Langage de Recherche d'Information

Travaux Pratiques

Jérôme Farinas
jerome.farinas@irit.fr

Institut de Recherche en Informatique de Toulouse
Université Paul Sabatier

5 décembre 2013

Conseils généraux pour les TP

- Chaque exercice idéalement doit être testé, compris et parfois complété. Pour être sauvegardé comme fichier éventuellement modifié puis exécuté, chaque source fourni devra être saisi dans un éditeur de texte (`nedit` par exemple)
- Il est conseillé de ranger les exercices dans des sous-répertoires spécifiques au TP. Les noms des fichiers sont proposés avec l'extension `.perl` (qui ne joue aucun role, juste un signe de reconnaissance...)
- Sous Unix, les fichiers pour être exécutables doivent porter la permission `x`. Si nécessaire donner cette permission en passant la commande, comme : `chmod a+x monfichier.perl`
- Chaque script doit indiquer le chemin vers Perl avec cette 1ère ligne : `#!/usr/bin/perl -w` (vérifier le chemin sur la machine que vous utilisez en tapant dans le shell : `which perl`)
- Un dossier sera constitué à la fin des TPs : tous les exercices marqués avec une étoile devront etre présents. La notation prend en compte le fait que le programme marche, la présentation du programme (commentaires et indentation) ainsi que les méthodes utilisées.

Plan

Plan

Présentation du TP 5

- Ce dernier sujet de TP n'est pas aussi dirigé que les précédents, vous allez être amené à orienter la réalisation de votre programme à la façon d'un mini-projet. Vous avez un mois pour réaliser le code.
- Dans le transparent suivant, je vous indique un embryon de programme que vous devrez réaliser.
- Je vous indique ensuite 6 améliorations ou fonctionnalités optionnelles à rajouter à ce programme de base.
- Vous pouvez piocher dans cette liste de suggestions, ou bien proposer les votre si vous les jugez pertinente.
- L'objectif du programme étant de récupérer des pages internet, d'y appliquer un nettoyage pour récupérer les informations de la page, et d'exporter ces informations pour une utilisation future.

Plan

Sujet de base I

- La finalité du programme est de pouvoir récupérer des informations sur Internet et les stocker de manière à pourvoir les réutiliser. Vous devrez récupérer des informations sur un site référençant des actualités : `http://fr.wikinews.org/`. Nous allons tous travailler sur la même page : `https://fr.wikinews.org/wiki/France_:_conférence_de_Richard_Stallman_à_Paris`
- Écrivez un programme Perl qui filtre cette page et permettre de ne garder que les champs de données interessant. Le résultat serait envoyé sous forme de texte sur la sortie standard, les intitulés seraient écrit en majuscules.
- Récupérez la page avec la commande Unix `wget`.

Sujet de base II

- Exemple de résultat :

```
TITRE : France :  conférence de Richard Stallman à
Paris
DATE : 14 novembre 2010
SOMMAIRE :
1 Rappel historique
2 Situation actuelle
3 Propositions pour le futur
4 Interview
5 Notes
6 Sources
CATEGORIES : 14 novembre 2010, Article publié, Article
archivé, France, Logiciel libre, Informatique, Paris
SOURCES : ...
```

Plan

Fonctionnalités optionnelles

1. Permettre de récupérer une URL directement et non plus passer par un fichier. Si ce n'est pas faisable avec le perl installé sur vos machines, indiquez ce qu'il faut faire pour que cela puisse fonctionner.
2. Permettre de traiter un fichier liste d'URL à traiter (ou une page contenant des liens vers les articles).
3. Transformez votre programe de façon à utiliser des fonctions et sous-fonctions pour réaliser les blocs de traitement. Stockez vos fonctions dans un fichier bibliothèque que vous appellerez (utilisation de `use`) lors de votre programme principal.
4. Stockez les informations dans un tableau associatif. Sauvegardez dans un fichier ce tableau associatif pour une utilisation ultérieure (astuce : cf. `fonction dbmopen`).
5. Exportez les informations sous forme d'un fichier SQL (à vous de définir la structure de la base de donnée).
6. Créer un menu texte pour guider l'utilisation de votre programme sur les différentes possibilités ou les choix de paramètres.

Plan

Notation du TP

- Pour évaluer ce TP, je vous propose le schéma suivant :
  - 10 points si le sujet de base est traité et fonctionnel
  - 1 point par fonctionnalité supplémentaire ajouté
- Pour avoir 20 il faudra donc 10 fonctionnalités en plus du programme de base. Donc au moins 4 proposées par vous.
- Question bonus : Cette approche est-elle la meilleure façon de faire pour parser un fichier HTML ? Existe-t-il des moyens plus performants et plus simples en Perl ? Lesquels ?

Bibliographie

1. `http://www.perl.org/`
   - le site officiel
2. `http://perso.univ-rennes1.fr/francois.dagorn/perl/`
   - un site avec un cours synthétique sur Perl
3. `http://www.enstimac.fr/Perl/DocFr.html`
   - documentation en français
4. `http://articles.mongueurs.net/planning.html`
   - articles de revues