os 5.x), ou des stations SGI[81].

Nous avons essayé ci-après de faire un double tri parmi les différents outils tournant peu ou prou autour de TeX et présents sur notre site. Nous avons d'abord distingué plusieurs classes d'outils puis, au sein de chaque classe, nous avons essayé d'établir une gradation allant des outils pour utilisateur de base aux outils réservés aux experts et autres gourous.

9.1 Chemins d'accès et variables d'environnement

Il n'y a aucun chemin particulier à connaître ou à positionner pour pouvoir utiliser correctement LaTeX et son environnement, puisque tous les binaires se trouvent dans le répertoire standard des binaires locaux, c'est-à-dire `/usr/local/bin` pour presque tout et `/usr/local/X11R5/bin` pour tout ce qui nécessite X-Window. Tous les fichiers utilisés par LaTeX se trouvent sous `/usr/local/lib/tex`[82]. C'est là qu'on trouve les classes, les *packages*, les polices et les formats.

Par défaut tous les outils utilisent des chemins standard, à savoir le répertoire courant (`.`) et le(s) répertoire(s) correspondant(s) sous l'arborescence `/usr/local/lib/tex` :

- `TEXINPUTS` → `.:/usr/local/lib/tex/inputs`
  (`.:/usr/local/lib/tex/inputs/latex2e:/usr/local/lib/tex/inputs` avec LaTeX 2ε)
- `TEXFONTS` → `.:/usr/local/lib/tex/fonts/tfm`
- `TEXFORMATS` → `.:/usr/local/lib/tex/formats`

Si vous voulez par exemple accéder à des *packages* qui ne se trouvent ni dans les chemins standard, ni dans le répertoire courant, il vous suffit d'indiquer le(s) nouveau(x) chemin(s) de la manière suivante :

- `setenv TEXINPUTS /mon/chemin/a/moi:/son/chemin/a/lui:`

Dans ce cas, TeX va ajouter ces chemins aux chemins standard décrits ci-dessus, ce qui signifie qu'au final il cherchera les *packages* dans les chemins :
`/mon/chemin/a/moi:/son/chemin/a/lui:.:/usr/local/lib/tex/inputs:`
Il suffit de faire de même pour les autres variables d'environnement.

Note : si vous utilisez un *package* particulier qui vous semble intéressant pour toute la communauté, vous pouvez envoyer un courrier électronique à latex@loria.fr afin que celui-ci soit installé parmi les *packages* standard.

81. Sauf peut-être quelques-uns des outils mentionnés dans ce chapitre pour ces dernières.
82. ou `/usr/local/include/tex`, un lien symbolique existant entre ces deux noms.

9.2 Les commandes de base

9.2.1 Pour utilisateurs de base

– `tex` est la commande qui lance TEX.

– `latex` lance LATEX, ou plus précisément LATEX 2ε. Si vous avez absolument besoin de lancer LATEX 2.09, il faut taper `latex.old`.

– `slitex.old` lance SLITEX. Rappelons que cette commande est obsolète avec LATEX 2ε, où la classe slides remplace avantageusement l'ancien SLITEX (cf. chapitre 6).

– `bibtex` appelle le programme BIBTEX (cf. chapitre 4).

– `xlatex` permet de lancer une session LATEX (ou TEX) de manière conviviale sous X-Window.

9.2.2 Pour experts

– `tangle` extrait d'un source WEB [Knuth, 1992; Knuth, 1983] la partie programme en Pascal.

– `ctangle` extrait d'un source CWEB [Knuth, 1992; Knuth et Levy, 1993] la partie programme en C.

– `weave` (resp. `cweave`) extrait d'un source WEB (resp. CWEB) la partie documentation en TEX.

– `patgen` : génération de motifs pour les divisions de mots.

9.3 Créer, éditer, corriger un fichier en TEX ou LATEX

– `detex` enlève les commandes TEX d'un texte, ce qui peut être pratique pour compter des mots, vérifier l'orthographe, livrer un texte ASCII « nu » quand on vous le demande, etc.

