

Utiliser Bib_TE_X

Rémi Peyre

2 septembre 2007

Résumé

Le présent document vise à regrouper au sein d'un même tutoriel l'ensemble des connaissances requises pour une utilisation sommaire de Bib_TE_X. L'essentiel de ce document est adapté de [1] ; j'ai également utilisé des éléments trouvés dans les autres références de la bibliographie (réalisée avec Bib_TE_X ☺).

Table des matières

1	Présentation	2
1.1	Qu'est-ce que Bib _T E _X ?	2
1.2	Comment ça marche ?	2
2	Le fichier .bib	3
2.1	Principe général	3
2.2	Types de sources	3
2.3	Types de champs	4
2.4	Remplir le champ <code>author</code>	5
3	Plus loin dans la syntaxe	6
3.1	L'entrée <code>@string</code>	6
3.2	Protection des majuscules	7
3.3	Faire des commentaires	7
4	Utiliser de Bib_TE_X dans un document	8
4.1	Faire appel au fichier .bib	8

4.2	Citer les références	8
4.3	Compiler	9
4.4	Et le français ?	9

1 Présentation

1.1 Qu'est-ce que BibTeX ?

BibTeX est un logiciel et un format de fichier conçu par Oren Patashnik et Leslie Lamport en 1985 pour L^AT_EX. Il sert à gérer et traiter des bases bibliographiques. En effet, lorsqu'on rédige un grand nombre de documents avec L^AT_EX, la rédaction des bibliographies peut vite apparaître fastidieuse quand on doit, pour chaque article, reporter « à la main » dans le fichier `.tex` ses références, et ce d'autant plus que les mêmes articles reviennent souvent dans différents document... Avec BibTeX au contraire, une seule base de données, que le rédacteur peut enrichir au fur et à mesure, est utilisée pour tous les documents, d'où gain de temps. Qui plus est, BibTeX inclut les standards de la typographie professionnelle concernant les bibliographies, ce qui produit ainsi un résultat plus soigné. C'est aussi le meilleur moyen de ne citer que les références effectivement mentionnées dans l'article, ce qui peut être utile lorsqu'on retouche son document en y ajoutant ou supprimant des références ⁽¹⁾.

1.2 Comment ça marche ?

BibTeX est, pour parler de façon générale, un utilitaire qui gère une base de données : il sait extraire, trier et mettre en page des éléments d'une base de données, afin d'en faire un résultat compilable par L^AT_EX. Évidemment, dit comme ça, c'est le rêve. La réalité est légèrement plus complexe : pour que BibTeX sache quelles entrées il doit extraire de la base de données, il faut le lui dire. Il faut donc passer un coup de L^AT_EX avant. Et forcément un coup après, pour que L^AT_EX prenne en compte le résultat. Voici comment ça marche de façon précise :

- La première étape est donc la compilation du document L^AT_EX. À ce moment, si rien n'a été extrait de la base de données, les références ne sont pas indiquées, bien entendu. Par contre, à chaque fois qu'une référence bibliographique est utilisée, c'est indiqué dans le fichier `.aux`. Au cours de cette compilation, L^AT_EX va également écrire dans le fichier `.aux` quelle base de données est utilisée, et quel « style » il faut appliquer pour la mise en page de ses éléments ;
- On lance alors BibTeX, qui va regarder dans le fichier `.aux` les informations qui le concernent : le style (fichier d'extension `.bst`) qu'il doit utiliser, la ou les bases de données (dont l'extension est `.bib`) d'où il doit extraire les renseignements bibliographiques, ainsi que les éléments à extraire. Il va ensuite faire ce qu'on lui a demandé,

¹Si on n'utilise pas BibTeX, on sait quand même s'il manque un `\bibitem` ou s'il y a des `\bibitem` de trop !

c'est à dire l'extraction et la mise en forme des données, et mettre le tout dans un fichier d'extension `.bbl`. Les messages d'avertissements et d'erreur sont mis dans un fichier d'extension `.blg`;

- Afin de prendre en compte ces données, on relance \LaTeX sur le document. Le fichier `.bbl` qui vient d'être créé est inclus. Les `\bibitem` sont donc exécutés, donc \LaTeX définit les labels à utiliser dans le fichier `.aux`. Cependant, il les écrit, mais il ne les crée pas pour l'exécution actuelle (il n'a aucun moyen de les connaître au début de l'exécution), les références ne seront donc pas bonnes après cette étape ;
- Il faut donc un dernier passage par \LaTeX , qui permet d'avoir les bonnes références dans l'ensemble du document.

