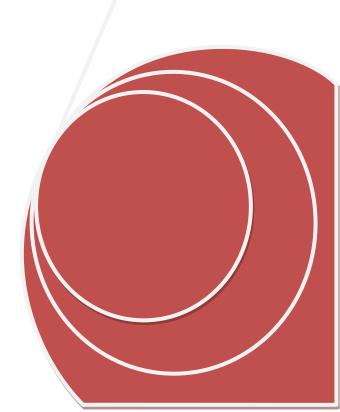


Projet de compilation

MyBib Utilitaire de manipulation de Bibtex

Projet réalisé par

- Belkhous Redha Nabil
- Zebouchi Mohammed



Sommaire

- 1. Description du projet
- 2. Présentation de la grammaire
- 3. Rapport de développement
- 4. Manuel d'utilisation
- 5. jeu d'exemple

Description du projet

Description du projet

Les fichiers de bases de données bibliographiques peuvent rapidement prendre une taille importante, et devenir, au fil des années, mal entretenus : par exemple, des doublons peuvent apparaitre, certaines conventions peuvent être respectées par endroits et pas d'autres, etc. Le sujet du projet est d'écrire un petit outil de manipulation de fichiers BibTEX, mybib, pour aider a résoudre ces problèmes.

Présentation de la grammaire

Présentation de la grammaire

Dans notre projet nous avons choisi de manipuler quatre types d'entités : *article*, *book*, *manual* et *misc*.

Un fichier Bibtex contient un ensemble d'entités, qui contiennent elles mêmes des attributs et il peut contenir des champs d'informations (Comment et String)

L'axiome S nous permet de donner soit des *Comments* suivies par une liste d'entités ou bien simplement une liste d'entités.

Une entité se compose d'un mot clef se référant à un des quatre types cités une accolade ouvrante et fermente puis entre les deux une clef et la liste des champs qui sont spécifiques à chaque types (champs obligatoires et optionnels)

Chaque champs est sous la forme d'un couple (Clef, Valeur).

```
LIST COMMENT LIST ENTITY
S:
      LIST ENTITY
COMMENT:
                                           chaine mc_par_fer
                  comment
                              mc_par_our
STRING:
            string
                        mc par our
                                          mc egal
                                                       chaine
                                                                   mc par fer
                                    mot
LIST_COMMENT:
                                    COMMENT
                                                 STRING
                  LIST COMMENT
                  COMMENT STRING
HEADING:
            nom_entity mc_virg
ENTITY:
            article mc_acc_ouv
                              HEADING
                                          LIST_CHAMPS_article
                                                                   mc_acc_fer |
                                          LIST_CHAMPS_book mc_acc_fer
            book mc_acc_ouv
                              HEADING
                                          LIST_CHAMPS_manual
            manual mc_acc_ouv
                              HEADING
                                                                   mc_acc_fer |
            misc mc_acc_ouv
                              HEADING
                                          LIST_CHAMPS_misc mc_acc_fer
LIST_ENTITY:
                  LIST_ENTITY
                                    ENTITY
                  ENTITY
```

CHAMPS_OBLIGATOIRES_article:	author mc_egal valeur_champs title mc_egal valeur_champs journal mc_egal valeur_champs year mc_egal valeur_champs
CHAMPS_OPTIONNELS_article:	volume mc_egal valeur_champs number mc_egal valeur_champs pages mc_egal valeur_champs month mc_egal valeur_champs note mc_egal valeur_champs key mc_egal valeur_champs
CHAMPS_OBLIGATOIRES_book:	author mc_egal valeur_champs title mc_egal valeur_champs publisher mc_egal valeur_champs year mc_egal valeur_champs
CHAMPS_OPTIONNELS_book:	volume mc_egal valeur_champs series mc_egal valeur_champs address mc_egal valeur_champs edition mc_egal valeur_champs note mc_egal valeur_champs key mc_egal valeur_champs month mc_egal valeur_champs
CHAMPS_OBLIGATOIRES_manual: ;	title mc_egal valeur_champs
CHAMPS_OPTIONNELS_manual:	author mc_egal valeur_champs organization mc_egal valeur_champs address mc_egal valeur_champs edition mc_egal valeur_champs note mc_egal valeur_champs key mc_egal valeur_champs month mc_egal valeur_champs year mc_egal valeur_champs
;	
CHAMPS_OPTIONNELS_misc:	author mc_egal valeur_champs title mc_egal valeur_champs howpublished mc_egal valeur_champs note mc_egal valeur_champs key mc_egal valeur_champs month mc_egal valeur_champs year mc_egal valeur_champs