– `delatex` et `l2a` font la même chose pour un texte en LATEX.

– `ortho` et `ispell` sont des correcteurs orthographiques français et anglais. Ils sont interfacés avec *emacs*. `epelle -latex` est un autre vérificateur d'orthographe, mis au point par notre collègue Paul Zimmermann ; voir le `man` pour plus de détails.

– Le *package* VORTEX sous *emacs* offre un certain nombre de facilités pour saisir un texte source, créer des environnements LATEX, mettre des mots ou des parties en gras ou italiques par exemple, insérer des index ou des références biblographiques, etc. Voir la notice complète d'emploi de ce *package*.

  Ce *package* n'est malheureusement plus maintenu et il faut vivre avec ses qualités **et** ses défauts[83], c'est pourquoi une alternative intéressante pour des débutants peut être de se mettre à AUC-TEX, dont l'approche est un peu différente de VORTEX puisqu'il repose plutôt sur la notion de complétion automatique, mais qui est très facilement adaptable à vos besoins propres. Voir les pages info sur AUC-TEX dans le menu `Help` de *XEmacs*, par exemple.

– `lacheck` vérifie la consistance d'un fichier LATEX.

– `texmex` est un *script* qui convertit des textes écrits avec des caractères 7-bits et un certain nombre de conventions (`l'ide'e est de saisir les caracte`res accentue's comme dans ce passage, ou` me^me en me forc,ant j'ai du mal a` trouver une phrase qui tient debout`).

  Cet utilitaire a rendu pendant des années d'indéniables services, mais à notre avis il n'y a maintenant plus de raison de l'utiliser. En effet, nos systèmes d'exploitation permettent maintenant de travailler directement sur des fichiers de caractères 8-bits (cf. par exemple le french-mode de *emacs*), et si vous vous trouvez dans la situation où il faut quand même avoir un fichier en 7-bits (par exemple parce que vous devez envoyer ce fichier par mail à un collègue qui n'a pas de *mailer* passant le huitième bit, et/ou qui n'a pas d'éditeur travaillant avec le codage ISO Latin-1), vous pouvez toujours utiliser un *package*

83. Un tout petit essai d'adaptation de VORTEX à LATEX 2ε peut cependant être testé en ajoutant `/users/image/tombre/-Emacs/Vortex2e` en tête de son `load-path`.

emacs tel que french-tex-mode, qui présente des caractères accentués à l'écran mais qui sauvegarde avec les conventions de LaTeX[84].

Par ailleurs, il faut tenir compte du fait que `texmex` n'effectue pas d'analyse contextuelle ; par conséquent, les résultats qu'il donne dans un `verbatim` ou un `tabbing` ne sont pas forcément ceux qu'on souhaite.

- `skeltex`, utilitaire local, vous permet de créer un « squelette » de fichiers source LaTeX. C'est un outil interactif ; laissez-vous simplement guider par les questions.
- `tr2tex` convertit un texte source `troff`[85] en source LaTeX. Une évolution de cet outil, `tr2latex`, existe aussi sur notre site mais n'est pas installée dans les binaires standard[86].

9.4 Manipulation des fichiers DVI

9.4.1 Pour utilisateurs de base

- `xdvi` permet de visualiser un fichier d'extension .dvi sous X-Window. Il prend un certain nombre d'options dont certaines peuvent être bonnes à connaître :
  - `-paper a4` indique que le format souhaité pour le support est A4. Dans la version la plus récente installée sur notre site, cette option est compilée par défaut.
  - `-hushspecials` supprime les avertissements (*warnings*) dûs au non traitement des commandes `\special`.
  - `-bg black -fg white` inverse les couleurs par défaut.
  - `-expert` supprime les boutons sur le côté, pour ceux qui connaissent par cœur les raccourcis clavier.
  - `-thorough` évite que du blanc écrase du noir, ce qui peut souvent arriver lorsque des caractères s'inter-pénètrent légèrement ou dans certains diagrammes flèches.
  - `-gamma` <*valeur*> permet de renforcer le contraste, ce qui peut être utile sur une station couleur.
  - `-l` donne les polices chargées par `xdvi` (utile en cas de débogage).
  - `-noghostscript` ou `-nopostscript` remplace les figures *PostScript* par des cadres vides, ce qui accélère la visualisation.

