Cahiers GUTenberg

TRAITEMENT D'INDEX AVEC LaTeX

Philippe Louarn

Cahiers GUTenberg, nº 7 (1990), p. 23-28.

<http://cahiers.gutenberg.eu.org/fitem?id=CG_1990___7_23_0>

Cahiers GUTenberg n° 7 — Nov. 1990

Traitement d'index avec LaTeX

Philippe LOUARN

Projet OPERA, INRIA/IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France
louarn@irisa.fr

Résumé LaTeX est un logiciel de manipulation de documents structurés. La création d'environnements tels qu'une table des matières ou une bibliographie est aisée ; il est bien plus difficile d'indexer un document. Nous tentons dans cet article de présenter une synthèse de différentes méthodes de génération d'index utilisables avec LaTeX. L'étude d'un cas concret nous permettra d'appuyer nos propos.

Abstract *As LaTeX handles structured documents, it is easy to create some features, as table of contents or bibliography. But it is more difficult to create an index. This article attemps to evaluate several methods of index generation utilizable with LaTeX. A real example will illustrate our purpose.*

Introduction

Si certains types de documents, par exemple un roman, doivent se parcourir de façon séquentielle, dans de très nombreux cas, en particulier pour les textes à caractère scientifique ou technique, le lecteur recherche des *outils* lui permettant de retrouver facilement et rapidement les informations qu'il recherche au sein du document, voire d'un ensemble de documents. LaTeX permet d'automatiser la génération de la tables des matières, des tables de figures et tableaux, ainsi que de la bibliographie. Par contre, la création d'index ou de glossaires n'est pas immédiate.

Qu'est-ce qu'un index ? D'après le *Petit Robert* un index est une **liste** de termes, **triée** (généralement sur un critère alphabétique) et comportant des **références.**

La création d'un index comporte trois aspects (nous n'employons pas ici le terme *étape*, les points ci-dessous pouvant être traités en parallèle) :

1. la sélection des termes à indexer ;
2. le traitement de ces termes (tri, gestion des références...) ;
3. l'édition de l'index.

Cette étude ne portera que sur le cas où l'auteur détermine lui même quels sont les termes à indexer. La recherche automatique de ces termes — à partir d'un thésaurus, ou du nombre d'occurences d'un terme dans le texte, ou de toute autre méthode — relève plus de l'intelligence artificielle que du traitement de texte. Le lecteur interessé par ce domaine pourra se reporter à [Salton89a] et à [Salton89b].

1. Traitement "manuel"

La méthode la plus simple mais aussi la plus lourde à gérer est celle décrite dans [Spiker54] et [Dufour71] : l'auteur utilise une pré-version de son texte qui doit être paginée comme l'édition finale ; il crée une fiche pour chaque page où une terme à indexer apparaît : il ne lui reste plus (!) qu'à trier les fiches et écrire son index...

Appliquons cette méthode à LaTeX :

1. formatter le document sans l'index ;
2. relever dans le texte les termes à indexer et les numéros de page associés ;
3. saisir l'index comme décrit dans [Lamport86] ; par exemple :

```
\begin{theindex}
\item lait 20, 22, 25-27
\item laiterie
  \subitem industrielle 23, 30
  \subitem artisanale 22, 23
\indexspace
\item vache laiti\'ere
  \subitem bretonne 15, 17
  \subitem normande 16-17
\end{theindex}
```

4. inclure l'index dans le document ;

5. relancer LaTeX. L'exemple précédent donnera :

Index

lait 20, 22, 25-27
laiterie
    industrielle 23, 30
    artisanale 22, 23

vache laitière
    bretonne 15, 17
    normande 16-17

Si le résultat imprimé est satisfaisant, la méthode pour l'obtenir est archaïque. Cette solution ne peut être envisagée que pour un très petit document, dont la durée de vie est courte (pas de mise à jour).

2. La commande `\makeindex`

Outre cette saisie entièrement manuelle, LaTeX possède quelques commandes qui peuvent aider l'auteur à créer un index. L'utilisation de la commande **`\makeindex`** dans le préambule du fichier racine (**`.tex`**), et de commandes **`\index`** dans le flot du texte, commandes auxquelles l'auteur passe en paramètres les termes à indexer, va inscrire les informations pour l'index dans un fichier spécial (**`.idx`**). Ces informations sont :

- la chaîne de caractères à insérer dans l'index
- le numéro de la page où cette chaîne a été déclarée comme devant être un élément de l'index.

