



Cours L^AT_EX EDITE de Paris

Bibliographies, commandes, packages



Bibliographies

BibT_EX

Citations

Commandes

Packages et classes

Application



Bibliographies

BibT_EX

Citations

Commandes

Packages et classes

Application



- Logiciel de gestion de **bibliographies**
- Document BibTeX : fichier texte **.bib** décrivant des références bibliographiques, avec une syntaxe spéciale.
- Chaque référence est identifiée par une **clef** alphanumérique.
- Principe général : on compile une fois avec **pdflatex** pour récolter les citations bibliographiques, on appelle **bibtex** sur le nom du document LaTeX sans le **.tex**, puis on recompile deux fois avec **pdflatex** pour intégrer les références bibliographiques.
- Vieux logiciel : parfois des petits côtés pénibles pour le faire marcher. En particulier, il faut parfois encoder les accents « à l'ancienne » (p. ex., `\'e` pour « é »), protéger les macros avec des accolades. . .





Fichiers auxiliaires

- À partir des documents (`.tex`), packages (`.sty`), classe (`.cls`), etc., \LaTeX produit le document final (`.pdf` ou `.dvi`) ainsi que des fichiers auxiliaires :
 - `.aux` toutes les références rencontrées dans le fichier
 - `.toc` les titres qui constitueront la table des matières
 - `.out` les bookmarks générés par hyperref
 - `.log` un log de compilation
 - etc.
- À partir du `.aux`, du fichier de références (`.bib`), du style de bibliographie (`.bst`), Bib \TeX produit un fichier `.bb1` qui contient la bibliographie formatée.
- Si un tel fichier `.bb1` existe, il est utilisé pour produire le document final à la prochaine compilation.





Exemple de référence BibT_EX

```
@article{ bryant92symbolic,  
  author = "Randal E. Bryant",  
  title = "Symbolic {Boolean} Manipulation with  
Ordered Binary-Decision Diagrams",  
  journal = {ACM Computing Surveys},  
  volume = 24,  
  number = 3,  
  pages = {293-318},  
  year = 1992}
```

Un fichier BibT_EX est une suite de telles références.

bryant92symbolic est la clef, les autres lignes décrivent les champs ;
la valeur d'un champ complexe est mis entre guillemets droites ou
entre accolades.





Types de références les plus courants

@article article de journal

@book livre

@incollection chapitre de livre

@inproceedings article publié dans les actes d'une conférence

@mastersthesis thèse de Master

@misc divers

@phdthesis thèse de doctorat

@techreport rapport technique

@unpublished non publié

16/02/2010





Champs existants

author auteurs du documents, doivent être séparés par des « and » (p. ex., "Jean Dupont and Jacques Durand" ou "Dupont, Jean and Durand, Jacques")

title titre du document ; si celui-ci contient des majuscules qui doivent impérativement être présente (acronyme, nom propre), doivent être protégées par des accolades

month mois, sous forme d'abréviation : jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov, dec

institution, school pour les rapports techniques ou les thèses

booktitle pour les articles de conférences et les chapitres de livre

editor pour l'éditeur d'un ouvrage collectif

publisher pour l'éditeur d'un livre

address pour l'adresse de l'éditeur

volume, number, pages, year, note

16/02/2010



Bibliographies

BibT_EX

Citations

Commandes

Packages et classes

Application



```
D'après~\cite{bryant92symbolic},  
les OBDD sont très intéressants.
```

```
...
```

```
\bibliographystyle{alpha}  
\bibliography{biblio}
```

- `\cite{clef}` pour citer une référence, `\nocite{clef}` pour inclure une référence dans la bibliographie sans la citer, `\nocite{*}` pour inclure toutes les références
- Styles de bibliographie standard : alpha, plain, unsrt
- **biblio** désigne le nom du fichier BibT_EX (sans l'extension)





Personnaliser les références

- Nombreux styles de références, y compris des styles français (unsrt-fr, plain-fr, alpha-fr, etc.)
- packages pour personnaliser les citations dans le corps du document : **natbib**, **jurabib**
- packages pour avoir des bibliographies multiples : **bibtopic**, **multibib**, **splitbib**



Bibliographies

Commandes

Définir des commandes \LaTeX

Définir des commandes en \TeX

Packages et classes

Application



Bibliographies

Commandes

Définir des commandes \LaTeX

Définir des commandes en \TeX

Packages et classes

Application





Commandes sans arguments

```
\newcommand{\hello}  
{Bonjour, ça va?\par}  
  
\hello\hello\hello
```

Bonjour, ça va ?
Bonjour, ça va ?
Bonjour, ça va ?





Redéfinir une commande

```
\[\epsilon\leq\phi\geq\emptyset\  
  
\renewcommand{\epsilon}{\varepsilon}  
\renewcommand{\phi}{\varphi}  
\renewcommand{\emptyset}{\varnothing}  
\renewcommand{\leq}{\leqslant}  
\renewcommand{\geq}{\geqslant}  
  
\[\epsilon\leq\phi\geq\emptyset\  
]
```

$$\epsilon \leq \phi \geq \emptyset$$

$$\varepsilon \leqslant \varphi \geqslant \varnothing$$





Commandes avec arguments

```
\newcommand{\textitbf}[1]  
{\itshape\bfseries #1}
```

C'est \textitbf{gras
italique}.