Il faut donc une compilation de \LaTeX , une exécution de \BibTeX et deux compilation de \LaTeX pour que le document soit correct.

2 Le fichier `.bib`

2.1 Principe général

C'est la base de données bibliographiques, celle qui contient les références de tous les travaux que vous citerez. Le fichier `.bib` est un fichier de texte simple, ce qui signifie que vous allez le créer puis le modifier avec votre éditeur de texte favori ; il faut juste prendre garde à lui donner la bonne extension.

Le fichier `.bib` contient une description des ouvrages sous la forme « mot-clef = valeur », par exemple :

```
@book{Ben62,  
  title = "L'Oxydation des métaux",  
  author = "J. Bénard and J. Bardolle and F. Bouillon and M. Cagnet  
    and J. Moreau and G. Valensi",  
  publisher = "Gauthier-Villars",  
  year = "1962"  
}
```

indique qu'il s'agit d'un livre (`@book`), que sa référence dans la source est `Ben62`, que son titre est *L'Oxydation des métaux*, ses auteurs sont J. Bénard *et al.*, et qu'il a été édité chez Gauthier-Villars en 1962.

2.2 Types de sources

Voici la liste des principaux types de sources connues par \BibTeX . Pour chacun de ces types, j'ai indiqué les champs qui doivent *obligatoirement* être renseignés. Ce qui n'empêche pas de renseigner d'autres champs ; au pire, \BibTeX les ignorera si il estime qu'ils sont superflus pour le type de source indiqué.

Type de source	Documents concernés	Champs obligatoires
@article	Article paru dans un journal.	author, title, year, journal
@book	Livre.	author <u>ou</u> editor, title, publisher, year
@booklet	Petit livre, qui n'a pas champ publisher.	title
@inbook	Partie (un chapitre, souvent) d'un livre.	author <u>ou</u> editor, title, chapter <u>ou</u> pages
@incollection	Grosse partie d'un livre. Pas juste un petit chapitre... Il faut en particulier que cette partie ait son propre titre.	author, title, booktitle, publisher, year
@inproceedings	Article dans les actes d'une conférence, d'un colloque, d'une rencontre...	author, title, booktitle, year
@manual	Une petite doc, comme celle-ci, par exemple.	title
@masterthesis	Mémoire de D.E.A., ou équivalent.	author, title, school, year
@misc	Quand on ne sait pas quoi mettre, on met @misc...	author <u>ou</u> title <u>ou</u> howpublished <u>ou</u> year <u>ou</u> note
@phdthesis	Thèse de doctorat, d'habilitation, ou un autre gros truc dans le même genre.	author, title, school, year
@proceedings	Actes d'une conférence.	title, year
@techreport	Rapport technique, publié par un labo, un centre de recherche...	author, title, institution, year
@unpublished	Document non publié. C'est souvent assez proche de l'entrée @misc, sauf que là, il y a un auteur et un titre.	author, title, note

2.3 Types de champs

Et maintenant évidemment, la liste des champs les plus courants :

