Module X
Expressions régulières

Contenu du module

- Présentation
- Méta-caractères
- Facteurs d’occurrences
- Motifs d’ancrage
- Caractères divers
- Captures et références arrières

Présentation

- Expressions régulières ou expressions rationnelles ;
- Modèles de chaînes de caractères ;
- Existent nativement dans certains outils ou dépendent de modules ou librairies tiers ;
- Trois formes principales *(compatibilité ascendante)*
  - E.R. de base — utilisées entre autres par `grep`, `sed`, `vi`
  - E.R. étendues — utilisées par exemple par `awk`, `sed -E`, `egrep`, ou `grep -E`
  - E.R. de Perl — communément nommées PCRE
- Précautions d'écriture
  - Doit faire juste ce qui lui est demandé ;
  - Ne doit pas en faire de trop ;
  - Doit en faire suffisamment.

Méta-caractères

Diverses formes

.	Caractère joker, en lieu et place de n'importe quel caractère ;
[...]	N'importe quel caractère présent dans la liste entre crochets ;

- La liste n'est pas une chaîne
- Possibilité d'indiquer une étendue par ex. `[m-t]` pour `[mnopqrst]`

[^...]	N'importe quel caractère non présent dans la liste ;

Extensions PCRE

\w	tout caractère alphanumérique et le caractère _ (souligné) ;
\d	tout chiffre
\s	tout caractère d'espacement
\W	négation de \w
\D	négation de \d
\S	négation de \s

NB *tout méta-caractère ne vaut que pour une et une seule occurrence.*

Classes de caractères

Une classe de caractères est un ensemble de caractères, ex. les chiffres, lettres, minuscules…
La syntaxe principale est la forme Posix, étendue par celle des PCRE aux caractères Unicode.

Formes Posix

La forme Posix définit les classes de caractères suivantes :

`[:alnum:]`	caractère alphanumérique
`[:alpha:]`	caractère alphabétique
`[:ascii:]`	caractère ascii
`[:cntrl:]`	caractère de contrôle
`[:digit:]`	chiffre
`[:graph:]`	caractère affichable sauf l'espace
`[:lower:]`	minuscule
`[:print:]`	caractère affichable, y compris l'expace
`[:punct:]`	caractère affichable sauf lettre, chiffre ete espace
`[:space:]`	caractère d'espacement (SP, VT, LF, HT, FF, CR)
`[:upper:]`	majuscule
`[:xdigit:]`	chiffre hexadécimal

Facteurs d'occurrences

- permettent de préciser le nombre d'occurrences d'un motif ;
- s'appliquent toujours au caractère/motif immédiatement précédent.

Syntaxe de base

- `\{n1,n2\}` *(ERB – **NB** : les \ et les {} doivent être protégés du SHELL)*
- `{n1,n2}` *(ERE/PCRE – **NB** : les {} doivent être protégées du SHELL)*

⇒ De n1 à n2 occurrences du caractère/motif précédent.

Variations *(présentées avec la syntaxe ERE mais existent en ERB)*

- `{n1,}` ⇒ Au moins n1 occurrences
- `{n1}` ⇒ Exactement n1 occurrences (!!!)

Simplifications *(spécifiques aux ERE/PCRE)*

- * ⇒ {0,}
- ? ⇒ {0,1}
- + ⇒ {1,}

Facteurs d'occurrences - Gloutonnement

Gloutonnement ⇒ Faire correspondre le maximum d'occurrences

Exemple

```
bab
baaab
baaaaaab
```

Voici une expression gloutonne

```
$ grep -E --color 'ba+' regex.txt
bab
baab
baaaaaab
```

Limitations du gloutonement *(spécifiques aux PCRE)*

Le caractère ? suffixant un facteur d'occurrences en limite le gloutonnement.

```
$ grep -P --color 'ba+?' regex.txt
bab
baab
baaaaab
```

Motifs d’ancrage

Les motifs d’ancrage permettent de symboliser le début et la fin d’une ligne :

^ symbolise un début de ligne *(si placé en début d’expression)*

$ symbolise la fin d’une ligne *(si placé en fin d’expression)*

Exemples

```
^[A-Z]    Chaîne débutant par une majuscule
^Bob$     Chaîne ne contenant que ‘Bob’
^$        Chaîne vide
[0-9]$    Chaîne se terminant par un chiffre
```

Remarque

Il est vivement conseillé d’utiliser les motifs d’ancrage chaque fois que cela est une évidence, cela limitera les situations d’expressions trop larges !

Caractères divers

- Présentation
- Méta-caractères
- Facteurs d’occurrences
- Motifs d’ancrage
- Caractères divers
- Captures et références arrières

Captures et références arrières

Les (), outre la fonctionnalité de groupement de motifs, possèdent la capacité de mémoriser les caractères auxquels elles correspondent.

```
Reste à faire...
```