  À noter que `xdvi` peut visualiser des polices *PostScript* à condition de positionner la variable d'environnement suivante :

  `setenv XDVIMAKEPK "xdvimakepk %n %d %b %m"`
- `dvips` convertit un fichier DVI en *PostScript*. Il a de très nombreuses options (cf. § 9.4.2), mais en utilisation standard on peut se contenter de connaître les suivantes :
  - `dvips` *fichier[.dvi]* convertit en *PostScript* et sort sur votre imprimante par défaut (variable d'environnement `PRINTER`).
  - `dvips -P lj3` indique explicitement l'imprimante de sortie.
  - `dvips -f` *fichier* | `lpr -h` est une variante qui peut intéresser ceux qui ne veulent pas de la page d'entête fournie par défaut par `lpr`.
  - `dvips -o` *res.ps fichier* écrit le résultat *PostScript* dans le fichier `res.ps`.
  - `dvips -p 12 -n 3` demande de commencer à la page 12 et de convertir au maximum 3 pages.
  - `dvips -p 12 -l 32` commence à la page 12 et termine à la page 32.
- `dvi2tty` convertit un fichier DVI en ASCII, au cas (malheureux) où vous n'auriez ni imprimante ni système X-Window pour visualiser le fichier, ou si vous n'avez qu'un fichier DVI et que vous voulez récupérer autant que faire se peut le texte ASCII qu'il contient.

84. Attention toutefois : ce *package* peut devenir très lent quand le fichier est gros et ne donne pas toujours le résultat escompté, en particulier dans un environnement `verbatim`.
85. Si vous avez encore des dinosaures de ce type dans vos tiroirs.
86. Voir dans `/local/tex/archive/support/tr2latex`.

9.4.2 Pour utilisateurs avertis

– Certaines versions expérimentales de `xdvi` commencent à apparaître, avec des ajouts tels que des liens de type hypertexte dans le fichier DVI lui-même [87] (ces informations sont gérées grâce à des commandes `\special`).

– Revenons à `dvips` pour en donner quelques options supplémentaires :

- `-D 600` force la résolution à 600 *dpi*, ce qui peut être intéressant quand on sort un texte final sur une imprimante qui a cette résolution (meilleure qualité). Mais attention : le fichier résultat risque d'être plus gros et l'ensemble de l'opération est parfois plus lent, puisque nous avons pour l'instant plus de polices stockées en 300 *dpi* qu'en 600 *dpi* et que par conséquent des appels à `MakeTeXPK` sont possibles.

  À noter que l'option `-P` choisit la résolution déclarée dans le fichier de configuration de l'imprimante en question [88].
- `-d` <*nombre sur 8 bits*> permet d'obtenir diverses informations de débogage, parfois bien utiles. Chaque bit du nombre donné en paramètre correspond à un type d'information. Voici leur signification, telle qu'on la trouve dans le fichier `debug.h` de la distribution de `dvips` :

```
#define D_SPECIAL          (1<<0)
#define D_PATHS            (1<<1)
#define D_FONTS            (1<<2)
#define D_PAGE             (1<<3)
#define D_HEADER           (1<<4)
#define D_COMPRESS         (1<<5)
#define D_FILES            (1<<6)
#define D_MEM              (1<<7)
```

- `-Z` demande une compression des polices *bitmap* avant de les charger, ce qui peut réduire considérablement la taille du fichier *PostScript* envoyé à l'imprimante, en particulier quand on a des polices de 600 *dpi*.
- `-A` ne sort que les pages impaires (`-B` sort les pages paires).
- `-M` désactive la génération automatique par `MakeTeXPK` des polices manquantes.
- `-O` offset déplace l'origine du repère de offset, qui est donné sous une forme telle que `.1cm,-.3cm`. Intéressant pour corrections fines de la position du texte sur une imprimante donnée (mais trop rudimentaire en pratique pour les corrections sur imprimantes recto-verso).
- `-pp 3,6,9-11,15` sort uniquement les pages correspondantes.
- `-q` passe en mode silencieux, où `dvips` ne sort plus que les erreurs sur `stdout`.