Nom	Contenu
author	Les noms des auteurs.
booktitle	Comme son nom l'indique, c'est le titre d'un livre. C'est utilisé dans le cas où on cite une partie dudit livre.
chapter	Le numéro du chapitre (ou n'importe quelle type de sectionnement) que l'on cite dans un livre. Si ce n'est pas un chapitre, le champ type permet de spécifier le type de sectionnement.
edition	Indique le numéro de l'édition, ou plutôt son ordinal. On mettra par exemple <code>edition = "Premi\`ere"</code> .
editor	Indique le nom du rédacteur en chef, et non pas de l'éditeur comme on pourrait le croire. C'est juste un faux-ami de la langue anglaise, mais « editor » se traduit bien par « rédacteur en chef ».
howpublished	Ne sert que dans les cas particuliers où le document cité n'est pas d'un type classique, comme un livre, un article de journal ou de conférence, ...
institution	Dans le cas d'un rapport, indique l'institution qui l'a publié.
journal	Le nom du journal dont est extrait un article.
note	Une remarque additionnelle quelconque.
number	Un numéro... Mais pas un numéro quelconque : ça peut être le numéro dans une série, ou le numéro d'un rapport. Pour les numéros de volume, on utilisera le champ volume.
pages	Les pages qui nous intéressent. Ça permet d'éviter au lecteur de se faire tout un bouquin pour trouver l'information à laquelle on fait référence. Cette information peut aussi être passée en argument optionnel de la commande <code>\cite</code> , auquel cas elle apparaîtra dans le texte, à l'endroit de la citation, mais pas dans la liste des références.
publisher	Pour le nom de la maison d'édition ou de l'organisme qui a publié le document cité.
school	Pour un mémoire ou une thèse, l'école où il a été réalisé.
series	Le nom d'une série d'ouvrages, d'une collection de bouquins, ...
title	Le titre du document que l'on cite.
type	Le type. Le type de quoi ? Ça dépend : le type de publication, au cas où c'est pas clair. Par exemple, on a vu que pour une thèse de doctorat ou une thèse d'habilitation, il n'y a qu'un type d'entrée possible. C'est dans le champ type que l'on peut apporter la nuance supplémentaire. Mais ça peut aussi être le type de sectionnement utilisé dans l'argument du champ chapter, si l'entrée est <code>@inbook</code> ou <code>@incollection</code> .
volume	Le numéro de volume dans une série ou dans une collection.
year	L'année de publication du document cité.

2.4 Remplir le champ author

Imaginons que nous voulons référencer le *L^AT_EX Companion*, écrit par Michel Goossens, Franck Mittelbach et Alexander Samarin. On rentrera alors dans le fichier `.bib` :

```
author = "Goossens, Michel and Mittelbach, Franck and Samarin, Alexander"
```

La première chose à constater, c'est que deux auteurs sont séparés par le mot-clef `and`. Ensuite, pour un auteur, on met d'abord son nom, une virgule, et son prénom. Enfin, ça, c'est dans le cas simple. . . Avant d'aller plus loin, remarquons un point crucial : Bib_{TEX} doit pouvoir distinguer le nom du prénom. Pour le tri, tout d'abord, et pour pouvoir, comme le font certains styles, abréger le prénom. Un autre point crucial est que certaines personnes ont plusieurs prénoms, ou des prénoms composés, d'autres ont plusieurs noms, d'autres ont une particule, et d'autres ont une « extension » (comme le « Fils » dans « Alexandre Dumas, Fils »). Dans l'explication un peu technique qui suit, le prénom sera appelé *First*, la particule sera notée *von*, le nom *Last* et l'extension, *Jr*.

Il faut donc que Bib_{TEX} puisse distinguer les différentes parties d'un nom. Dans ce but, Bib_{TEX} reconnaît trois formats pour un nom :

- `First von Last`;
- `von Last, First`;
- `von Last, Jr, First`.

Il lui suffit de compter le nombre de virgules pour savoir dans quelle situation il se trouve.

Remarque : comment faire pour que « Philippe » soit abrégé en « Ph. » ?

La fonction qui abrège les prénoms est en fait la même que celle qui extrait les parties *First*, *von*, *Last* et *Jr* des noms. Lorsqu'on lui demande d'abrégé un prénom, il va garder uniquement le premier caractère (qui peut être un « caractère spécial »). Donc, bien entendu, « Philippe » sera abrégé en « P. », ce que certains n'aiment pas trop. Pour contourner ce problème, on va utiliser un caractère spécial : on écrira donc « `{\relax Ph}ilippe` ». Ainsi, la version abrégée sera « `{\relax Ph}`. », ce qui donnera, en L^AT_EX, « Ph. ». La version non abrégée sera « `{\relax Ph}ilippe` », qui donne bien « Philippe ».