C'est ***gras italique***.

Il y a aussi `\newcommand*` qui impose que son argument ne comporte pas de saut de paragraphe.





Commandes avec argument optionnel

```
\newcommand{\monsieur}[2] [M.]  
{#1~\textsc{#2}}
```

```
\monsieur[Jean]{Dupont} et  
\monsieur{Durand} sont venus.
```

Jean DUPONT et M. DURAND
sont venus.

On donne la valeur par défaut de l'argument optionnel (il ne peut y en avoir qu'un) après le nombre d'arguments.



Environnements

```
\newenvironment{listeunel}  
{\begin{itemize}\item }  
{\end{itemize}}
```

```
\begin{listeunel}  
toto  
\end{listeunel}
```

■ toto

On peut aussi définir des environnements avec arguments, avec argument optionnel, redéfinir des environnements, etc.



Bibliographies

Commandes

Définir des commandes \LaTeX

Définir des commandes en \TeX

Packages et classes

Application



```
\let\t=\textbf
\renewcommand{\textbf}[1]
{\textit{#1}}

\t{toto} \t{titi}

\textbf{toto} \textbf{titi}
```

toto titi
toto titi



Définition de macro T_EX

```
\def\bouh#1(#2)+#3  
{\textbf{#1}\textit{#2}%  
\textsc{#3}}
```

totiTA

```
\bouh to(ti)+ta
```

Définition d'une macro avec syntaxe (à peu près) arbitraire !



Bibliographies

Commandes

Packages et classes

Créer un package

Créer une classe

Installation

Application



Bibliographies

Commandes

Packages et classes

Créer un package

Créer une classe

Installation

Application





Principes généraux

- Package : fichier \LaTeX avec extension **.sty**
- Dans un package, on utilise `\RequirePackage` plutôt que `\usepackage`
- Seule différence importante : dans un package, des noms de commandes peuvent contenir un caractère `@`, pas dans un document. On réserve donc les noms de commande avec `@` à des commandes internes.
- Au cas où on veut quand même utiliser une commande avec `@` dans un document on peut utiliser `\makeatletter` avant et `\makeatother`
- Possibilité de gérer des options de package





Squelette d'un package

```
\NeedsTeXFormat{LaTeX2e}

\ProvidesPackage{monpackage} % Nom du package

\DeclareOption*{
  \PackageError{monpackage}{Unknown option '\CurrentOption'}{
}
\DeclareOption{optiona}{codea}
\DeclareOption{optionb}{codeb}
\ProcessOptions

% Code arbitraire
```

codea est exécuté si l'option optiona est fournie.





Faciliter le traitement des options

```
\newif\if@optiona
\DeclareOption{optiona}{\@optionatrue}

\ProcessOptions

\if@optiona
...
\else
...
\fi
```



Bibliographies

Commandes

Packages et classes

Créer un package

Créer une classe

Installation

Application





Classe de document

- Très similaire à un package, fichier **.cls**
- En général, construit à partir d'une classe de base
- Squelette :

```
\NeedsTeXFormat{LaTeX2e}  
\ProvidesClass{classe} % Nom de la classe  
  
% On peut passer des options à la classe de base  
\PassOptionsToClass{options}{classebase}  
  
% On peut traiter des options comme pour les  
% packages avec \DeclareOption, \ProcessOptions  
  
% On charge la classe de base  
\LoadClass{classebase}  
  
% Autres instructions
```



Bibliographies

Commandes

Packages et classes

Créer un package

Créer une classe

Installation

Application





Installation d'un package ou d'une classe

- Si c'est juste un fichier : le copier dans le répertoire de compilation
- Sinon, plus complexe. Sous Unix :
 - Créer un répertoire `$HOME/texmf/tex/latex/`
 - Y placer l'ensemble des fichiers du package (décompressés), dans un sous-répertoire au nom du package
 - Taper `texhash` en ligne de commande
 - Tester !



Bibliographies

Commandes

Packages et classes

Application





Application

Suivre les instructions de la feuille de TP disponible sur la page du cours.





Licence de droits d'usage



Contexte public } avec modifications

Par le téléchargement ou la consultation de ce document, l'utilisateur accepte la licence d'utilisation qui y est attachée, telle que détaillée dans les dispositions suivantes, et s'engage à la respecter intégralement.

La licence confère à l'utilisateur un droit d'usage sur le document consulté ou téléchargé, totalement ou en partie, dans les conditions définies ci-après et à l'exclusion expresse de toute utilisation commerciale.

Le droit d'usage défini par la licence autorise un usage à destination de tout public qui comprend :

- le droit de reproduire tout ou partie du document sur support informatique ou papier,
- le droit de diffuser tout ou partie du document au public sur support papier ou informatique, y compris par la mise à la disposition du public sur un réseau numérique,
- le droit de modifier la forme ou la présentation du document,
- le droit d'intégrer tout ou partie du document dans un document composite et de le diffuser dans ce nouveau document, à condition que :
 - L'auteur soit informé.

Les mentions relatives à la source du document et/ou à son auteur doivent être conservées dans leur intégralité.

Le droit d'usage défini par la licence est personnel et non exclusif.

Tout autre usage que ceux prévus par la licence est soumis à autorisation préalable et expresse de l'auteur : sitopedago@telecom-paristech.fr

16/02/2010

