Conférence LaTeX n° 8 [1]

LuaLaTeX et polices OpenType, dessins vectoriels avec PSTricks

Denis Bitouzé

denis.bitouze@univ-littoral.fr

https://mt2e.univ-littoral.fr/Members/denis-bitouze/pub/latex

Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville

https://lmpa.univ-littoral.fr/

&

But Métiers de la Transition et de l'Efficacité Énergétiques de Dunkerque

https://mt2e.univ-littoral.fr/

1. Document disponible à l'adresse https://dgxy.link/en-ligne8

Cours LaTeX

De niveaux débutant à avancé

Le présent cours LaTeX est :

- conçu pour des utilisateurs de tous niveaux :
  - débutant
  - intermédiaire
  - avancé
- émaillé de passages [1] de niveau plus avancé, alors signalés par des couleurs de fond spécifiques :

  niveau intermédiaire : jaune pâle

  niveau avancé : rouge pâle

1. Parfois sur plusieurs transparents consécutifs

Plan

Plan

Motivation : LualaTeX... vous avez dit LualaTeX ?

Moteur, format, mode

Moteur : programme informatique

Format : (ensemble de) macros :

- exécutées par un moteur
- utilisant les outils de base du moteur

Mode [1] **:** type de fichiers produits

Remarque

Parfois, développement des formats : limité par le moteur [2]

$\Longrightarrow$ nouveaux moteurs

($\Longrightarrow$ formats [3] plus évolués)

1. De sortie
2. P. ex. prise en charge de certaines langues, des polices OpenType
3. Ou packages

Motivation : LualaTeX... vous avez dit LualaTeX ?

Moteurs + formats + modes courants et commandes les appelant

Format	Mode	Moteur			
		TeX	pdfTeX	XeTeX	LuaTeX
Plain	DVI	tex	etex		dviluatex
	PDF		pdftex	xetex	luatex
LaTeX	DVI		latex		dvilualatex
	PDF		pdflatex	xelatex	lualatex

Définition (LualaTeX)

LualaTeX = moteur LuaTeX + format LaTeX

Remarque

Moteur LuaTeX + format LaTeX + mode PDF : appelé par `lualatex`

Motivation : LualaTeX... vous avez dit LualaTeX ?

Mais vous n'avez pas dit ce qu'était LuaTeX !

- Moteur LuaTeX + format LaTeX = LualaTeX : OK
- mais LuaTeX + LaTeX : quèsaco ?

ulco Université Littoral Côte d'Opale

Motivation : LualaTeX... vous avez dit LualaTeX ?

Mais vous n'avez pas dit ce qu'était LuaTeX !

Format LaTeX :

- macros `\documentclass`, `\usepackage`, etc.
- titres courants, pieds de pages, notes [1], etc.
- flottants
- etc.

Moteur LuaTeX :

- successeur de pdfTeX [2]
- fonctionnalités nouvelles :
  - prise en charge :
    - native du standard Unicode [3]
    - de formats de fontes tels que TrueType et OpenType
  - langage de script Lua embarqué
  - nombreuses bibliothèques Lua dédiées [4]

1. En pied de page, marginales
2. En inclut les fonctionnalités de base (génération directe de pdf, etc.)
3. But : classer/coder les caractères de tous les systèmes d'écriture du monde
4. P. ex. `luamplib` : interpréteur METAPOST intégré à LuaTeX

Motivation : LualaTeX... vous avez dit LualaTeX ?

Et XeTeX dans tout ça ?

Format LaTeX :
- macros `\documentclass`, `\usepackage`, etc.
- titres courants, pieds de pages, notes [1], etc.
- flottants
- etc.

Moteur XeTeX :
- ~~successeur~~ de pdfTeX [2]
- fonctionnalités :
  - prise en charge :
    - native du standard Unicode
    - de formats de fontes tels que TrueType et OpenType
  - ~~langage de script Lua embarqué~~
  - ~~nombreuses bibliothèques Lua dédiées~~
  - < celles de LuaTeX (sauf exceptions)
- développement moins actif ces derniers temps

1. En pied de page, marginales
2. ~~En inclut~~ les fonctionnalités de base (~~génération directe de pdf~~, etc.)

Motivation : LualaTeX... vous avez dit LualaTeX ?

OK, allons-y pour LualaTeX

Tout ceci $\Longrightarrow$ motivés pour l'étude de LualaTeX !

Ce que nous détaillons maintenant

Codage d'entrée des fichiers

Attention !

Avec LualaTeX, codages d'entrée des fichiers :

obligatoire : UTF-8

proscrits :

- ~~ISO-8859-1~~
- ~~Windows-1252~~
- ~~macintosh~~
- etc.

Codage UTF-8

Définition (UTF-8)

- Signifie *Unicode Transformation Format* — 8-bit
- Codage de caractères :
  - pouvant représenter chacun des 1 112 064 caractères Unicode
  - compatible avec l'ASCII
  - utilisant 1 à 4 octets [1]

Définition (codage de caractères)

Processus attribuant des n^{os} aux caractères graphiques [2] :

- notamment ceux écrits du langage humain
- pour les stocker/transmettre/transformer à l'aide d'ordinateurs

1. Et 1 octet (o) = 8 bit (= 1 byte (B))
2. Ou plutôt à leurs n^{os} (points de code dans le cas d'Unicode)

Codage de caractères

Remarque

Plus d'informations sur les codages de caractères ici

Format de sortie : PDF

Remarque (rappel)

$\langle fichier \rangle$.tex $\xrightarrow{\texttt{lualatex}}$ $\langle fichier \rangle$.pdf

Modification des packages à charger

Remarque

LualaTeX = variante de laTeX

$\Longrightarrow$ structure des fichiers `.tex` conservée

Seulement :

- 3 packages à supprimer
- 1 package à ajouter (éventuellement)

Fichier source de base LaTeX

Code source

```
1 \documentclass[french]{article}
2 \usepackage[⟨codage⟩]{inputenc} % Pas vraiment nécessaire[1]
3 \usepackage[T1]{fontenc}
4 \usepackage{lmodern}
5 \usepackage[a4paper]{geometry}
6 \usepackage{babel}
7 \begin{document}
8
9 \end{document}
```

1. Inutile si ⟨*codage*⟩ doit être utf8

Fichier source de base LualaTeX

Code source

```
1 \documentclass[french]{article}
2 % \usepackage{fontspec} % Chargement de polices OT
3 \usepackage[a4paper]{geometry}
4 \usepackage{babel} % ou \usepackage{polyglossia}
5 \begin{document}
6
7 \end{document}
```

Remarque

- fontspec : seulement si polices ≠ *Latin Modern* souhaitées
- Fichier de base sous-entendu désormais [1]

1. Dans cette section consacrée à LualaTeX

Fonctionnalités de TeXstudio

Insertion de modèle de document minimum

Remarque

Modèle de document minimum fourni pour LualaTeX [1]

1. Fichier > Nouveau à partir d'un modèle... > Article (French) for LuaLaTeX

Police par défaut : *Latin Modern*

Exemple

Code source

```
1 Dès Noël où un zéphyr haï me vêt de glaçons
2 würmiens je dîne d'exquis rôtis de bœuf au kir
3 à l'aÿ d'âge mûr \& cætera! 0123456789.
```

Résultat

Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera! 0123456789.

Ce que nous détaillons maintenant

Accès à d'autres polices

... disponibles avec la TeX Live (TL) et avec le système d'exploitation (SE)

Remarque

LualaTeX peut accéder à toutes polices disponibles :

- avec la TeX Live (TL) [1]
- mais aussi avec le système d'exploitation (SE) !

Attention !

Polices { fournies avec la TL/les SE : de (très) bonne qualité [2]
gratuites sur le Net : ~~toujours~~ de (très) bonne qualité [3]

Pas de miracle : police médiocre $\Longrightarrow$ typographie médiocre

1. Tout comme pdfLaTeX
2. En général
3. Site réputé de qualité : https://fontsquirrel.com/

Accès à d'autres polices

... disponibles avec la TL et avec le SE

Attention !

Accès à toutes ces polices :

- complexe *a priori*
- fastidieux *a priori*
- grandement facilité par le package `fontspec`

Dans la suite, on suppose `fontspec` chargé :

Syntaxe

```
\usepackage{fontspec}
```

Accès à d'autres polices

... disponibles avec la TL et avec le SE

Pour une ⟨*police*⟩ autre que *Latin Modern* :

Syntaxe

```
\setmainfont{⟨nom de police⟩}
```

Remarque

`\setmainfont` :

- stipule le ⟨*nom de police*⟩ principale (par défaut) du document
- nécessite le package `fontspec`
- est utilisable en préambule ou dans le corps du document
- est une commande semi-globale

Question : Comment trouver le ⟨*nom de police*⟩ ?

Réponse : Méthodes ≠ selon le SE

Accès à d'autres polices

Obtention de la liste des (noms des) ⟨*polices*⟩

Sous GNU/Linux et Windows [1] :

1. ouvrir un terminal [2]
2. y saisir la commande [3] [4] :

```
fc-list | grep -e opentype -e truetype > polices.txt
```

3. ⏎
4. ouvrir et examiner le contenu de `polices.txt`

Sous macOS : via l'application Livre des polices (*Font Book*)

Attention !

Autres moyens + simples vus plus loin

1. Et éventuellement sous macOS
2. P. ex. via TeXstudio : Outils > Ouvrir le terminal externe
3. Qui liste les polices OpenType/TrueType dans un fichier nommé `polices.txt`
4. Qu'on peut copier-coller depuis ce fichier PDF

Accès à d'autres polices

Exemple : *QTGraphLite*

Nombreuses lignes dans le fichier `polices.txt`, p. ex. :

```
.../QTGraphLite.otf: QTGraphLite:style=Medium
```

On y voit :

- le nom du fichier : `QTGraphLite.otf`
- le nom de la police [1] : *QTGraphLite*
- le style : *Medium* (étudié + loin)

1. Entre les 2 deux-points (1er espace à ignorer)

Accès à d'autres polices

Exemple : *QTGraphLite* (suite)

Exemple

Code source

```
\setmainfont{QTGraphLite}
```

```
1 test
```

Résultat

test

Attributs (rappel)

Notamment sous LaTeX, les polices sont organisées en attributs :

(Sous-)Attribut		Valeur	Signification	Commandes locales/semi-globales	Exemple
Famille		*roman* [1]	romaine	`\textrm{...}/{\rmfamily ...}`	Test
		sans serif	sans empattement	`\textsf{...}/{\sffamily ...}`	Test
		typewriter type	à chasse fixe	`\texttt{...}/{\ttfamily ...}`	Test
Série	graisse [2]	*medium* [1]	moyen	`\textmd{...}/{\mdseries ...}`	Test
		bold face	gras	`\textbf{...}/{\bfseries ...}`	**Test**
	largeur [3]	*medium* [1]	moyen		Test
Forme		*up(right)* [1]	droite	`\textup{...}/{\upshape ...}`	Test
		italic	italique	`\textit{...}/{\itshape ...}`	*Test*
		slanted	incliné [4]	`\textsl{...}/{\slshape ...}`	*Test*
		small caps	petites capitales	`\textsc{...}/{\scshape ...}`	Test

Attention !