Notons aussi pour terminer que vous pouvez personnaliser les options par défaut de `dvips` grâce au fichier `.dvipsrc` dans votre *Home Directory*.

– `dviconcat` concatène plusieurs fichiers DVI pour en faire un nouveau.

– `dviselect` prend une partie d'un fichier DVI, en extrait une liste de pages donnée pour créer un nouveau fichier DVI.

9.4.3 Pour experts

– `dvitype` traduit un fichier de type .dvi en une trace lisible par un humain. Cet utilitaire peut servir à vérifier qu'un fichier DVI est valide, mais son intérêt principal est probablement que c'est un exemple (écrit par Knuth lui-même) de programme lisant dans un fichier DVI, pour les programmeurs qui voudraient manipuler de tels fichiers.

– `dvicopy` copie des pages sélectionnées dans un fichier DVI et remplace les références à des caractères existant dans des fontes virtuelles par les instructions permettant d'adresser ces fontes virtuelles.

87. Voir par exemple dans `/local/tex/archive/Incoming/xhdvi`.
88. `/usr/local/lib/tex/ps/config.`<*imprimante*>.

9.5 Polices de caractères

9.5.1 Pour utilisateurs avertis

- `mf` : appel de METAFONT (génération des fontes).
- `MakeTeXPK` : script d'appel de METAFONT, utilisé automatiquement par `dvips` pour créer les tailles de polices manquantes.
- `xdvimakepk` : variante du précédent pour les fontes utilisées par `xdvi`.
- `bm2font` convertit un certain nombre de formats *bitmap* en polices utilisables par TEX (cf. § 3.7).
- `fig2MF` convertit un fichier .fig en un fichier METAFONT (cf. § 3.7).

9.5.2 Pour experts

Nous rappelons succinctement quelques outils ici. Le lecteur se reportera au chapitre 3 pour plus de détails.

- `afm2tfm` : transformation des métriques de fontes .afm en .tfm (Adobe → TEX).
- `gftopk` : transformation .gf à .pk (pk est le format compact des polices).
- `gftodvi` : transformation .gf à .dvi (gf est le format des fichiers de polices engendrées par METAFONT).
- `gftype` : traduit un fichier .gf en un langage plus humain.
- `mftobdf` : conversion de polices (.mf) en X-Window (.bdf).
- `pktogf` : inverse de `gftopk`.
- `pktype` : conversion du format de police .pk en quelque chose de plus compréhensible.
- `pltotf` : conversion d'une liste de propriétés en fichier de métrique (.tfm).
- `tftopl` : conversion d'une métrique de fonte (.tfm) en une liste de propriétés (inverse de `pltotf`).

9.6 Programmes de manipulation de fichiers *PostScript*

Bien qu'ils ne fassent pas *stricto sensu* partie de l'environnement LATEX, il nous semble utile de donner quelques pointeurs sur une partie des outils *PostScript* disponibles sur le site, puisque nos moyens d'impression utilisent *PostScript* et que par conséquent tout utilisateur de LATEX utilise forcément aussi du *PostScript*, même s'il n'en est pas conscient.

9.6.1 Pour utilisateurs de base

- `ghostview` permet de visualiser un fichier *PostScript* sous X-Window. C'est en fait une interface plus conviviale de *GhostScript*, qui est appelé par la commande `gs`.
- `psrev` inverse l'ordre des pages dans un fichier *PostScript*.
- `alltops`[89] est un utilitaire local qui transforme pratiquement tous les formats *bitmap* couleurs, niveaux de gris ou noir et blanc en *PostScript* encapsulé couleurs ou niveaux de gris.

89. `/local/outils/alltops/alltops`.