3 Plus loin dans la syntaxe

3.1 L'entrée `@string`

Ce n'est pas à proprement parler une entrée, mais ça y ressemble. L'entrée `@string` permet de définir des abréviations. Imaginons une bibliographie où on cite deux livres publiés par Addison-Wesley. Il pourrait être utile de définir un raccourci pour cette maison d'édition. On modifie donc notre fichier `.bib` de la façon suivante :

```
@string{AW = "Addison-Wesley"}

@book{companion,
  title = "The {\LaTeX} {C}ompanion",
  publisher = AW,
  ...
}
```

☛ On remarquera qu'on ne met pas de guillemets autour d'un élément défini dans un `@string`.

Cela permet non seulement de gagner un peu de temps, mais surtout d'homogénéiser la base de données, notamment ici, par exemple, de mettre le tiret partout ou nulle part, mais pas juste une fois de temps en temps... C'est aussi pratique pour les noms d'auteurs ; ça évite, en plus, de faire des fautes dans les noms, ce qui ne fait jamais très sérieux.

Pour concaténer des chaînes de caractères, il faut utiliser l'opérateur `#`. Par exemple, avec des entrées `@string` convenables, on indiquera les auteurs du *LaTeX Companion* par :

```
author = goossens # and "# mittelbach #" and "# samarin
```

3.2 Protection des majuscules

Lorsque le style de la bibliographie convertit les lettres majuscules d'un titre en lettres minuscules, vous pouvez le forcer à garder les majuscules en protégeant le titre par des accolades. Par exemple :

```
title = "Pattern Classification and Scene Analysis"
```

doit être remplacé par :

```
title = "{Pattern Classification and Scene Analysis}"
```

Vous pouvez aussi ne protéger que les lettres qui vous intéressent, en rajoutant des accolades uniquement autour de ces lettres.

Cette technique de protection entre accolades est générale et peut être utilisée dès qu'on veut imposer à BibTeX un rendu particulier (nom d'un auteur...), ou pour employer le symbole `@` sans qu'il soit interprété.

3.3 Faire des commentaires

Tout ce qui n'est pas à l'intérieur de la définition d'une entrée est considéré comme étant du commentaire, et BibTeX l'ignore. Comme une nouvelle entrée doit toujours commencer par un `@`, cela donne un moyen assez simple de « commenter » une entrée : il suffit de

supprimer son @ initial. Il existe un type d'entrée spécial, @comment, dont le contenu n'est pas interprété non plus par BibTeX. Une utilisation en est de supprimer facilement un grand nombre d'entrées de manière temporaire.

4 Utiliser de BibTeX dans un document

4.1 Faire appel au fichier .bib

Jusque-là, j'ai expliqué plein de choses sur BibTeX, mais tant qu'on ne saura pas ce qu'il faut écrire concrètement dans son fichier .tex, c'est un peu inutile...

Ici nous supposerons que je suis en train d'écrire un article `article.tex`, et que les données bibliographiques que je veux extraire sont contenues dans le fichier `bibliographie.bib`, situé dans le répertoire courant. Le fichier `article.tex` est lu par L^AT_EX, il doit contenir la séquence :

```
\bibliography{bibliographie}
\bibliographystyle{machin}
```

(nous expliquerons un peu plus loin le rôle de la commande `\bibliographystyle`). On remarquera que l'extension du fichier `bibliographie.bib` n'a pas été indiquée. Par ailleurs, l'argument de la commande `\bibliography` est en fait un chemin d'accès ; si mon fichier `bibliographie.bib` avait été situé dans le répertoire parent — ce qui est généralement le cas quand une même base de données sert pour plusieurs documents à la fois —, j'aurais écrit :

```
\bibliography{../bibliographie}.
```

La commande `bibliographystyle` indique à BibTeX comment il va devoir mettre en forme la bibliographie. Voici les quatre styles les plus courants :

- `plain` : Les entrées sont triées alphabétiquement et sont labélisées avec des nombres.
- `unsrt` : Idem à `plain`, excepté que les entrées apparaissent dans leur ordre de citation.
- `alpha` : Idem à `plain`, excepté que les labels sont de la forme [Knu66], à partir du nom de l'auteur et de l'année de publication.
- `abbrv` : Plus compact que `plain`.