Famille *typewriter type* (à chasse fixe) : dite aussi *monospace*

On étudie maintenant ❶ les familles ❷ les séries et formes

1. Par défaut
2. Dites aussi « poids » (*weight*)
3. C.-à-d. « ampleur d'élargissement »
4. Guère fourni que par des polices supportées par LaTeX

Ce que nous détaillons maintenant

Utilisation des polices (familles)

Exemple : *QTGraphLite* (suite)

Exemple

Code source

```
\setmainfont{QTGraphLite}
```

```
1 test
2 \textsf{test} % famille sans serif
3 \texttt{test} % famille monospace
```

Résultat

test test test

Remarque

Seul le 1er « test » est en *QTGraphLite* !

Utilisation des polices (familles)

Solution de repli si familles non spécifiées pour sans serif/monospace

Attention !

Polices non spécifiées pour sans serif/monospace
$\Longrightarrow$ solution de repli (*Latin Modern*) [1]

1. Même si des déclinaisons sans serif et monospace de la famille principale existent

Utilisation des polices (familles)

Commandes

Sous LualaTeX, ~~possibilité~~ quasi-nécessité [1] de sélectionner :

- une police
- pour chacune des 3 familles :
  - principale [2]
  - *sans serif*
  - *monospace*

Syntaxe

```
\setmainfont{⟨nom de police par défaut⟩}
\setsansfont{⟨nom de police pour le sans serif⟩}
\setmonofont{⟨nom de police pour le monospace⟩}
```

1. Sauf recours à un package facilitant les choses
2. Souvent une famille romaine (pas toujours, p. ex. police « fraktur »)

Utilisation des polices (familles)

Exemple

Exemple

Code source

```
\setmainfont{QTGraphLite}
\setsansfont{QTFuture}
\setmonofont{QTPristine}
\NewDocumentCommand{\pangramme}{}{Dès Noël où un zéphyr haï me vêt de glaçons würmiens
  je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr \& cætera! 0123456789.\par}
\begin{itemize}\item\textrm{\pangramme}\item\textsf{\pangramme}\item\texttt{\pangramme}\end{itemize}
```

Résultat

- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera! 0123456789.
- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera! 0123456789.
- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera! 0123456789.

Utilisation des polices (familles)

Exemple (moins fantaisiste : unicité de la police de caractères)

Exemple

Code source

```
\setmainfont{DejaVuSerif}
\setsansfont{DejaVuSans}
\setmonofont{DejaVuSansMono}
\NewDocumentCommand{\pangramme}{}{Dès Noël où un zéphyr haï me vêt de glaçons würmiens
  je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr \& cætera! 0123456789.\par}
\begin{itemize}\item\textrm{\pangramme}\item\textsf{\pangramme}\item\texttt{\pangramme}\end{itemize}
```

Résultat

- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera ! 0123456789.
- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera ! 0123456789.
- Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera! 0123456789.

Ce que nous détaillons maintenant

Utilisation des polices (graisses et formes)

Exemple : *QTGraphLite* (suite)

Remarque

Dans `polices.txt`, *QTGraphLite* n'apparaît qu'à la ligne :

```
.../QTGraphLite.otf: QTGraphLite:style=Medium
```

$\Longrightarrow$ *QTGraphLite* est disponible (seulement) en :

- 1 série [1] : ici gras moyen
- 1 forme : ici droite

1. Ou style

Utilisation des polices (graisses et formes)

Exemple : *QTGraphLite* (suite)

Exemple

Code source

```
\setmainfont{QTGraphLite}
```

```
1 test
2 \textbf{test}                                    % graisse
3 \textit{test} \textsl{test} \textsc{test} % formes
```

Résultat

test ***test*** ***test*** ***test*** ***test***

Remarque

En cas de combinaison indisponible, solution de repli [1]

1. Substitution par une série et/ou une forme par défaut

Utilisation des polices (graisses et formes)

Exemple : *QTAgateType*

Le fichier `polices.txt` contient p. ex. les lignes [1] :

```
.../QTAgateType.otf: QTAgateType:style=Medium
.../QTAgateType-Bold.otf: QTAgateType:style=Bold
.../QTAgateType-Italic.otf: QTAgateType:style=Italic
```

$\Longrightarrow$ *QTAgateType* disponible (seulement) en :

- 2 séries [2] : ici moyen et gras
- 2 formes : droite et italique

1. Et, concernant la police `QTAgateType`, ne contient que ces lignes
2. Ou style

Utilisation des polices (graisses et formes)

Exemple : *QTAgateType* (suite)

Exemple

Code source

```
\setmainfont{QTGraphLite}
```

```
1 test
2 \textbf{test}                                  % graisse
3 \textit{test} \textsl{test} \textsc{test} % formes
```

Résultat

test **test** *test* *test* test

Remarque

En cas de combinaison indisponible, solution de repli

Utilisation des polices

Exemple : *Kp-Fonts*

Le fichier `polices.txt` contient p. ex. :

```
.../KpRoman-Bold.otf:KpRoman:style=Bold
.../KpRoman-BoldItalic.otf:KpRoman:style=BoldItalic
.../KpRoman-Italic.otf:KpRoman:style=Italic
.../KpRoman-Light.otf:KpRoman:style=Light
.../KpRoman-LightItalic.otf:KpRoman:style=LightItalic
.../KpRoman-Regular.otf:KpRoman:style=Regular
.../KpRoman-Semibold.otf:KpRoman:style=Semibold
.../KpRoman-SemiboldItalic.otf:KpRoman:style=SemiboldItalic
.../KpSans-Bold.otf:KpSans:style=Bold
.../KpSans-BoldItalic.otf:KpSans:style=BoldItalic
.../KpSans-Italic.otf:KpSans:style=Italic
.../KpSans-Regular.otf:KpSans:style=Regular
.../KpMono-Bold.otf:KpMono:style=Bold
.../KpMono-BoldItalic.otf:KpMono:style=BoldItalic
.../KpMono-Italic.otf:KpMono:style=Italic
.../KpMono-Regular.otf:KpMono:style=Regular
```

donc *Kp-Fonts* disponible en :

- plusieurs familles [1]
- plusieurs séries
- plusieurs formes

1. Romaine, sans empattement, *monospace*

Utilisation des polices

Attributs (suite)

Remarque

Il peut exister d'autres séries et formes, p. ex. :

graisses : *extra light, light, regular, medium, semibold, extrabold, black* [1] :
Coucou Coucou **Coucou Coucou Coucou Coucou Coucou**

largeurs : *extra condensed, condensed, semi condensed, normal* [2] : Coucou Coucou Coucou Coucou

formes : *up(right), italic, upright italic* [3] : Coucou *Coucou* Coucou

1. Exemples avec la police *NotoSerif*
2. Exemples avec la police *NotoSerif*
3. Exemples avec la police *CMU Serif*

Ce que nous détaillons maintenant

Utilisation des polices

Fonctionnalités des polices

Remarque

Des fonctionnalités peuvent être activées [1] au moyen d'⟨*options*⟩ :

- \setmainfont{⟨*police*⟩}[⟨*options*⟩]
- \setsansfont{⟨*police*⟩}[⟨*options*⟩]
- \setmonofont{⟨*police*⟩}[⟨*options*⟩]
- \defaultfontfeatures{⟨*options*⟩}
- \defaultfontfeatures+{⟨*options*⟩}
- \addfontfeatures{⟨*options*⟩}

1. De façon semi-globale

Utilisation des polices

Fonctionnalités indépendantes des polices

Certaines fonctionnalités sont indépendantes des polices [1], p. ex. :

Couleur : `Color=`⟨*couleur connue de xcolor*⟩

Mise à l'échelle : `Scale=`⟨*nombre*⟩ [2]

ou `Scale=MatchLowercase` [3]

ou `Scale=MatchUppercase` [3]

1. Effet assuré pour toutes polices OpenType
2. P. ex. pour un agrandissement de 20 % : `Scale=1.2`
3. Minuscules (resp. majuscules) de même hauteur que celles de la police :
   - sélectionnée
   - romaine par défaut

Utilisation des polices

Fonctionnalités indépendantes des polices : couleurs (exemple)

Exemple

Code source

```
\usepackage{xcolor}
\setmainfont{Linux Libertine O}[Color=magenta]
\setsansfont{QTDoghaus}[Color=red]
\setmonofont{Nimbus Mono}[Color=blue]
```

```
1 \begin{description}
2 \item[Romaine    :] \textrm{Test}
3 \item[Sans serif :] \textsf{Test}
4 \item[Monospace  :] \texttt{Test}
5 \end{description}
```

Résultat

Romaine : Test

Sans serif : Test

Monospace : Test

Code source (mise à l'échelle)

```
\usepackage{xcolor}
\setmainfont{QTBodiniPoster}[Color=red]
\setsansfont{Cyklop}
```

```
1  The perfect match \textsf{is hard to find}.
2
3  \setsansfont[Scale=0.4]{Cyklop}
4  The perfect match \textsf{is hard to find}.
5
6  \setsansfont[Scale=MatchLowercase]{Cyklop}
7  The perfect match \textsf{is hard to find}.
8
9  \setsansfont[Scale=MatchUppercase]{Cyklop}
10 The perfect matcH \textsf{is hard to find}.
```

The perfect match is hard to find.

The perfect match is hard to find.

The perfect match is hard to find.

The perfect matcH is hard to find.

Utilisation des polices

Fonctionnalités dépendantes des polices

Certaines fonctionnalités sont dépendantes des polices [1]

1. Effet pas assuré pour toutes polices

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres

Aspect des chiffres modifiable au moyen de l'option :

Syntaxe

`Numbers=`⟨*valeur*⟩

`Numbers={`⟨*valeur*$_1$⟩`,`⟨*valeur*$_2$⟩`}`

où chaque ⟨*valeur*⟩ à choisir parmi :

1.
   - soit `Monospaced` : monospace [1]
   - soit `Proportional` : proportionnel
2.
   - soit `Uppercase` : en capitales [1] [2]
   - soit `Lowercase` : en bas de casse [3]

1. Par défaut
2. = `Lining` : sur la m̂m ligne (pas de jambages) et de m̂m hauteurs
3. = `OldStyle` : style ancien (chiffres elzéviriens)

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres

Code source (exemple)

```
1 \defaultfontfeatures{Numbers=OldStyle}
```

Code source (exemple)

```
1 \addfontfeatures{Numbers={Proportional,OldStyle}}
```

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (effets)

	Numbers	Monospaced [1]	Proportional
Tableau	Uppercase [1]	9876543210 1111111111	9876543210 1111111111
	Lowercase	9876543210 1111111111	9876543210 1111111111
Texte	Uppercase [1]	RDV le 11 mai 2011	RDV le 11 mai 2011
	Lowercase	RDV le 11 mai 2011	RDV le 11 mai 2011

Attention

`Numbers=Proportional` :

- conseillé dans un tableau
- conseillé dans du texte ordinaire

1. Par défaut

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (effets)

Exemple

Code source

```
\setmainfont{TeX Gyre Adventor}[Numbers={Proportional,Lowercase}]
```

```
1 %
2   \begin{tabular}{@{}cccc@{}}
3     1842 & 999 & 75 & 13 \\
4     1923 & 111 & 54 & 56
5   \end{tabular}
6 %
```

Résultat

1842	999	75	13
1923	111	54	56

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (effets)

Exemple

Code source

```
\setmainfont{TeX Gyre Adventor}[Numbers={Proportional,Lowercase}]
```

```
1 {\addfontfeatures{Numbers={Monospaced,Uppercase}}
2   \begin{tabular}{@{}cccc@{}}
3     1842 & 999 & 75 & 13 \\
4     1923 & 111 & 54 & 56
5   \end{tabular}
6 }
```

Résultat

1842	999	75	13
1923	111	54	56

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (effet)

Code source (exemple)

```
1 {\addfontfeatures{Numbers={Monospaced,Uppercase}}
2   ⟨tableau⟩
3 }
```

à chaque ⟨*tableau*⟩ ? Pénible !