9.6.2 Pour utilisateurs avertis

- `pstops` est très utile pour amender un fichier *PostScript*, entre autres pour corriger une mise en page. Par exemple, si on veut imprimer un document en *landscape* sur une imprimante recto-verso, on doit *a priori* agrafer le document sur le bord de 29,7 cm pour pouvoir le feuilleter facilement. Si on veut tourner les pages de manière plus « naturelle », et agrafer sur le bord de 21 cm, il est nécessaire de retourner une page sur deux, ce qui se fait par la commande :

  `pstops '2:0,1U(210mm,297mm)'` *fichier1.ps fichier2.ps*

  De même, pour mettre deux pages sur une seule, en recto simple, on pourra écrire :

  `pstops '2:0L@.7(21cm,0)+1L@.7(21cm,14.85cm)'`

  et en recto-verso :

  `pstops '4:0L@.7(21cm,0)+1L@.7(21cm,14.85cm),3R@.7(0,14.85cm)+2R@.7(0,29.7cm)'`

- `psdraft` permet d'ajouter un texte (« Confidentiel », « Brouillon », « DRAFT » par défaut...) en diagonale en travers de chaque page d'un fichier *PostScript*.

- `psnup` et `psmulti` sont deux utilitaires permettant de placer plusieurs pages sur une seule feuille.

- `psselect` sélectionne certaines pages d'un fichier *PostScript* pour en faire un nouveau.

- `psbook` arrange un certain nombre de pages sur une même page pour préparer l'impression d'un livre.

- `bbps` est un utilitaire local qui calcule la plus petite *BoundingBox* possible pour le fichier *PostScript* de départ et qui rend une copie de ce fichier munie d'une *BoundingBox*.

9.6.3 Pour experts

- `psfix` permet d'éditer et de modifier directement un certain nombre de paramètres et de données dans un fichier *PostScript*. Voir le manuel en ligne pour plus de détails.

9.7 Divers

- `xlp` est un utilitaire local permettant de lancer et de gérer des impressions de manière très conviviale.

- `makeindex` permet d'engendrer des index à partir du fichier créé par LaTeX en analysant toutes les commandes engendrées par `\index` dans le texte. Voir le chapitre 12 du *LaTeX Companion* [Goossens *et al.*, 1994] pour plus de précisions. Cet outil permet aussi de créer des glossaires (commandes `\glossary`).

  À noter que `makeindex` utilise la variable d'environnement `INDEXSTYLE`, qui indique les répertoires dans lesquels chercher le fichier de style souhaité.

- `tgrind` permet de créer des listings de programmes avec mise en page claire et nette avec TeX.

- `lgrind` fait la même chose avec LaTeX.

- `transfig` permet l'insertion de dessins `fig` ou `xfig` (utilise `fig2dev`...).

- `fig2dev` traduit un code `fig` en langage graphique (fonctionne avec `transfig`).

- `fig2ps2tex` engendre un fichier TeX qui inclut lui-même un fichier *PostScript* (fonctionne avec `transfig`). Pour ces trois dernières commandes, voir aussi en § 5.3.

- `latex2html` permet de convertir un document LaTeX en un document hypertexte HTML pouvant être lu par un client WWW.

Nous n'avons sûrement pas été exhaustifs dans cette énumération, mais nous espérons ne pas avoir oublié d'utilitaire vraiment important. Nous sommes bien évidemment reconnaissants pour toutes suggestions permettant de compléter cette liste.

Chapitre 10

Pour en savoir plus

Les utilisateurs de TeX ou LaTeX sont souvent confrontés aux mêmes petits problèmes. L'étude détaillée de la documentation standard (ouvrages de Knuth, Lamport, manuels en ligne) répond généralement — mais pas directement — à ces questions. Des informations complémentaires correspondant à des besoins spécifiques sont disponibles.