LIST_CHAMPS_article: LIST_CHAMPS_article mc_virg CHAMPS_OBLIGATOIRES_article LIST_CHAMPS_article mc_virg CHAMPS_OPTIONNELS_article CHAMPS_OBLIGATOIRES_article CHAMPS_OPTIONNELS_article LIST_CHAMPS_book: LIST_CHAMPS_book mc_virg CHAMPS_OBLIGATOIRES_book LIST_CHAMPS_book mc_virg CHAMPS_OPTIONNELS_book CHAMPS OBLIGATOIRES book CHAMPS_OPTIONNELS_book LIST_CHAMPS_manual: LIST_CHAMPS_manual mc_virg CHAMPS_OBLIGATOIRES_manual LIST_CHAMPS_manual mc_virg CHAMPS_OPTIONNELS_manual CHAMPS_OBLIGATOIRES_manual CHAMPS_OPTIONNELS_manual LIST CHAMPS misc: LIST_CHAMPS_misc mc_virg CHAMPS_OPTIONNELS_misc CHAMPS OPTIONNELS misc

Remarque : Plus de détails sur la partie Flex, Bison et routines sémantiques seront données.

Rapport de développement

Rapport de développement

Nous nous basons tout au long du projet sur une structure en liste chainée, qui reprend le concept d'ensemble d'entités qui eux-mêmes regroupes des champs spécifiques.

Au lancement de la compilation, un fichier .*bib* est donné en entrée afinl'analyseur lexical Flex lise chaque entité et l'identifie comme étant propre au langage.

Par la suite, chaque entité est envoyée pour une analyse syntaxico-sementique faite par Bison en comparent avec la grammaire les entités reçues et leurs sens à travers les routines sémentiques.

Pour chaque entité lue par Flex et Bison, elle est stockée dans la structure en pensent à sauvegarder avec, quelques informations importantes comme indiquer si l'entité existe déjà dans la structure ou encore l'appartenance de chaque champs à son entité, chaque valeur à son champs et chaque clef à son type d'entité.

L'ajout des entités se fait grâce à une fonction « *addEntity* » et celui des champs et leurs valeurs avec une autre fonction « *addParamToCurrentEntity* » (*Routines sémentiques*)

Pour les fonctionnalités que nous avons développées, elle sont lancé en suffixant l'executable avec une otion (Ex:./lunch.exe -c).

Les fonctionnalités que nous avons réalisé sont :

- 1. mybib –c toto.bib
- 2. mybib –k toto.bib
- 3. mybib –s toto.bib
- 4. mybib –u toto.bib
- 5. mybib t type toto.bib

4

Manuel d'utilisation

Manuel d'utilisation

L'utilisation de l'application est faite de manière a être très simple d'utilisation et intuitif.

On lance le makefile pour compiler et générer l'exécutable, ensuite il vous suffit de lancer l'exécutable avec l'option que vous désirez.

Si l'utilisateur veut :

vérifie que chaque entrée a une clef qui lui est propre : il ajout à la fin de l'exécutable
 « -c »

Exemple: ./lunch.exe-c

- Remplace les clefs de toto. bib par des clefs normalisées : il ajoute le suffixe « -k » Exemple : ./lunch.exe –k
- Pour tous les enregistrements, extraire (s'ils existent) les champs *publisher*, *organization*, *series*, *journal*, et les définir à partir d'une chaine de caracteres BibTEX : il lance l'option « -s »

Exemple: ./lunch.exe –s

- supprime les doublons : pour le même type, les doublons sont retirés et stockés dans un fichier annexe de nom *bib.doublons*

Exemple: ./lunch.exe -u

- Ne garde que les entrées de type type (article, book, misc et manual)

Exemple: ./lunch.exe -t

Jeu d'exemple

Jeu d'exemple

Pour vous guider et vous permettre de voir toutes les fonctionnalités développées nous vous suggérons de suivre ce scénario :

- o Lancer le *Command.sh*
- Lancer l'exécutable sans aucune option pour bien voir que le fichier exemple.bib se compile correctement.
- Faire des modifications dans ce dernier en faisant délibérément des erreurs et relancer l'exécutable pour distinguer la ligne ou se trouve l'erreur.
- o Revenir au fichier tel qu'il été initialement.
- Lancer l'exécutable avec l'option « -c » pour pouvoir constater les éléments qui ont la même clef.
 - ./mybib.out –c filename.bib
- o Lancer ensuite l'exécutable avec l'option « -k » pour normaliser les clefs.
 - ./ mybib. out –k *filename.bib*
- o Lancer l'exécution avec l'option « -s » pour générer les *Comments*.
 - ./ mybib. out –s *filename.bib*
- Lancer l'exécutable avec l'option « -u » pour supprimer les doublons et vérifier dans le fichier généré.
 - ./ mybib. out –u *filename.bib*
- O Lancer une dernière exécution avec l'option « -t ».
 - / mybib. out –t article *filename.bib* pour ne prendre que les entités de type article et les mettrent dans un autres fichier *Bibtex_Selection.bib*