Remarque

On peut s'en sortir au moyen de « *hooks* » [1]

1. « Crochets » : fonctionnalité relativement récente (cf. *LaTeX's hook management*)

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (ex. de configuration)

Exemple

Code source

```
\setmainfont{TeX Gyre Adventor}[Numbers={Proportional,Lowercase}]
%
```

```
1 %
2   \begin{tabular}{@{}cccc@{}}
3     1842 & 999 & 75 & 13 \\
4     1923 & 111 & 54 & 56
5   \end{tabular}
6 %
```

Résultat

1842	999	75	13
1923	111	54	56

Utilisation des polices

Fonctionnalités dépendantes des polices : aspects des chiffres (ex. de configuration)

Exemple

Code source

```
\setmainfont{TeX Gyre Adventor}[Numbers={Proportional,Lowercase}]
\AddToHook{env/tabular/before}{\addfontfeatures{Numbers={Monospaced,Lining}}}
```

```
1 %
2   \begin{tabular}{@{}cccc@{}}
3     1842 & 999 & 75 & 13 \\
4     1923 & 111 & 54 & 56
5   \end{tabular}
6 %
```

Résultat

1842	999	75	13
1923	111	54	56

Utilisation des polices

Fonctionnalités (in)dépendantes des polices : ligatures de TeX

Avec (La)TeX, plusieurs ligatures par défaut :

Code	Résultat (~~Résultat~~)
`--`	– [1] (--)
`---`	— [2] (---)
`` ` ``	‘ (`)
`'`	’ (')
`` `` ```	“ (``)
`'' ` [3]	” (")

1. Tiret demi-cadratin
2. Tiret cadratin
3. Ou " (sauf si `babel-french` est en action)

Utilisation des polices

Fonctionnalités dépendantes des polices : ligatures communes

Avec de nombreuses polices, ligatures communes suppl. [1] :

Code	Résultat (~~Résultat~~)
`ff`	ff (ff)
`fi`	fi (fi)
`fl`	fl (fl)
`ft`	ft (ft)
`tt`	tt (tt)
`ffi`	ffi (ffi)
`ffl`	ffl (ffl)
`Th`	Th (Th)
`Qu`	Qu (Qu)
`?!`	?! (?!)
`!!`	‼ (!!)

1. Exemples ici avec *Linux Libertine O*

Utilisation des polices

Fonctionnalités dépendantes des polices : ligatures historiques

Avec certaines polices, ligatures historiques [1] :

Exemple

Code source

```
1 Quelle actrice époustouflante !
2
3 \addfontfeatures{Ligatures=Historic}
4 Quelle actrice époustouflante !
```

Résultat

Quelle actrice époustouflante !
Quelle actrice époustouflante !

1. Exemples ici avec *Linux Libertine O*

Utilisation des polices

Fonctionnalités dépendantes des polices : glyphes alternatifs

Avec certaines polices, glyphes alternatifs [1] :

Exemple

Code source

```
1 Roméo \& Juliette
2
3 \addfontfeatures{Style=Alternate}
4 Roméo \& Juliette
```

Résultat

Roméo & Juliette
Roméo & Juliette

1. Exemple ici avec *Linux Libertine O*

Choix de polices

Disponibles avec la TL

La rubrique *Fonts with OpenType or TrueType Support* [1] :

- liste/exemplifie toutes les polices disponibles avec la TL
- pour chacune d'elle, en indique l'usage

Attention !

Pour la plupart de celles « *OTF or TTF available* », usage :

- pour LaTeX
- plutôt que pour LualaTeX

Pour LualaTeX :

- supprimer ~~`\usepackage[T1]{fontenc}`~~ si présent
- cf. + loin une autre façon de procéder

1. Du *The LaTeX Font Catalogue*

Choix de polices

Disponibles avec la TL

Remarque

Ajustement des options du package `fontspec` :

- pas toujours simple
- simplifié par plusieurs packages

Choix de polices

Disponibles avec la TL : exemple

Exemple

Code source

```
\usepackage{accanthis}
```

```
1 Dès Noël où un zéphyr haï me vêt de glaçons
2 würmiens je dîne d'exquis rôtis de bœuf au kir
3 à l'aÿ d'âge mûr \& cætera! 0123456789.
```

Résultat

Dès Noël où un zéphyr haï me vêt de glaçons würmiens je dîne d'exquis rôtis de bœuf au kir à l'aÿ d'âge mûr & cætera ! 0123456789.

Choix de polices

Disponibles avec la TL et le SE

Remarque

On peut consulter ce *catalogue* [1] de Daniel FLIPO pour :

- choisir visuellement
- une police libre

1. Non exhaustif

Ce que nous détaillons maintenant

Utilisation des polices

Mathématiques : package unicode-math

Attention !

Sélection des polices math. avec :

- fontspec : limitée
- unicode-math : assurée [1]

1. Autre package : mathspec, mais moins puissant

Utilisation des polices

Avec fontspec seulement

Exemple

Code source

```
\usepackage{fontspec}      % fontspec seul
\setmainfont{TeX Gyre Bonum}
%
```

```
1 La plus belle formule de mathématiques est :
2 \[e^{i\pi} + 1 = 0\]
```

Résultat (polices texte et math. inadaptées)

La plus belle formule de mathématiques est :

$$e^{i\pi} + 1 = 0$$

Utilisation des polices

Avec unicode-math à la place de fontspec

Exemple

Code source

```
\usepackage{unicode-math} % charge fontspec
\setmainfont{TeX Gyre Bonum}
\setmathfont{TeX Gyre Bonum Math}
```

```
1 La plus belle formule de mathématiques est :
2 \[e^{i\pi} + 1 = 0\]
```

Résultat (polices texte et math. adapatées)

La plus belle formule de mathématiques est :

$$e^{i\pi} + 1 = 0$$

Utilisation des polices

Mathématiques : package unicode-math

Le package unicode-math :

- charge en sous-main :
  - fontspec
  - amsmath
- doit être chargé après :
  - amsmath, si chargé explicitement [1]
  - tout package chargeant amsmath [2]

Syntaxe

```
% Après \usepackage{mathtools}
\usepackage{unicode-math} % supposé chargé dans la suite
```

1. P. ex. pour lui passer des options spécifiques
2. P. ex. mathtools

Utilisation des polices

Mathématiques : package unicode-math

Syntaxe

\setmathfont{⟨police mathématique⟩}

Remarque

\setmathfont :

- stipule le ⟨nom de police mathématique⟩ [1] du document
- nécessite le package unicode-math
- est utilisable en préambule ou dans le corps du document
- est une commande semi-globale

1. OpenType nécessairement

Utilisation des polices

Mathématiques : package `unicode-math`

Question : Comment trouver une ⟨*police math.*⟩ ?

Réponse : Cf. p. ex. :

- le *catalogue de polices libres OpenType pour LuaTeX/XeTeX* [1]
- *Which OpenType Math fonts are available?*

1. Repérer celles qui sont estampillées « M »

Utilisation des polices

Mathématiques : package unicode-math (autres fonctionnalités)

Exemple

Code source

```
1 \[ζ(z) = \frac{1}{Γ(z)}
2          \int_0^\infty \frac{t^{z-1}}{e^t-1}\,dt\]
```

Résultat

$$\zeta(z) = \frac{1}{\Gamma(z)} \int_0^\infty \frac{t^{z-1}}{e^t - 1}\, dt$$

Remarque

Possibilité de caractères Unicode en entrée dans les formules

Utilisation des polices

Mathématiques : package unicode-math (autres fonctionnalités)

Possibilité de styles math. autres que celui de TeX :

math-style= [1]	Latin		Grec	
TeX [2]	$a\ z$	$B\ X$	$\alpha\ \beta$	$\Gamma\ \Xi$
french	$a\ z$	$\mathrm{B}\ \mathrm{X}$	$\alpha\ \beta$	$\Gamma\ \Xi$
ISO	$a\ z$	$B\ X$	$\alpha\ \beta$	$\mathit{\Gamma}\ \mathit{\Xi}$
upright	$\mathrm{a}\ \mathrm{z}$	$\mathrm{B}\ \mathrm{X}$	$\alpha\ \beta$	$\Gamma\ \Xi$

1. Option à passer au package unicode-math
2. Par défaut

Utilisation des polices

Mathématiques : package unicode-math (autres fonctionnalités)

Nombreuses autres fonctionnalités non détaillées, p. ex. [1] :

caractères mathématiques :

gras : $\alpha \neq \mathbf{\alpha} \neq \boldsymbol{\alpha}$

sans serif : $a \neq \mathsf{a} \neq \mathsf{\alpha}$

ajourés :

- $\mathbb{ABCDEFGHIJKLM\,NOPQRSTUVWXYZ}$
- $\mathbb{abcdefghijklmnopqrstuvwxyz}$
- $\mathbb{0123456789}$
- $\mathbb{\gamma\pi\Gamma\Pi}$
- $\mathbb{\sum}$

calligraphiques : $\mathscr{ABCXYZ}$

négation de symboles : $\nleftrightarrow$

1. Ici avec la police XITS Math

Ce que nous détaillons maintenant

LualaTeX versus pdflaTeX

Package PSTricks et dérivés

Moteur	.pdf obtenu directement	Compat. PSTricks
pdflaTeX	✔	✘
laTeX	✘[1]	✔
XeLaTeX	✔	✔
LualaTeX	✔	✔[2]

1. .tex $\xrightarrow{\text{latex}}$ .dvi $\xrightarrow{\text{dvips}}$ .ps $\xrightarrow{\text{ps2pdf}}$ .pdf
2. Depuis 09/2021, grâce au package luapstricks agissant en sous-main