10.1 La documentation écrite

Tout au long de ce guide, nous n'avons cessé de vous renvoyer à un certain nombre d'ouvrages ou de manuels. La plupart de ceux qui sont mentionnés en bibliographie sont disponibles sur notre site, que ce soit au Centre de Documentation ou directement en ligne. À ces ouvrages, il faut ajouter des publications périodiques comme « La lettre Gutenberg » et les « Cahiers Gutenberg » édités par le groupe des utilisateurs francophones de TeX, ainsi que la revue TUGboat qui est la publication officielle du *TeX Users Group*. Ces revues rendent compte des dernières recherches concernant TeX.

10.2 Les « FAQ »

Les FAQ (*Frequently Asked/Answered Questions*) rassemblent sous la forme d'un *digest* les questions qui reviennent le plus souvent et les réponses qu'on peut leur apporter. Un FAQ est généralement un fichier associé à un *newsgroup* ou une *mailing-list*.

Un grand nombre de FAQ sont regroupés dans le répertoire `/local/doc/FAQ`. Ces FAQ sont automatiquement remis à jour. Ce sont les FAQS `tex-faq`, `tex-faq-supplement.*` et `fonts-faq.*` qui nous intéressent plus particulièrement. Ces FAQ sont aussi accessibles par `xwaisq`, en demandant le source `faq.src`.

En cas de problème, ces fichiers sont le premier endroit où regarder. Il est mal vu de poser une question s'y trouvant.

Un FAQ sous forme .dvi et .html est accessible via WWW.[90]

10.3 Les *News*

Les principaux *forums* qui nous intéressent ici sont :

- `loria.mi.logiciels.tex` (« couarail TeX »).

  Le couarail TeX est très lu et il est rare qu'une question n'y trouve pas de réponse. Il est cependant clair que son audience est très faible, comparée à celle de `comp.text.tex`.

- `fr.comp.text.tex`

  Il s'agit d'un avatar francophone du *forum* `comp.text.tex`.

- `comp.text.tex`

90. `http://www.loria.fr/tex`.

– `comp.fonts`

Avant de poser une question dans les *news*, il faut chercher dans les FAQ.

10.4 La documentation en ligne

Un certain nombre de documents, soit d'initiation, soit complémentaires, ont été regroupés dans le répertoire `/local/tex/doc`. Son contenu n'est pas détaillé ici car il s'enrichit régulièrement. On peut y accéder sous WWW.[91]

10.5 Archives locales

L'ensemble des *packages* disponibles sur notre site est centralisé dans le répertoire `/local/tex/archive`. Afin de faciliter les recherches, ce répertoire est organisé comme les sites `ftp` du réseau CTAN. Un index des fichiers disponibles est `/local/tex/archive/00files`.

10.6 Sites ftp classiques

Le site `ftp` le plus riche et le plus proche de nous est :

– `LORIA` ⇒ `ftp ftp.loria.fr`, répertoire `pub/unix/tex/ctan`.

Ce site est un miroir des trois sites du réseau CTAN (*Comprehensive TEX Archive Network*) qui sont :

– `Aston` ⇒ `ftp ftp.tex.ac.uk`

– `Houston` ⇒ `ftp ftp.shsu.edu`

– `Stuttgart` ⇒ `ftp ftp.dante.de`

En outre, on pourra consulter :

– `Stanford` ⇒ `ftp labrea.stanford.edu` qui est le site d'origine pour les sources de TEX et METAFONT.

Pour les fichiers plus particulièrement destinés à la version francisée de TEX ou LATEX, on puisera dans :

– serveur `GUT` ⇒ `ftp ftp.univ-rennes1.fr` qui est le site d'origine pour la distribution french.

– Université Paris-Sud, Orsay ⇒ `ftp matups.math.u-psud.fr`

On trouvera une liste plus complète de sites dans le `tex-faq-supplement`.(10.2)

Si vous ramenez des *packages* n'existant pas ici et si vous les estimez intéressants pour la communauté, faites nous le savoir en utilisant l'adresse latex@loria.fr.

10.7 Les *mailing lists*

Voici quelques adresses de *mailing lists* liées à TEX ou à ce qui tourne autour. La requête d'abonnement doit faire partie du corps du message.