Les notions de sous- et de sous-sous-entrées n'existent pas. L'exemple précédent aurait été saisi :

```
page 15   \index{vache laitière bretonne}
page 16   \index{vache laitière normande}
page 17   \index{vache laitière bretonne}
          \index{vache laitière normande}
page 20   \index{lait}
page 22   \index{lait}
          \index{laiterie artisanale}
page 23   \index{laiterie artisanale}
          \index{laiterie industrielle}
page 25   \index{lait}
page 26   \index{lait}
page 27   \index{lait}
page 30   \index{laiterie industrielle}
```

Le fichier **`.idx`** créé aura l'aspect suivant :

```
\indexentry{vache laiti\'ere bretonne}{15}
\indexentry{vache laiti\'ere normande}{16}
\indexentry{vache laiti\'ere bretonne}{17}
\indexentry{vache laiti\'ere normande}{17}
\indexentry{lait}{20}
\indexentry{lait}{22}
\indexentry{laiterie artisanale}{22}
\indexentry{laiterie artisanale}{23}
\indexentry{laiterie industrielle}{23}
\indexentry{lait}{25}
\indexentry{lait}{26}
\indexentry{lait}{27}
\indexentry{laiterie industrielle}{30}
```

Le fichier **`.idx`** n'est pas traité : pas de tri, de fusion des entrées multiples, de gestion des intervalles (une même entrée dans l'index apparaissant sur plusieurs pages consécutives), etc. Ce traitement peut se faire, soit manuellement, soit en utilisant les outils informatiques disponibles sur le site où tourne LaTeX (tri, etc).

Un problème supplémentaire se pose pour les langues utilisant des caractères accentués : dans l'exemple précédent,

les commandes d'accentuation des termes indexés restent conformes aux standards TeX. Par contre, une indexation dans certains environnements, comme une note de bas de page (`footnote`) développe la commande d'accentuation. Le résulat deviendrait :

```
\indexentry{vache laiti{\accent 18 e}re
        normande}{17}
```

Ceci peut bien sûr être évité en protégeant la commande d'accentuation :

```
\index{... laiti\protect\'ere...}
```

mais la lisibilité du texte source en pâtira...

L'édition se fait avec l'environnement `theindex`. La plupart des styles LaTeX standard offrent une présentation de l'index sur 2 colonnes.

Le seul outil qu'offre LaTeX pour les index est la gestion des références aux pages. Il n'y a par contre aucun traitement disponible : tout est complètement dépendant du site.

Un cas concret

L'Irisa a assisté la société *Composcript* lors de la réalisation d'un ouvrage de sciences humaines [Vatin90]. L'index de cet ouvrage (environ 350 entrées) est de structure relativement simple : pas de sous-clef, et tri sur l'ordre alphabétique. L'emploi de la commande `\makeindex` a permis de créer le fichier `.idx`. Notre objectif était de retraiter ce fichier selon les critères suivants :

- les caractères accentués doivent avoir le même poids pour le tri que les caractères correspondants non accentués (comme le réalisé déjà Word 4) ;
- les entrées multiples doivent être fusionnées ;
- la gestion des intervalles de pages doit être automatisée (nous désirons obtenir par exemple 26-30, et non 26, 27, 28, 29, 30).

Une partie de ce travail a déjà été traitée lors de la réalisation du rapport d'activité de l'Inria [Louarn88].

Un premier programme lit séquentiellement le fichier `.idx` et capitalise le terme à indexer (ce passage en capitale tient compte des problèmes liés aux caractères accentués évoqués plus haut). Nous nous retrouvons donc avec une entrée sous la forme suivante :

```
\indexentry{VACHE LAITIERE NORMANDE}
    {vache laiti\'ere normande}{17}
```

Le résultat est sauvegardé dans un fichier temporaire. Ce fichier est trié par un appel à la commande unix :

```
sort -bu -t\{ +1 -2fd +2n -o file1 file2
```

Le fichier trié est alors parcouru séquentiellement par un programme qui fusionne les entrées multiples, détermine les intervalles de pages, et détecte les ruptures d'initiales.

Les traitements décrits ci-dessus sont intégrés dans un `script shell` afin de rendre leur utilisation transparente.

Il est alors possible de remplacer manuellement certains numéros de page par des références entre entrées de l'index (α *voir* β).

Mais cette méthode présente des inconvénients : pas de différenciation minuscule/majuscule (le prénom *Constance*, et le qualificatif *constance* seront une seule et même entrée dans l'index) ; la gestion de symboles spéciaux (lettres grecques, symboles mathématiques, ...) reste problématique. Cette solution a été très satifaisante pour ce cas particulier. Elle n'est malheureusement pas généralisable.

3. Le style `tmlindex`

Le fichier d'option de style LaTeX `tmlindex.sty`[1], de Tor Lillqvist, est, comme le dit son auteur, " pas très sophistiqué, mais si vous n'avez rien de mieux, c'est toujours ça... ".