LualaTeX versus pdflaTeX

Package listings

On a :

$$\left.\begin{array}{l}\text{codage d'entrée UTF-8}\\ \text{package listings}\\ \text{accents dans un listing}\end{array}\right\} + \left\{\begin{array}{l}\text{pdflaTeX} \Longrightarrow \text{complications}\,[1]\\ \text{LualaTeX} \Longrightarrow \cancel{\text{complications}}\end{array}\right.$$

1. Cf. p. ex. https://dgxy.link/en-ligne4, « Accents dans les listings »

LualaTeX versus pdflaTeX

Packages pas ou peu compatibles

Remarque

Certains packages :

- compatibles avec pdflaTeX
- sont pas ou peu compatibles avec LualaTeX

P. ex., sur 6 fonctionnalités de `microtype` [1] :

avec pdflaTeX : toutes activées ou disponibles

avec LualaTeX : 3 non disponibles

1. Améliore le gris typographique notamment par *font expansion* et *character protrusion*

LualaTeX versus pdflaTeX

Fonctionnalités pas ou peu compatibles

Remarque

Certaines fonctionnalités :

- compatibles avec pdflaTeX
- sont pas ou peu compatibles avec LualaTeX

Avec `pgfplots` [1] [2], l'option [3] `/pgf/number format/read comma as period` [4]

$$\Longrightarrow \begin{cases} \text{soit compilateur} \neq \sout{\texttt{lualatex}} \\ \text{soit compilateur} = \texttt{lualatex} \text{ mais alors option } \texttt{lua backend=false} \text{ [5]} \end{cases}$$

1. Package dédié à la création de courbes/surfaces de fonctions/données expérimentales
2. Cf. p. ex. `https://dgxy.link/en-ligne10`
3. Fournie par `tikz`
4. À employer si et seulement si virgule (~~point~~) = sép. décimal dans les fichiers de données
5. $\Longrightarrow$ renoncer à des temps de compilation et de gestion de la mémoire améliorés

Transparents de niveau avancé

Séquence du ou des quelques transparents suivants :

- de niveau avancé, significativement plus élevé
- traite de détails omissibles en 1re approche
- peut, sur chacun d'eux, être :

  passée au moyen du bouton ⏭
  réentamée au moyen du bouton ◀

Remarque

Présent transparent : ~~pas répété~~ avant la ou les prochaines séquences de transparents de niveau avancé (signalés par leur fond de couleur rouge pâle)

Ce que nous détaillons maintenant

LualaTeX versus XeLaTeX

Packages ne fonctionnant qu'avec XeLaTeX

Certains packages fonctionnent :

- avec XeLaTeX
- pas avec LualaTeX. Parmi ceux-ci, p. ex. :
  - `xpinyin`
  - `hanzibox`
  - etc. [1]

1. D'autres se trouvent (certainement) parmi ceux listés ici

Compilation indifférente pdflaTeX/LualaTeX

Possibilité de compiler un `.tex`

- indifféremment :
  - soit avec pdflaTeX
  - soit avec LualaTeX [1]
- sans modification du préambule

Syntaxe

```
\usepackage{iftex}
\ifpdftex
⟨code spécifique à pdflaTeX⟩
\else
⟨code spécifique à un autre moteur : LualaTeX (ou XeLaTeX, etc.)⟩
\fi
```

1. Ou avec XeLaTeX

Compilation indifférente pdfLaTeX/LuaLaTeX

Exemple d'usage

Code source (exemple d'usage)

```
\usepackage{mathtools}
\usepackage{iftex}
\ifpdftex
  \usepackage[T1]{fontenc}
  \usepackage[utf8]{inputenc}
  \usepackage{kpfonts}
  \usepackage{listingsutf8}           % complic. propre à pdfLaTeX
  \lstset{inputencoding=utf8/latin1} % complic. propre à pdfLaTeX
\else
  \usepackage{unicode-math}
  \usepackage{kpfonts-otf}
  \usepackage{listings}
\fi
%
\lstset{⟨options de `listings` communes à pdfLaTeX et à LuaLaTeX⟩}
```

Compilation indifférente pdfLaTeX/LuaLaTeX

Exemple d'usage (en fait simplifié de nos jours : `kpfonts-otf` autom. appelé si LuaLaTeX)

Code source (exemple d'usage (en fait simplifié))

```
\usepackage{mathtools}
\usepackage{iftex}
\ifpdftex
  \usepackage[T1]{fontenc}
  \usepackage[utf8]{inputenc}
  %
  \usepackage{listingsutf8}           % complic. propre à pdfLaTeX
  \lstset{inputencoding=utf8/latin1} % complic. propre à pdfLaTeX
\else
  \usepackage{unicode-math}
  %
  \usepackage{listings}
\fi
\usepackage{kpfonts}                  % commun à pdfLaTeX et à LuaLaTeX
\lstset{⟨options de `listings` communes à pdfLaTeX et à LuaLaTeX⟩}
```

Ce que nous détaillons maintenant

Exécution de code Lua avec LualaTeX

Exécution de code Lua possible au moyen notamment de :

Syntaxe

```
\directlua{tex.sprint(⟨code Lua⟩)}
```

Attention!

Fonctionnalité :

- disponible avec LualaTeX
- ~~disponible~~ avec XeLaTeX

Exécution de code Lua avec LuaLaTeX

Exemple

Exemple

Code source

```
1 Une valeur approchée de $\pi$
2 est $\directlua{tex.sprint(math.pi)}$.
```

Résultat

Une valeur approchée de π est 3.1415926535898.

Exécution de code Lua avec LuaLaTeX

Exemple bis

Exemple

Code source

```
\NewDocumentCommand\ImageFonction{ m m }{%
  $\directlua{x=#1 tex.sprint(#2)}$%
}
```

```
1 Soit $f$ la fonction définie par $f:x \mapsto x^2-3x+7$.
2 \begin{itemize}
3 \item L'image de $10$ par $f$ est \ImageFonction{10}{x*x-3*x+7}.
4 \item L'image de $15$ par $f$ est \ImageFonction{15}{x*x-3*x+7}.
5 \end{itemize}
```

Résultat

Soit f la fonction définie par $f : x \mapsto x^2 - 3x + 7$.

- L'image de 10 par f est 77.
- L'image de 15 par f est 187.

Exécution de code Lua avec LuaLaTeX

Exemple (moins poussif)

Exemple

Code source

```
\usepackage{luacas} % système de calcul formel portable
```

```
1 \begin{CAS}
2 vars('x','y')
3 f = 3*x*y - x^2*y
4 fxy = diff(f,x,y)
5 \end{CAS}
6 $\print{fxy} = \print*{fxy}$
```

Résultat

$$\frac{\partial^2}{\partial y \partial x}\left(3xy - x^2y\right) = 3 - 2x$$

Ce que nous détaillons maintenant

Pour aller plus loin

- `texdoc␣lualatex` (un peu ancien) [1]
- `texdoc␣-l␣lualatex` : liste les packages propres à LuaLaTeX
- `texdoc␣fontspec`
- `texdoc␣unicode-math`
- De pdfLaTeX à LuaLaTeX (de Daniel Flipo)
- *Cahiers* GUTenberg consacrés à LuaTeX
- Articles des *Lettres* GUTenberg consacrés aux fontes OpenType
- Si on en a le courage :
  - `texdoc␣luatex`
  - `https://luatex.org/`

1. Et traduit en français

Plan

ulco Université Littoral Côte d'Opale

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

Qu'est-ce que PSTricks ?

PSTricks = ensemble de commandes

- créant des dessins
- utilisables directement dans les fichiers `.tex`

PSTricks peut être chargé *via* :

Code source

```
\usepackage{pst-all}
```

Remarque

`pst-all` : ne charge en fait pas tous les packages PSTricks

Dans la suite, on suppose `pst-all` chargé

Avantages et inconvénients

Avantages :

- Outil extrêmement puissant
- Outil extrêmement précis (non-WYSIWYG)
- Nombreuses extensions spécialisées

Inconvénients :

- Outil non-WYSIWYG
- ~~Utilisable~~ avec pdfLaTeX

Remarque

PSTricks utilisable avec :

- LaTeX ⟶ `dvips` ⟶ `ps2pdf`
- LuaLaTeX [1] ou XeLaTeX

1. Depuis septembre 2021

Ce que nous détaillons maintenant

Premier exemple

et dimension des objets PSTricks

Exemple

Code source

```
1 \psline(2,1)
```

Résultat

Remarque

Objets graphiques créés :

- pas de dimension propre
- c.-à-d. aucun espace réservé par LaTeX [1]

1. Mais c'est bien sûr modifiable : cf. + loin

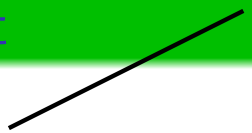
Origine du système de coordonnées

Exemple

Code source

```
1 Bonjour \psline(2,1)
```

Résultat

Bonjour 

Remarque

Origine du système de coordonnées : point courant de LaTeX

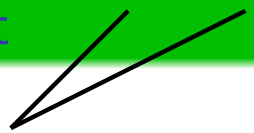
Système de coordonnées des objets PSTricks

Exemple

Code source

```
1 Bonjour \psline(2,1) \psline(1,1)
```

Résultat

Bonjour 

Remarque

Objets :

- créés au même point (LaTeX)
- partagent le même système de coordonnées

Dimensions des objets PSTricks (bis)

Leur réserver de l'espace

Exemple

Code source

```
1 Bonjour \psline(2,1) les amis !
```

Résultat

Bonjour les amis !

Remarque

Objets graphiques créés :

- aucun espace réservé par LaTeX (déjà vu)
- sauf si recours à l'environnement `pspicture`

Réserver de l'espace à un dessin

Syntaxe

```
\begin{pspicture}(x0,y0)(x1,y1)
  ⟨code PSTricks⟩
\end{pspicture}
```

réserve au ⟨*code PSTricks*⟩ une zone rectangulaire :

- de point bas gauche (x_0, y_0)
- de point haut droit (x_1, y_1)

Remarque

(x_0, y_0) omis $\Longrightarrow$ remplacé par $(0, 0)$

Réserver de l'espace à un dessin

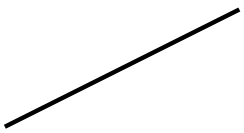
Exemple

Exemple

Code source

```
1 Bonjour
2 \begin{pspicture}(0,0)(2.1,1.1)
3   \psline(2,1)
4 \end{pspicture}
5 les amis !
```

Résultat

Bonjour  les amis !

Réserver de l'espace à un dessin

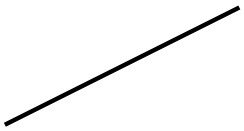
Exemple (bis)

Exemple

Code source

```
1 Bonjour
2 \begin{pspicture}(2.1,1.1)
3   \psline(2,1)
4 \end{pspicture}
5 les amis !
```

Résultat

Bonjour  les amis !