4.2 Citer les références

On cite une référence comme d'habitude, par la commande `\cite{clef}`, où `clef` est la désignation de la référence dans le fichier .bib (i.e., la séquence de caractère située entre l'accolade ouvrante et la première virgule dans la description de la référence pour le fichier .bib — voir § 2.1). Pour ajouter une référence non citée dans le texte, il faut utiliser `\nocite{clef}` avant `\bibliography{bibliographie}`, ou bien `\nocite{*}` pour citer tout le contenu du fichier `bibliographie.bib`.

4.3 Compiler

Le principe a été expliqué au § 1.2. Dans la pratique, on entre en ligne de commande :

```
latex article.tex
bibtex article
latex article.tex
latex article.tex
```

☛ **Attention** : ce n'est ni le fichier `.bib` ni le fichier `.tex` qu'on indique à BibTeX, mais le fichier `.aux` produit dans la première compilation du fichier `.tex`. Cependant, comme l'indique la commande ci-dessus, cela fonctionne quand même si on ne précise pas l'extension.

4.4 Et le français ?

C'est un des défauts du système : par défaut, la bibliographie générée par BibTeX est prévue pour être en anglais. Ainsi, tous les mots-clés sont en anglais (*in*, *Technical Report*, ...). Là, il n'y a qu'une solution : il faut créer un style de bibliographie spécifique ! Heureusement, vous n'êtes pas obligé d'étudier la structure du format `.bst` (qui définit les styles de bibliographie), car il existe déjà des feuilles de style adaptées au français. Vous pouvez télécharger de telles feuilles de style ⁽²⁾, écrites par Nicolas Markey, à partir de l'adresse :

<http://www.ctan.org/tex-archive/biblio/bibtex/contrib/bib-fr/> .

Ensuite, il vous faudra placer ces fichiers au bon endroit... Pour cela, il faut savoir que par défaut, L^AT_EX et BibTeX cherchent les fichiers de style et les fichiers bibliographiques :

- dans le répertoire courant (pas très pratique si vous voulez utiliser la même feuille de style pour plusieurs fichiers...);
- dans l'arborescence de L^AT_EX (`/usr/share/texmf/`).

Références

- [1] Nicolas MARKEY : *Tame the BeaST*.
http://www.lsv.ens-cachan.fr/~markey/BibTeX/doc/ttb_en.pdf.
- [2] Jürgen FENN : *Managing Citations and Your Bibliography with BibTeX*.
<http://www.tug.org/pracjournal/2006-4/fenn/fenn.pdf>.
- [3] Andrew ROBERTS : *Getting to Grips with L^AT_EX – Bibliography Management*.
<http://www.andy-roberts.net/misc/latex/latextutorial3.html>.

²Il s'agit de simples traductions en français des feuilles de style usuelles; ainsi, la feuille de style `plain-fr.bst` remplit exactement les mêmes fonctions que `plain.bst`, mais avec les mots-clés en français.

- [4] *BibTeX* – *Wikipedia, the free encyclopedia*, août 2007.
<http://en.wikipedia.org/wiki/BibTeX>.
- [5] Anne BUTTIGHOFFER, Alain COULAIS et Éric LEFÈVRE : *Utilisation simplifiée de BibTeX ou BibTeX pour les nuls*.
<http://butti.free.fr/BibTeX.html>.
- [6] Frédéric MEYNADIER : *Faire une bibliographie avec BibTeX*.
<http://www.tuteurs.ens.fr/logiciels/latex/bibtex.html>.
- [7] Gaël JAFFRE : *BibTeX pour les nuls*.
http://www.irit.fr/~Gael.Jaffre/LOGICIELS/LATEX_BIBTEX/bibtex1.html.
- [8] Gaël JAFFRE : *BibTeX pour les courageux*.
http://www.irit.fr/~Gael.Jaffre/LOGICIELS/LATEX_BIBTEX/bibtex2.html.
- [9] Anne-Marie POQUET : *Aide BibTeX*.
http://www.irit.fr/ACTIVITES/EQ_TCI/BIBLIO/help_bibtex_fr.html.