La commande `\index` fonctionne telle qu'elle est décrite dans le manuel de Lamport, mais une notion de clef de tri optionnelle y est rajoutée. Elle supporte également les notions de sous- et de sous-sous-entrée dans l'index. Par exemple :

```
\index[clef de tri]{entr\'ee principale,
      sous-entr\'ee, sous-sous-entr\'ee}
```

L'environnement `theindex` est également modifié ; il détecte les ruptures d'initiale et propose la fonctionnalité *(continued)* décrite dans [Knuth84] : si une coupure de page intervient entre des sous-entrées d'une entrée donnée, la chaîne "*(continued)*" apparaît après le terme indexé, afin d'informer le lecteur d'une suite sur la page suivante.

Tous les traitements (notamment le tri) doivent toujours être réalisés par des outils annexes.

4. MakeIndex

MakeIndex[2] est un programme (en C) écrit par Peehong Chen, avec l'aide de Leslie Lamport [Lamport87] [Chen et al. 88]. Ce programme tourne sur de nombreux matériels (unix, vms, ms-dos, etc).

La commande `\index` est modifiée afin d'y inclure les notions de sous- et de sous-sous-clef. La notion d'intervalle est prise en compte, mais c'est à l'auteur d'en signaler les bornes. La notion de références croisées entre termes de l'index existe (ex : α, *see* alpha), mais l'auteur doit les gérer. La notion de clef de tri est incluse. Il est possible d'éditer des symboles non ascii (mathématiques ou autre) ; dans ce cas, la clef de tri est obligatoire. Ce traitement s'applique aussi aux caractères accentués. La gestion de symboles spéciaux (!, @, | et ") est délicate. Par exemple :

```
page 15   \index{vache laitière!bretonne}
page 16   \index{vache laitière!normande}
page 17   \index{vache laitière!bretonne}
          \index{vache laitière!normande}
page 20   \index{lait}
page 22   \index{lait}
          \index{laiterie!artisanale}
page 23   \index{laiterie!artisanale}
          \index{laiterie!industrielle}
page 25   \index{lait|(} % debut d'interv.
page 27   \index{lait|)} % fin d'intervalle
page 30   \index{laiterie!industrielle}
```

Le tri du fichier d'index (selon l'ordre ascii) et le traitement des entrées sont automatiques. C'est un produit très intéressant, bien documenté, et d'utilisation assez aisée ; mais l'auteur doit encore intervenir à un niveau qui serait programmable (gestion des intervalles, ...).

5. IdxTeX

Comme MakeIndex, IdxTeX [3]est un processeur annexe à LaTeX permettant de traiter les fichiers `.idx` [Aurbach87]. Les buts de ce projet sont :

- offrir un mécanisme automatique de génération d'index, de haute qualité typographique ;
- créer aisément des index, afin d'inciter les auteurs à indexer leurs documents ;

[1] Disponible sur la plupart des serveurs ftp aux USA. Voir à ce sujet l'article de Peter Flynn dans ce numéro des *Cahiers GUTenberg*. Si vous n'avez pas accès à ftp, envoyez un mail à l'auteur de cet article.

[2] MakeIndex est disponible par *anonymous ftp* sur le serveur `ymir.claremont.edu`.

[3] IdxTeX est distribué par le TUG (TeX Users Group) et par DECUS (DEC User's Society) ; actuellement, il ne tourne que dans un environnement Vax-vms.

- offrir un choix multiple de méthodes d'indexation, même les plus complexes ;
- utiliser toutes les capacitées de LaTeX pour l'indexation (en particulier l'indexation sur trois niveaux, qui n'est possible que *manuellement* en LaTeX natif) ;
- pouvoir générer un index global à un ensemble de document.

Nous ne reviendrons pas sur les 4 premiers points, qui, avec une mise en œuvre différente, offrent les mêmes services que MakeIndex. La gestion des intervalles est prise en compte par le programme. Un point intéressant : dans le cas de documents divisés en chapitres, une même entrée sur 2 pages consécutives, mais sur 2 chapitres ne sera pas considérée comme intervalle.