– `INFO-TeX` : abonnement avec

```
mail LISTSERV@SHSU.edu
SUBSCRIBE INFO-TeX "Prenom Nom"
```

– `TeX-Pubs` : abonnement avec

```
mail LISTSERV@SHSU.ed
SUBSCRIBE TeX-Pubs "Prenom Nom"
```

91. `http ://www.loria.fr/tex`.

– `ctt-Digest` (`comp.text.tex`) : abonnement avec

```
mail LISTSERV@SHSU.edu
SUBSCRIBE ctt-Digest "Prenom Nom"
```

– `LITPROG` (programmation littéraire) : abonnement avec

```
mail LISTSERV@SHSU.edu
SUBSCRIBE LITPROG "Prenom Nom"
```

– `METAFONT` : abonnement avec

```
mail listserv@ens.fr
SUBSCRIBE METAFONT "Prenom Nom"
```

– `nts-l` (New Typesetting System, un successeur de TEX) : abonnement avec

```
mail listserv@vm.urz.Uni-Heidelberg.de
subscribe nts-l
```

– `GUT` (Groupe (français) des Utilisateurs de TEX) : abonnement avec

```
mail listserv@ens.fr
SUB GUT "Prenom Nom"
```

– `LaTeX 3` : abonnement avec

```
mail listserv@vm.urz.uni-heidelberg.de
subscribe LaTeX-L Prenom Nom
```

10.8 WWW

On pourra trouver de nombreux pointeurs sur WWW,[92] en particulier sur des fichiers de documentation de *packages* de LATEX 2_ε.

Il est possible de naviguer dans les ressources offertes par les sites CTAN. Il suffit — dans votre client WWW — de demander la connexion au serveur mis en place par Norman Walsh[93].

Enfin, depuis la sortie de la version de décembre 1994 de LATEX 2_ε, une nouvelle page html est apparue :

`http://www.tex.ac.uk/CTAN/latex/`

On y trouve de nombreux pointeurs, que ce soit vers des guides ou même vers la liste des disfonctionnements (*bugs*) ayant été signalés.

10.9 Archie

L'outil `archie` (ou `xarchie`) peut permettre de localiser des fichiers ou styles LATEX ne se trouvant pas sur un site `ftp` classique.

92. `http ://www.loria.fr/tex`.
93. Actuellement en `http ://jasper.ora.com/ctan.html`, mais très bientôt de manière locale sur notre site.

Bibliographie

[André et Bur, 1991] Jacques ANDRÉ et Justin BUR. Métrique des fontes PostScript. *Cahiers GUTenberg*, 8 :29–50, Mars 1991.

[André et Grimault, 1990] Jacques ANDRÉ et Jeanine GRIMAULT. Leçons de microtypographie : 1–emploi des capitales (première partie). *Cahiers GUTenberg*, 6 :42–50, Juillet 1990.

[André, 1993] Jacques ANDRÉ. Titres : à lire, à voir ou à dire? *Cahiers GUTenberg*, 15 :33–42, Avril 1993.

[Chicago, 1982] The Chicago Manual of Style. The University of Chicago Press, 1982. The 13th edition of a manual of style revised and expanded.

[CT, 1989] Code typographique. Choix de règles à l'usage des professionnels du livre. Fédération nationale du personnel d'encadrement des industries polygraphiques et de la communication, Paris, 1989. 16e édition.

[Désarménien, 1986] Jacques DÉSARMÉNIEN. La division par ordinateur des mots français : application à TeX. *TSI*, 5(4) :251–265, 1986.

[Girou, 1994] Denis GIROU. Présentation de PSTricks. *Cahiers GUTenberg*, 16 :21–70, Février 1994.

[Goossens *et al.*, 1994] Michel GOOSSENS, Frank MITTELBACH, et Alexander SAMARIN. *The LaTeX Companion.* Addison-Wesley, Reading, MA, USA, 1994.

