|  |
| --- |
| INSA Lyon |
| TP 4 C++ : analyse des logs apache |
| Manuel d’utilisation |

|  |
| --- |
| Martin BONNEFOY Ambre HUTIER  B3203 |

## Exécutable

L’exécutable s’appelle analog. La commande make permet de le créer.

L’utilisation la plus simple est :

$./analog anonyme.log

## Configuration

L’url de base du serveur est configurable dans le fichier Controleur.h. Il s’agit de la variable baseUrl.

## Options

L’outil proposé prend en paramètres le nom du fichier de log ainsi qu’une série d’options en ligne de commande :

$./analog [options] nomfichier.log

Par défaut, c’est-à-dire quand il n’y a aucune option, il affichera sur la console sous forme textuelle la liste des 10 documents les plus consultés par ordre décroissant de popularité. Aucun fichier “.dot” n’est généré dans ce cas. Le fichier à analyser doit être une .log.

Les options en ligne de commande sont les suivantes :

[-g nomfichier.dot]

Cette option permet de produire un fichier au format GraphViz du graphe analysé. Chaque document apparaîtra sous la forme d’un nœud et chaque arc indiquera le nombre de parcours associés. Le fichier généré doit être un .dot

[-e]

Cette option permet d’exclure tous les documents qui ont une extension de type : js, css, png, jpg, jpeg, gif, ico, bmp.

[-t heure]

Cette option permet de ne prendre en compte que les hits qui sont dans le créneau horaire correspondant à l’intervalle [heure, heure+1[.

L’ordre de ces options n’a pas d’importance. Elles sont toutes utilisées indépendamment, et peuvent donc être utilisées selon n’importe quelle combinaison. Le fichier de log doit simplement être le dernier paramètre de la commande.

## Tests

5 tests fonctionnels sont réalisés dans le dossier Tests. Chaque test vérifie le comportement d’un composant du système et se découpe en plusieurs sous-tests pour les différents cas possibles.

Les tests sont exécutables via le script mktest.sh, depuis le dossier Tests. Il n’y a pas besoin de construire l’exécutable au préalable pour les exécuter.

## Exemple

Pour les exemples, le jeu d’essai suivant est utilisé :

192.168.0.1 - - [08/Sep/2012:11:15:00 +0200] "GET /page2.html HTTP/1.1" 200 2000 "http://intranet-if.insa-lyon.fr/page1.html" "Mozilla/5.0"

192.168.0.1 - - [08/Sep/2012:11:37:00 +0200] "GET /page1.html HTTP/1.1" 200 1000 "http://intranet-if.insa-lyon.fr/page2.html" "Mozilla/5.0"

192.168.0.1 - - [08/Sep/2012:12:12:00 +0200] "GET /page2.html HTTP/1.1" 200 - "http://intranet-if.insa-lyon.fr/page3.html" "Mozilla/5.0"

192.168.0.1 - - [08/Sep/2012:12:16:00 +0200] "GET /page3.html HTTP/1.1" 200 3000 "-" "Mozilla/5.0"

192.168.0.1 - - [08/Sep/2012:12:16:01 +0200] "GET /image.jpg HTTP/1.1" 200 50000 "http://intranet-if.insa-lyon.fr/page3.html" "Mozilla/5.0"

192.168.0.1 - - [08/Sep/2012:12:59:00 +0200] "GET /page2.html HTTP/1.1" 200 2000 "http://intranet-if.insa-lyon.fr/page1.html" "Mozilla/5.0"

Le résultat du top 10 est le suivant, lorsqu’aucune option de filtre n’est utilisée :

/page2.html - 3 hits

/image.jpg - 1 hits

/page3.html - 1 hits

/page1.html - 1 hits

Le graphe généré est le suivant :

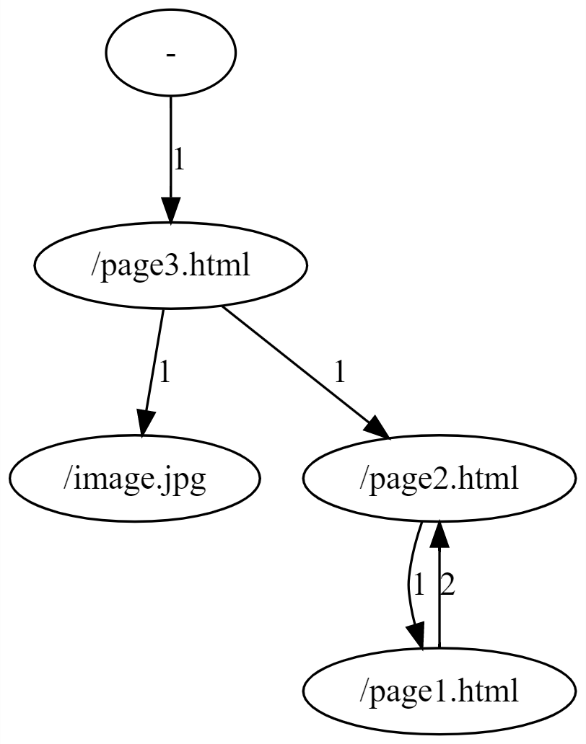


Figure : graphe généré