Exemple de base

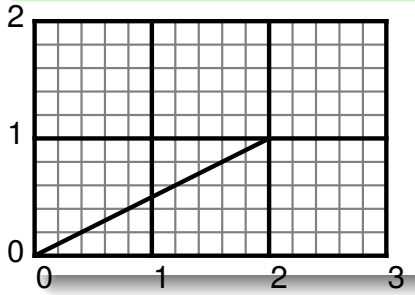
Pour mieux voir ce que l'on fait

Exemple

Code source

```
1 \begin{pspicture}(3,2)
2   \psgrid
3   \psline(2,1)
4 \end{pspicture}
```

Résultat



Remarque

`\psgrid` sans argument $\Longrightarrow$ grille occupant toute la figure

Exemple de base

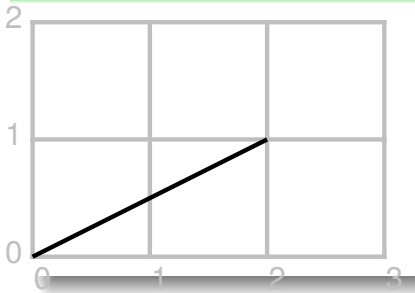
Pour encore mieux voir ce que l'on fait

Exemple

Code source

```
\psset{%
  subgriddiv=0,
  gridcolor=lightgray,
  gridlabelcolor=lightgray
}
% PSTricks charge xcolor
\begin{pspicture}(3,2)
  \psgrid
  \psline(2,1)
\end{pspicture}
```

Résultat



Configuration désormais adoptée pour toutes les grilles

Ordre non indifférent

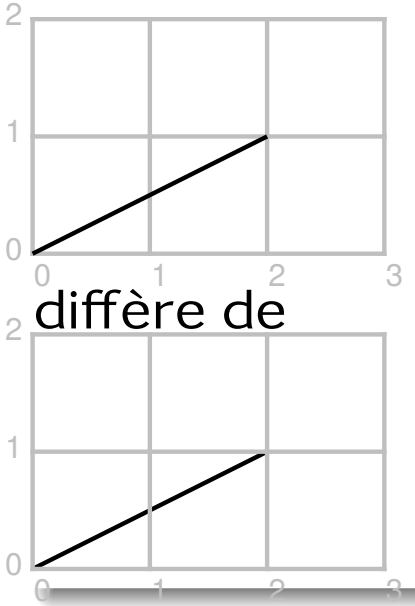
Attention à l'ordre !

Exemple

Code source

```
1  \begin{pspicture}(3,2)
2    \psgrid
3    \psline(2,1)
4  \end{pspicture}
5  \par
6  diffère de
7  \par
8  \begin{pspicture}(3,2)
9    \psline(2,1)
10   \psgrid
11 \end{pspicture}
```

Résultat



Exemple de base

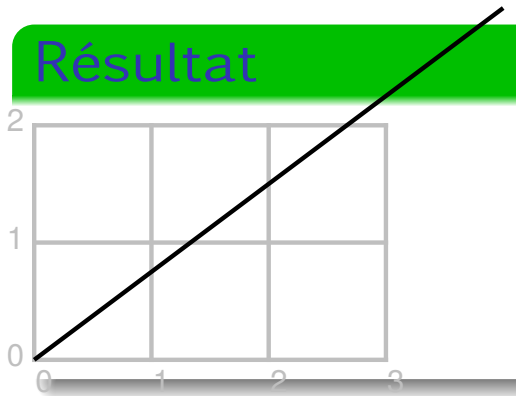
Ça déborde !

Exemple

Code source

```
1 \begin{pspicture}(3,2)\psgrid
2   \psline(4,3)
3 \end{pspicture}
```

Résultat



Ce qui déborde de la figure : dessiné avec `pspicture`

Exemple de base

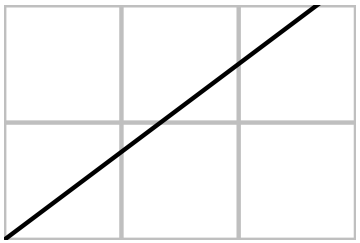
Ça ne déborde plus !

Exemple

Code source

```
1 \begin{pspicture*}(3,2)\psgrid
2   \psline(4,3)
3 \end{pspicture*}
```

Résultat



Ce qui déborde de la figure : ~~dessiné~~ avec `pspicture*`

Unités

Unité :

- par défaut : centimètre
- redéfinissable :

Syntaxe

```
\psset{unit=n⟨dimension⟩}
```

où ⟨*dimension*⟩ vaut p. ex. `1.5cm`

Unités

Attention (recommandation)!

Dans les graphiques, unités des dimensions non explicitées

Ainsi, changements d'échelles : par simple redéfinition de l'unité par défaut

Unités des abscisses et des ordonnées

Unités des abscisses/ordonnées : modifiables indépend^{t}

Syntaxe (modification des unités par défaut)

```
\psset{xunit=⟨dimension1⟩,yunit=⟨dimension2⟩}
```

Modification des unités

Exemple

Exemple

Code source

```
\NewDocumentCommand{\test}{}{Bonjour \psline(2,1)les amis !\par
    }
```

```
1                                                    \test{}
2 \psset{unit=.5cm,              linecolor=red  } \test{}
3 \psset{xunit=.5cm,yunit=1cm, linecolor=blue } \test{}
4 \psset{xunit=.5cm,yunit=-1cm,linecolor=green} \test{}
```

Résultat

Bonjour les amis!
Bonjour les amis!
Bonjour les amis!
Bonjour les amis!

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

Les lignes

Syntaxe

\psline[⟨*param.*⟩]{⟨*flèche(s)*⟩}$(x_0,y_0)(x_1,y_1)\cdots(x_n,y_n)$

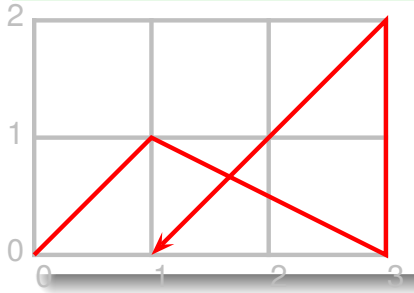
Les lignes

Exemple

Code source

```
1 \begin{pspicture}(3,2)\psgrid
2   \psline[linecolor=red]{->}(0,0)(1,1)(3,0)(3,2)(1,0)
3 \end{pspicture}
```

Résultat



Les lignes

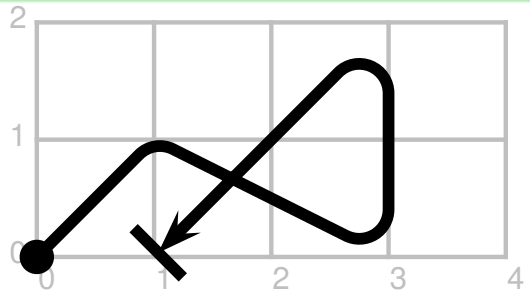
Nombreux paramètres

Exemple

Code source

```
1 \begin{pspicture}(-.5,-.5)(3.5,2)\psgrid
2   \psline[linewidth=.1,linearc=.25]{*->|}(0,0)(1,1)
      (3,0)(3,2)(1,0)
3 \end{pspicture}
```

Résultat



Les polygones

Syntaxe

\pspolygon[⟨*param.*⟩]$(x_0,y_0)(x_1,y_1)\cdots(x_n,y_n)$

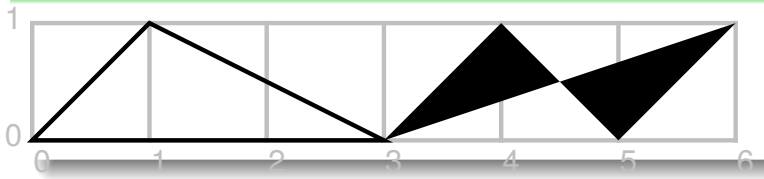
Les polygones

Exemple

Code source

```
1 \begin{pspicture}(6,1)\psgrid
2   \pspolygon(1,1)(3,0)
3   \pspolygon*(3,0)(4,1)(5,0)(6,1)
4 \end{pspicture}
```

Résultat



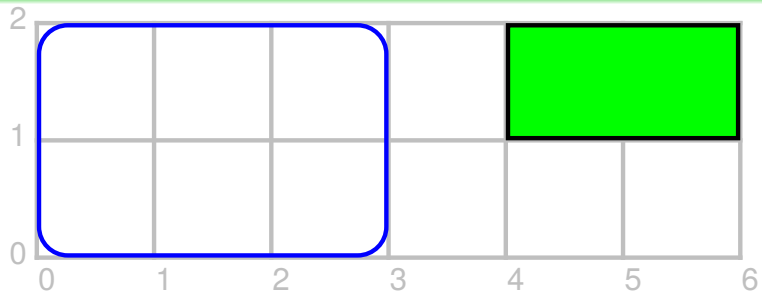
Les rectangles

Exemple

Code source

```
1 \begin{pspicture}(-.5,-.5)(6,2)\psgrid
2   \psframe[framearc=.25,linecolor=blue](3,2)
3   \psframe[fillstyle=solid,fillcolor=green](4,1)(6,2)
4 \end{pspicture}
```

Résultat



Les cercles

Syntaxe

\pscircle[⟨*param.*⟩](x_0,y_0){⟨*rayon*⟩}

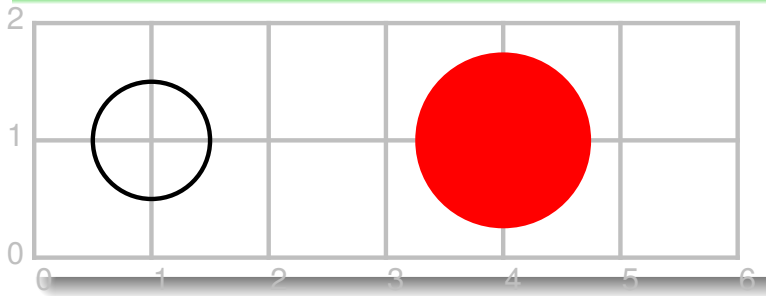
Les cercles

Exemple

Code source

```
1 \begin{pspicture}(6,2)\psgrid
2   \pscircle(1,1){.5}
3   \pscircle*[linecolor=red](4,1){.75}
4 \end{pspicture}
```

Résultat



Les ellipses

Syntaxe

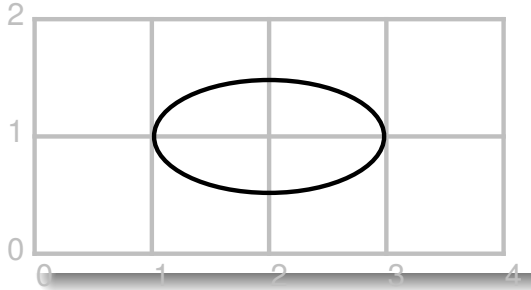
\psellipse[⟨param.⟩](x_0,y_0)(⟨ray. hor.⟩,⟨ray. vert.⟩)

Exemple

Code source

```
1 \begin{pspicture}(4,2)
2   \psgrid
3   \psellipse(2,1)(1,.5)
4 \end{pspicture}
```

Résultat



Boîtes encadrant du texte

Syntaxe (pour encadrer du texte)

```
\psframebox      % boîte rectang.
\psdblframebox   % boîte rectang. à double bordure
\psshadowbox     % boîte rectang. ombrée
\pscirclebox     % boîte circulaire
\psovalbox       % boîte ovale
\pstribox        % boîte triangulaire
\psdiabox        % boîte losange
```