Index global à plusieurs documents

La notion d'index maître utilise les potentialités de gestion mémoire du système *vms*. La représentation interne de l'index est un ensemble de listes liées les unes aux autres. Chaque entrée dans l'index est un nœud de cette liste, pointant sur l'entrée suivante, et éventuellement sur une sous-entrée, pouvant elle même pointer sur une sous-sous-entrée. Un champ spécial est défini pour les tris. Les informations spécifiques aux entrées (volume, pages, références croisées) sont stockées dans des structures particulières. Les nœuds sont liés dans un ordre alphabétique, ce qui autorise un traitement séquentiel (donc rapide). Pour que la création de cet index global soit efficace, il est nécessaire qu'IdxTeX sache quels sont les fichiers auxiliaires à traiter. Ce problème est résolu par la création d'un nouveau fichier auxiliaire (`.mdx`) listant l'ensemble des fichiers `.idx` devant être pris en compte.

En guise de conclusion...

Il n'existe pas de produit miracle permettant la gestion des index dans LaTeX. Le traitement le plus sophistiqué ne tourne que sur un seul type de matériel. Il serait souhaitable de le voir généralisé.

MakeIndex est une solution satisfaisante dans bien des cas : rédaction d'articles, de thèses, de rapports ou de polycopiés de cours. Il montre très rapidement ses limites pour une documentation de gros volume.

Un traitement simple des homographes n'existe dans aucun produit. Il est possible de le simuler avec MakeIndex et IdxTeX en jouant sur les clefs de tri. Aucun des produits présent ne tient réellement compte des caractères accentués et le problème des tris sur un ordre de codification des caractères (ascii ou autre) reste entier : le passage à un codage sur 8 bits ne le résoudra pas. De plus, cet ordre des caractères dépend de la langue : si en français, un caractère accentué a le même poids que le caractère sans l'accent (É et E, par ex.), en danois ou en norvégien, les lettres Å et Ø se placent après le Z dans un classement alphabétique.

Il serait souhaitable d'avoir également les potentialités suivantes :

- référence au texte par autre chose que le numéro de page (numéro de section, par exemple) ;
- traitement de données non textuelles dans l'index (par exemple, dans un manuel informatique, indexer des icones...) ;
- indexation de données provenant d'un autre outil que LaTeX incluses

Tableau 1 : Récapituatif

	`\makeindex`	"irisa"	tmlindex	MakeIndex	IdxTEX
sous-clefs	N	N	3 niv.	3 niv.	3 niv.
"*see*"	N	M	N	I	I
tri	A	I	A	I	I
car. accentués	N	I	M (a)	M (a)	M (a)
car. spéciaux	N	N	N	I	I
diff. maj./minus.	I	N	I	I	I
intervalle	N	I	N	M	I
interv. selon structure	N	N	N	N	I (b)
index global	N	N	N	N	I

(a) : notion de clef de tri.

(b) : les intervalles tiennent compte des changements de chapitre.

M : traitement *à la main.* A : traitement effectué par un outil annexe.
I : traitement intégré dans le produit. N : n'existe pas.

dans le document (image numérisée, graphique,...) ;

- obtention d'index thématiques (index des noms de lieux, des personnes, etc) ;
- mise en œuvre du "*see also*" : si une entrée se réfère à une page du document et pointe ailleurs sur une autre entrée de l'index, cette double référence doit apparaître :
  α 12, 20, *voir aussi* β

Le tableau 1 donne un synopsis des méthodes d'indexation présentées.

Références bibliographiques

[Aurbach87] Richard L. AURBACH, Automated Index Generation for LaTeX, *TUGboat*, vol. 8(2), p 201-209, 1987.

[Chen et al. 88] Pehong CHEN et Michael H. HARRISON, Index preparation and processing, *Software—practice and experience*, vol. 18(9), p 897-915, John Wiley & Sons ed., septembre 1988.

[Dufour71] M.L. DUFOUR, *Le tapuscript*, CID, Paris, 1971.

[Knuth84] Donald E. KNUTH, *The TEXbook*, Addisson-Wesley, Reading, Massachussetts, 1984.

[Lamport86] Leslie LAMPORT, LaTeX *user's guide and reference manual*, Addisson-Wesley, Reading, Massachussetts, 1986.

[Lamport87] Leslie LAMPORT, *MakeIndex: an index processor for* LaTeX, février 1987.

[Louarn88] Philippe LOUARN, Une expérience d'utilisation de LaTeX : le rapport d'activité de l'Inria, *Cahiers GUTenberg*, no 0, p 17-23, avril 1988.

[Salton89a] Gerard SALTON, *Automatic text processing*, Addisson-Wesley, Reading, Massachussetts, 1989.

[Salton89b] Gerard SALTON, *A comparison of book indexing methods*, rapport technique no TR 89-1033, Cornell University, Ithaca (NY), août 1989.

[Spiker54] Sina SPIKER, *Indexing your book (a practical guide for authors)*, The university of Wisconsin press, 1954.

[Vatin90] François VATIN, *L'industrie du lait*, éditions L'Harmattan, Paris, 1990.