[Goossens et Rahtz, 1994a] Michel GOOSSENS et Sebastian RAHTZ. Composition en couleur avec LaTeX. *Cahiers GUTenberg*, 16 :5–20, Février 1994.

[Goossens et Rahtz, 1994b] Michel GOOSSENS et Sebastian RAHTZ. Préparer des transparents avec Seminar. *Cahiers GUTenberg*, 16 :71–82, Février 1994.

[Graham *et al.*, 1989] Ronald L. GRAHAM, Donald E. KNUTH, et Oren PATASHNIK. *Concrete Mathematics : A Foundation for Computer Science.* Addison-Wesley, Reading, MA, USA, 1989.

[Grévisse, 1988] Maurice GRÉVISSE. *Le bon usage.* Duculot, 1988. Grammaire française. 12e édition refondue par André Goosse.

[Knuth et Levy, 1993] Donald E. KNUTH et Silvio LEVY. *The CWEB System of Structured Documentation, Version 3.0.* Addison-Wesley, Reading, MA, USA, 1993.

[Knuth, 1983] Donald E. KNUTH. The WEB System of Structured Documentation—ver.2.3. Report No. STAN-CS-83-980, Stanford University, Department of Computer Science, 1983.

[Knuth, 1984] Donald E. KNUTH. The TeXbook. Addison-Wesley, Reading, MA, USA, 1984.

[Knuth, 1986a] Donald E. KNUTH. *Computer Modern Typefaces*, volume E de *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Knuth, 1986b] Donald E. KNUTH. The METAFONTbook, volume C de *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Knuth, 1990] Donald E. KNUTH. Virtual Fonts : More Fun for Grand Wizards. *TUGboat*, 11(1) :13–23, Avril 1990.

[Knuth, 1992] Donald E. KNUTH. *Literate Programming.* CSLI Lecture Notes Number 27. Stanford University Center for the Study of Language and Information, Stanford, CA, USA, 1992.

[Lamport, 1985] Leslie LAMPORT. *LaTeX—A Document Preparation System—User's Guide and Reference Manual.* Addison-Wesley, Reading, MA, USA, 1985.

[Lamport, 1994] Leslie LAMPORT. *LaTeX—A Document Preparation System—User's Guide and Reference Manual.* Addison-Wesley, Reading, MA, USA, 1994. 2nd edition.

[Mauger, 1968] G. MAUGER. *Grammaire pratique du français d'aujourd'hui, langue parlée, langue écrite.* Hachette, Paris, 1968.

[Rolland, 1993] Christian ROLLAND. *LaTeX. Guide pratique.* Addison-Wesley France, SA, Juillet 1993.

[Spivak, 1990] Michael D. SPIVAK. *The Joy of TeX—A Gourmet Guide to Typesetting with the AMS-TeX macro package.* American Mathematical Society, Providence, RI, USA, 2nd revised édition, 1990.

[Strunk Jr. et White, 1979] William STRUNK JR. et E. B. WHITE. *The Elements of Style.* Macmillan Publishing Company, New York, 1979.

[Swan, 1980] Michael SWAN. *Practical English Usage.* Oxford University Press, 1980.

[Thomson et Martinet, 1980] A. J. THOMSON et A. V. MARTINET. *Practical English Grammar.* Oxford University Press, 1980.

[Thomson et Martinet, 1984] A. J. THOMSON et A. V. MARTINET. *Grammaire de l'anglais d'aujourd'hui.* Presses Pocket, 1984. Traduction française de [Thomson et Martinet, 1980] par G. Hardin.

Index

Cet index contient des références vers l'ensemble des classes, *packages*, options, commandes LaTeX ou TeX, commandes UNIX, etc., décrites ou simplement mentionnés dans ce guide. Nous avons voulu ce guide utile et nous n'avons pas hésité à répéter certaines entrées pour permettre de les retrouver selon plusieurs critères.

Les commandes propres à la classe TheseCRIN ne sont pas données ici, mais à la fin du chapitre 8, afin que la distinction entre des commandes « standard », ou à tout le moins répandues, et des commandes purement locales soit la plus claire possible.