Toutes ces ⟨*boîte*⟩s partagent la même syntaxe :

Syntaxe

\⟨*boîte*⟩[⟨*param.*⟩]{⟨*texte*⟩}

Le texte en boîte

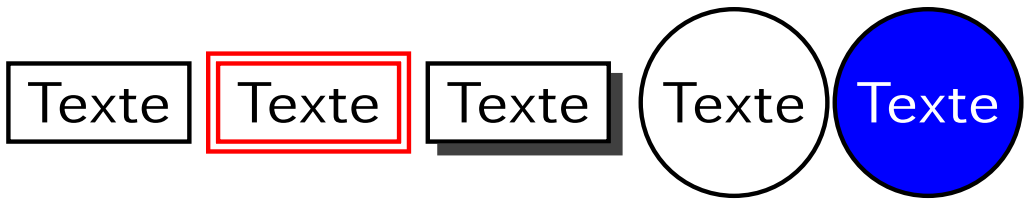
Suite

Exemple

Code source

```
1 \psframebox{Texte}
2 \psdblframebox[linecolor=red]{Texte}
3 \psshadowbox{Texte}
4 \pscirclebox[fillstyle=solid]{Texte}
5 \pscirclebox[fillstyle=solid,fillcolor=blue]{\white
      Texte}
```

Résultat



Le texte en boîte

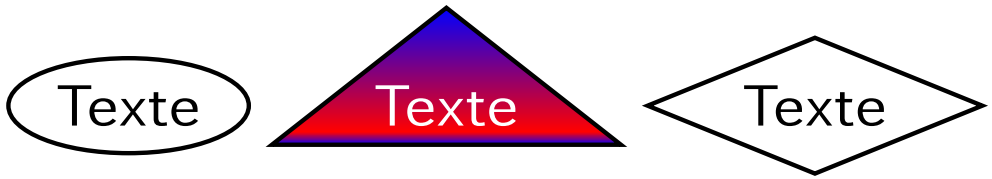
Suite

Exemple

Code source

```
1 \psovalbox{Texte}
2 \pstribox[fillstyle=gradient,gradbegin=blue,gradend=red
      ]{\white Texte}
3 \psdiabox{Texte}
```

Résultat



(Dé)Placer des objets

Syntaxe

\rput[⟨*point de référence*⟩]{⟨*angle*⟩}(x_0, y_0){⟨*objet*⟩}

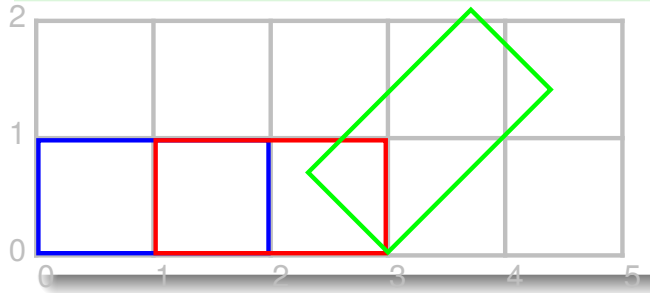
(Dé)Placer des objets

Exemple

Code source

```
1 \begin{pspicture}(5,2)\psgrid
2                   \psframe[linecolor=blue ](2,1)
3   \rput     (1,0){\psframe[linecolor=red  ](2,1)}
4   \rput{45}(3,0){\psframe[linecolor=green](2,1)}
5 \end{pspicture}
```

Résultat



(Dé)Placer des objets

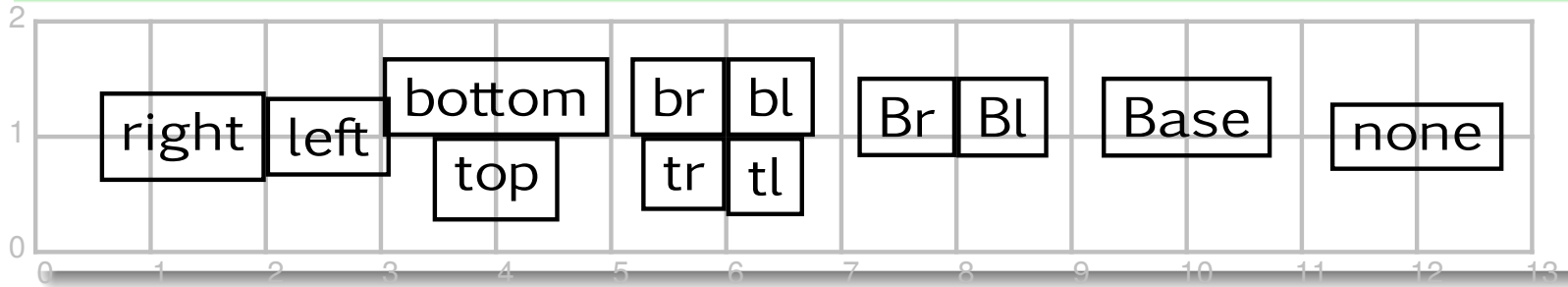
Points de référence

Exemple

Code source

```
1 \begin{pspicture}(13,2)\psgrid
2   \rput[l](2,1){\psframebox{left}}  \rput[r](2,1){\psframebox{right}}
3   \rput[t](4,1){\psframebox{top}}   \rput[b](4,1){\psframebox{bottom}}
4   \rput[tl](6,1){\psframebox{tl}}   \rput[tr](6,1){\psframebox{tr}}
5   \rput[bl](6,1){\psframebox{bl}}   \rput[br](6,1){\psframebox{br}}
6   \rput[Bl](8,1){\psframebox{Bl}}   \rput[Br](8,1){\psframebox{Br}}
7   \rput[B](10,1){\psframebox{Base}} \rput(12,1){\psframebox{none}}
8 \end{pspicture}
```

Résultat



Placer des labels

Syntaxe

Syntaxe

\uput{⟨*sép. label*⟩}[⟨*réf. angle*⟩]{⟨*angle*⟩}(x,y){⟨*texte*⟩}

Placer des labels

Exemple

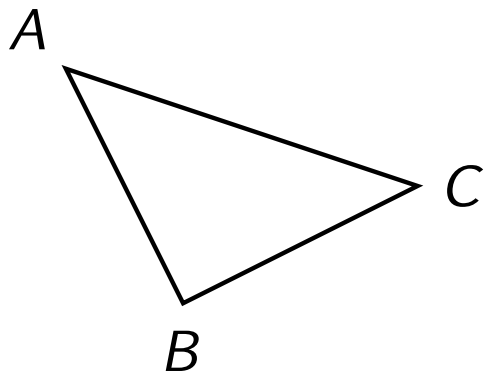
Exemple

Code source

```
1 Soit $ABC$ un triangle.
2
3 \begin{pspicture}(5,4)
4 \pspolygon(1,3)(2,1)(4,2)
5 \uput[ul](1,3){$A$}
6 \uput[d](2,1){$B$}
7 \uput[r](4,2){$C$}
8 \end{pspicture}
```

Résultat

Soit ABC un triangle.



Placer des labels

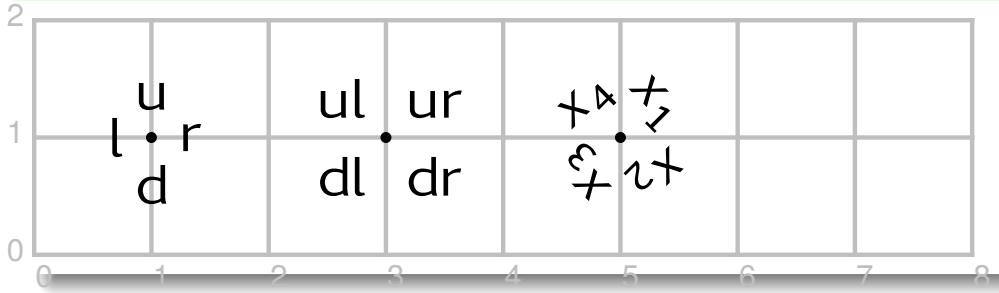
Exemples

Exemple

Code source

```
1 \begin{pspicture}(8,2)\psgrid
2 \qdisk(1,1){1pt}\uput[u](1,1){u}\uput[r](1,1){r}\uput[d](1,1){d}\uput[l](1,1){l}
3 \qdisk(3,1){1pt}
4 \uput[ur](3,1){ur}\uput[dr](3,1){dr}
5 \uput[dl](3,1){dl}\uput[ul](3,1){ul}
6 \qdisk(5,1){1pt}
7 \uput[ur]{-45}(5,1){$x_1$}\uput[dr]{135}(5,1){$x_3$}
8 \uput[dl]{-135}(5,1){$x_2$}\uput[ul]{45}(5,1){$x_4$}
9 \end{pspicture}
```

Résultat



Autres objets

- Bien d'autres objets sont disponibles
- On consultera
  - la documentation : *complète*
  - *une FAQ visuelle*
  - le site Web dédié à PSTricks

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

Les graphiques

À partir d'un fichier de données

Syntaxe

```
\readdata{⟨commande⟩}{⟨fichier de données⟩}
\begin{pspicture}(x0,y0)(x1,y1)
  \dataplot[⟨param.⟩]{⟨commande⟩}
\end{pspicture}
```

Les graphiques

À partir d'un fichier de données

Exemple (de fichier de données)

```
1 [
2 (2.1,1.4)(2.5,1.5)(2.7,1.8)(2.5,2.1)(1.9,2.2)
3 (1.3,2)(1.1,1.6)(1.6,1.2)(2.4,1.1)(3.2,1.4)
4 (3.3,2.1)(2.5,2.6)(1.3,2.6)(0.4,2)(0.4,1.1)
5 (1.7,0.5)(3.4,0.6)(4,1.6)(3.9,2.8)(2.1,3.5)
6 ]
```

Remarque

- Crochets fortement conseillés
- Crochet ouvrant : en début de ligne nécessairement
- Le fichier ne doit rien contenir d'autre [1]

1. Néanmoins admis : commentaires commençant par %

Les graphiques

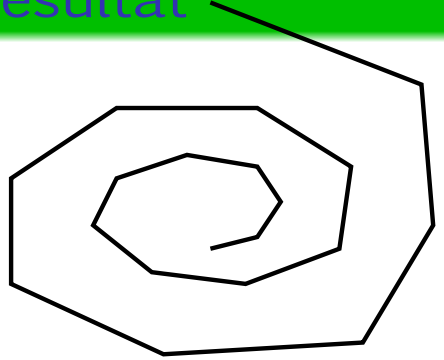
À partir d'un fichier de données : exemple

Exemple

Code source

```
1 \readdata{\donnees}{donnees.txt}
2 \begin{pspicture}(4,3)
3   \dataplot{\donnees}
4 \end{pspicture}
```

Résultat



Les graphiques

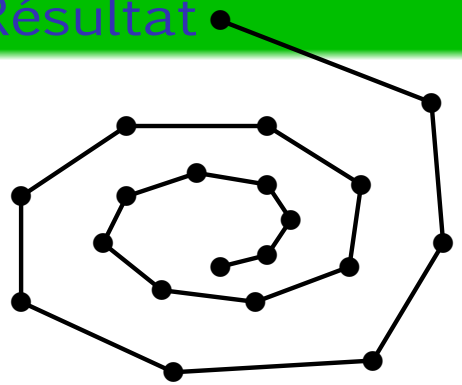
À partir d'un fichier de données : exemple

Exemple

Code source

```
1 \readdata{\donnees}{donnees.txt}
2 \begin{pspicture}(4,3)
3   \dataplot[showpoints=true]{\donnees}
4 \end{pspicture}
```

Résultat



Les graphiques

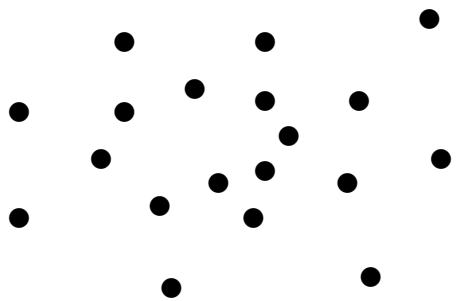
À partir d'un fichier de données : exemple

Exemple

Code source

```
1 \readdata{\donnees}{donnees.txt}
2 \begin{pspicture}(4,3)
3   \dataplot[plotstyle=dots]{\donnees}
4 \end{pspicture}
```

Résultat



Les graphiques

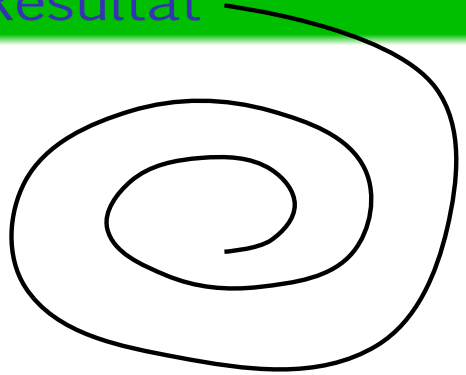
À partir d'un fichier de données : exemple

Exemple

Code source

```
1 \readdata{\donnees}{donnees.txt}
2 \begin{pspicture}(4,3)
3   \dataplot[plotstyle=curve]{\donnees}
4 \end{pspicture}
```

Résultat



Les graphiques

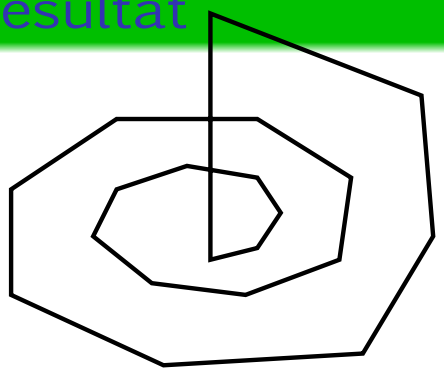
À partir d'un fichier de données : exemple

Exemple

Code source

```
1 \readdata{\donnees}{donnees.txt}
2 \begin{pspicture}(4,3)
3   \dataplot[plotstyle=polygon]{\donnees}
4 \end{pspicture}
```

Résultat



Les courbes

Fonctions

Syntaxe (fonction numérique d'une variable)

\psplot[⟨param.⟩]{⟨min. de x⟩}{⟨max. de x⟩}{⟨f(x)⟩}

Syntaxe (courbe paramétrée)

\parametricplot[⟨param.⟩]{⟨min. de t⟩}{⟨max. de t⟩}{⟨x(t)⟩}{⟨y(t)⟩}

Attention!

⟨f(x)⟩, ⟨x(t)⟩, ⟨y(t)⟩ : en notation polonaise inverse!

ulco Université Littoral Côte d'Opale

Les courbes : exemple

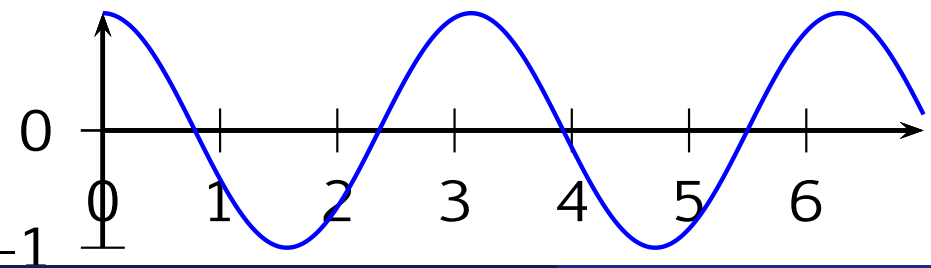
Courbe d'équation $y = \cos 2x$

Exemple

Code source

```
1 \psset{plotpoints=1000}
2 \begin{pspicture}(-0.5,-1.5)(8.5,1.5)
3   \psaxes{->}(0,0)(0,-1)(7,1)
4   \psplot[linecolor=blue]{0}{7}{
5     x 2 mul 180 mul 3.1415 div cos
6   }
7 \end{pspicture}
```

Résultat



Les courbes

Pour éviter la notation polonaise inverse !

Attention !

Notation polonaise inverse évitable :

- package `pstricks-add`
- option `algebraic`

Les courbes

Pour éviter la notation polonaise inverse : exemple

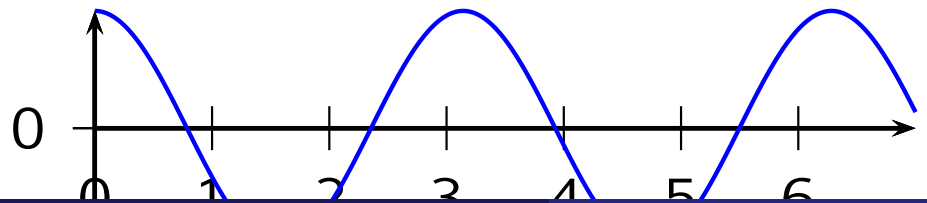
Exemple

Code source

```
\usepackage{pstricks-add}
```

```
1 \psset{plotpoints=1000,algebraic}
2 \begin{pspicture}(-0.5,-1.5)(8.5,1.5)
3   \psaxes{->}(0,0)(0,-1)(7,1)
4   \psplot[linecolor=blue]{0}{7}{cos(2*x)}
5 \end{pspicture}
```

Résultat



Les courbes

Package pstricks-add et courbes paramétrées

Attention !

Syntaxe des courbes paramétrées : ≠ avec pstricks-add

Syntaxe

```
\usepackage{pstricks-add}
```

```
\parametricplot[⟨param.⟩]{⟨min. de t⟩}{⟨max. de
    t⟩}{⟨x(t)⟩|⟨y(t)⟩}
```

Remarque

$$\{\langle x(t)\rangle|\langle y(t)\rangle\} \neq \{\langle x(t)\rangle\}\{\langle y(t)\rangle\}$$

Les courbes

Package pstricks-add et courbes paramétrées

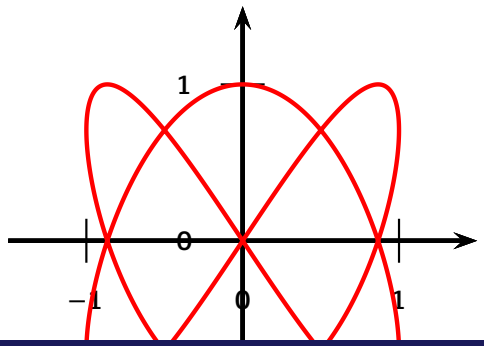
Exemple

Code source

```
\usepackage{pstricks-add}
\psset{algebraic=true,plotpoints=1000,unit=2cm}
```

```
1 \begin{pspicture}(-1.5,-1.5)(1.5,1.5)
2   \psaxes{->}(0,0)(-1.5,-1.5)(1.5,1.5)
3   \parametricplot[linecolor=red]{-3.14}{3.14}{sin(2*t)|sin(3*t)}
4 \end{pspicture}
```

Résultat



Package pstricks-add

Remarque

pstricks-add : étend les fonctionnalités de PSTricks [1]

1. Il règle aussi quelques bugs

Courbes de fonctions mathématiques

Package pst-func

Package pst-func [1] : courbes utiles en math., p. ex. :

- de Bézier d'ordres 1 à 9
- de polynômes et de leurs dérivées
- de Fourier
- de fonctions de Bessel
- des fonctions $x \mapsto \int_0^x \frac{\sin t}{t}\,\mathrm{d}t$ et $x \mapsto -\int_{-\infty}^x \frac{\cos t}{t}\,\mathrm{d}t$
- des fonctions Γ et $\ln\Gamma$
- des distributions statistiques les plus courantes [2]
- de fonctions implicites
- définies par l'utilisateur

1. Charge pstricks-add
2. Normales, binomiales, Poisson, Γ, du χ^2, de Student, F, β

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

Les nœuds

Liste non exhaustive

Syntaxe

```
\rnode[⟨point de référence⟩]{⟨nom⟩}{⟨objet⟩}
\circlenode[⟨param.⟩]{⟨nom⟩}{⟨objet⟩}
\dianode[⟨param.⟩]{⟨nom⟩}{⟨objet⟩}
\ovalnode[⟨param.⟩]{⟨nom⟩}{⟨objet⟩}
```

Les nœuds

Exemple

Exemple

Code source

```
1 \rnode{A}{\psframebox{$A$}}
2 \hspace{2cm}
3 \rnode{B}{\psframebox{$B$}}
```

Résultat

A B

Remarque

Absence de l'environnement `pspicture`

Connecteurs de nœuds

Liste non exhaustive

Syntaxe

```
\ncline    % segment de droite
\ncarc     % arc
\nccurve   % courbe de Bézier
\ncbar     % ligne brisée avec 2 bras parallèles,
           % de même « sens »
\ncangle   % ligne brisée avec 2 bras d'angles
           % à spécifier
\ncangles  % analogue à \ncangle mais avec
           % angle supplémentaire
\ncdiag    % ligne brisée avec bras d'angles et de
           % longueurs à spécifier
\ncloop    % « boucle »
```

Connecteurs de nœuds

Syntaxe

Tous ces connecteurs partagent la même syntaxe :

Syntaxe

\⟨*connecteur*⟩[⟨*param.*⟩]{⟨*flèches*⟩}{⟨*nœud*$_1$⟩}{⟨*nœud*$_2$⟩}

Connecteurs de nœuds

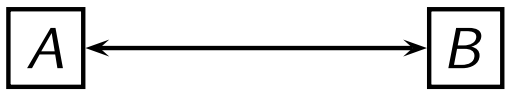
Exemple

Exemple

Code source

```
1 \rnode{A}{\psframebox{$A$}}
2 \hspace{2cm}
3 \rnode{B}{\psframebox{$B$}}
4 \ncline{<->}{A}{B}
```

Résultat



Connecteurs de nœuds

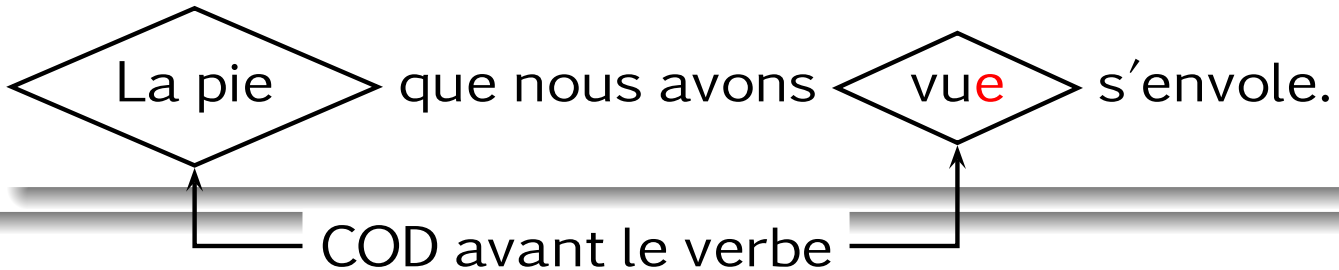
Exemple

Exemple

Code source

```
1 \dianode{A}{La pie} que nous avons
2 \dianode{B}{vu{\red e}} s'envole.
3 \ncbar[angle=-90,arm=.5cm]{<->}{A}{B}
4 \ncput*{COD avant le verbe}
```

Résultat



Connecteurs de nœuds

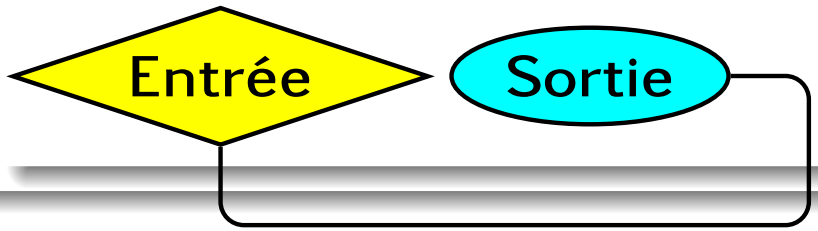
Exemple

Exemple

Code source

```
1 \dianode[fillstyle=solid,fillcolor=yellow]{A}{\textbf{Entrée}}
2 \ovalnode[fillstyle=solid,fillcolor=cyan]{B}{\textbf{Sortie}}
3 \ncangles[angleA=-90,armA=.5cm,armB=.5cm,linearc=0.2]{A}{B}
```

Résultat



Les nœuds

Dans du texte ordinaire

Remarque

Nœuds [1] : utilisables

- en dehors des environnements `pspicture`
- notamment dans des paragraphes ordinaires

1. De même que l'ensemble des commandes PSTricks

Ce que nous détaillons maintenant

Packages dérivés

Pour divers domaines

Nombreux packages dérivés, p. ex. pour :

- circuits électriques : `pst-circ`
- géométrie euclidienne : `pst-eucl`
- figures et graphiques 3D : `pst-3dplot`
- l'optique : `pst-optic`
- éléments de laboratoires de chimie : `pst-labo`
- etc. [1]

1. Liste consultable ici et ici

Bibliothèques

Circuits électriques : exemples (bipôles)

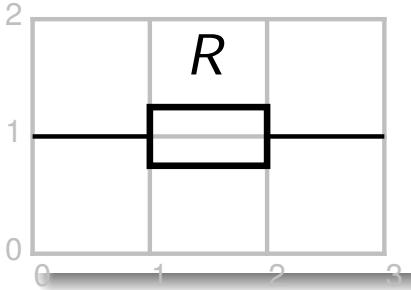
Exemple

Code source

```
\usepackage{pst-circ}
```

```
1 \begin{pspicture}(3,2)
2   \psgrid
3   \pnode(0,1){A}
4   \pnode(3,1){B}
5   \resistor(A)(B){$R$}
6 \end{pspicture}
```

Résultat



Bibliothèques

Circuits électriques : exemples (bipôles)

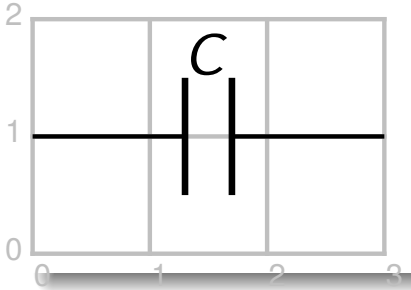
Exemple

Code source

```
\usepackage{pst-circ}
```

```
1 \begin{pspicture}(3,2)
2   \psgrid
3   \pnode(0,1){A}
4   \pnode(3,1){B}
5   \capacitor(A)(B){$C$}
6 \end{pspicture}
```

Résultat



Bibliothèques

Circuits électriques : exemples (bipôles)

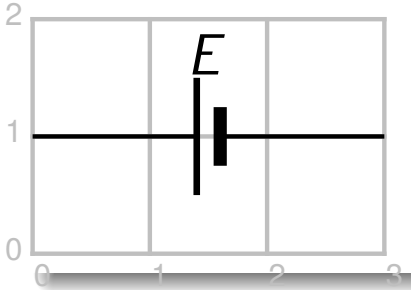
Exemple

Code source

```
\usepackage{pst-circ}
```

```
1 \begin{pspicture}(3,2)
2   \psgrid
3   \pnode(0,1){A}
4   \pnode(3,1){B}
5   \battery(A)(B){$E$}
6 \end{pspicture}
```

Résultat



Bibliothèques

Circuits électriques : exemples (tripôles)

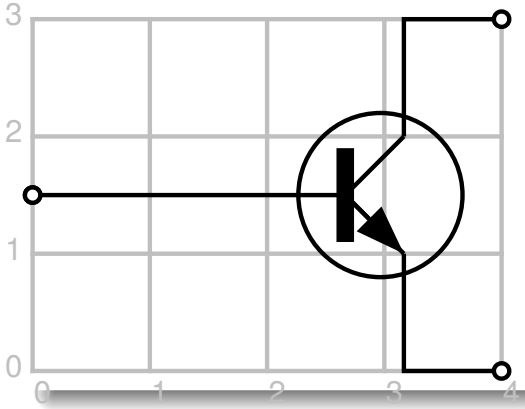
Exemple

Code source

```
\usepackage{pst-circ}
```

```
1 \begin{pspicture}(4,3)
2   \psgrid
3   \pnode(0,1.5){A}
4   \pnode(4,0){B}
5   \pnode(4,3){C}
6   \transistor[basesep=2cm,
      arrows=o-o](A)(B)(C)
7 \end{pspicture}
```

Résultat



Bibliothèques

Circuits électriques : exemples (quadripôles)

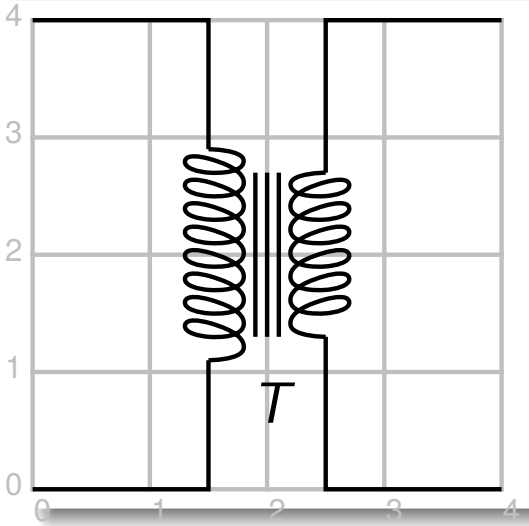
Exemple

Code source

```
\usepackage{pst-circ}
```

```
\begin{pspicture}(4,4)
  \psgrid
  \pnode(0,4){A}
  \pnode(0,0){B}
  \pnode(4,4){C}
  \pnode(4,0){D}
  \transformer(A)(B)(C)(D){$T$
    }
\end{pspicture}
```

Résultat



Bibliothèques

Circuits électriques : exemples

Bibliothèques – suite

Circuits électriques : exemples

Exemple

Code source

```
\usepackage{pst-circ}
```

```
\begin{pspicture}(-1.5,-1)(6.5,5)
\psset{
  intensitycolor=red,
  intensitylabelcolor=red,
  intensitywidth=3pt,
  tensionlabelcolor=blue,
  tensioncolor=blue
}
\psgrid[subgriddiv=1,griddots=10]
\pnode(0,0){A}\pnode(0,3){B}\pnode(4.5,3){C}\pnode(4.5,0){D}
\Ucc[tension,dipoleconvention=generator](A)(B){$E$}
\multidipole(B)(C)
\switch[intensitylabel=$j$]{$K$}
\resistor[labeloffset=0,tensionlabel=$u_R$]{$R$}.
\capacitor[tensionlabel={$u_C$},tensionlabeloffset=-1.4,tensionoffset=-1,directconvention=false](D)(
      C){$C$}
\wire(A)(D)\ground(D)
\end{pspicture}
```

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

Interfaces graphiques

- JasTeX
- TeXgraph

Ce que nous détaillons maintenant

ulco Université Littoral Côte d'Opale

TikZ/PGF

- Développé par le concepteur de Beamer
- Par rapport à PSTricks :

  caractéristiques analogues :
  - code directement dans le fichier `.tex`
  - ~~nécessite~~ une compilation par un programme tiers
  - utilisable avec LuaLaTeX ou XeLaTeX

  caractéristiques différentes : utilisable avec pdfLaTeX

Attention !

Documentation pléthorique (1321 pages)

PGF

Packages dérivés pour divers domaines

Nombreux packages dérivés, p. ex. pour :

- circuits électriques : `circuitikz`
- géométrie euclidienne : `tkz-euclide`
- figures et graphiques 2D et 3D : `pgfplots` [1]
- représentations graph. de fonctions en 2D : `tkz-fct`
- tableaux de signes et de variations : `tkz-tab`
- visualisation de réseaux : `tikz-network`
- etc. [2]

1. Cf. p. ex. https://dgxy.link/en-ligne10
2. Liste consultable ici

METAPOST

Remarque

Outil nécessitant :

- apprentissage d'un autre langage informatique
- code créé dans un fichier `.mp` autre que le source `.tex` [1]
- compilation `mpost` du fichier `.mp`
- inclusion du graphique généré dans le `.tex`

Remarque

Malgré ces inconvénients, METAPOST est très puissant!

1. Sauf si recours aux packages :
   - `gmp` avec pdfLaTeX
   - `luamplib` avec LuaLaTeX (intégration alors transparente)

METAPOST

On consultera :

- des exemples :
  - nombreux et variés
  - sur le site Syracuse
  - d'animations
-
  - une page dédiée
  - un manuel en français

Asymptote

- Présenté comme un successeur de METAPOST
- Avantages et inconvénients analogues à METAPOST

On consultera

- *la documentation*
- des exemples :
  - généraux ou plus spécifiquement :
    - de graphiques 2D
    - de graphiques 3D
    - d'animations
  - sur les pages de Gaëtan Marris