LES EXPRESSIONS REGULIERES

Présentation

Une expression régulière (en anglais Regular Expression ou RE) sert à identifier une chaîne de caractère répondant à un certain critère (par exemple chaîne contenant des lettres minuscules uniquement). L'avantage d'une expression régulière est qu'avec une seule commande on peut réaliser un grand nombre de tâche qui seraient fastidieuses à faire avec des commandes UNIX classiques.

Les commandes ed, vi, ex, sed, **awk**, **expr** et **grep** utilisent les expressions régulières. Les expressions régulières sont aussi utilisé en Perl.

L'exemple le plus simple d'une expression régulière est une chaîne de caractères quelconque **toto** par exemple. Cette simple expression régulière va identifier la prochaine ligne du fichier à traiter contenant une chaîne de caractère correspondant à l'expression régulière.

Si l'on veut chercher une chaîne de caractère au sein de laquelle se trouve un caractère spécial (/, *****, **$**, **.**, **[**, **]**, **{**, **}**, **!**, entre autres) (appelé aussi méta caractère), on peut faire en sorte que ce caractère ne soit pas interprété comme un caractère spécial mais comme un simple caractère. Pour cela vous devez le faire précéder par \ (backslash). Ainsi si votre chaîne est **/dev**, pour que le / ne soit pas interprété comme un caractère spécial, vous devez tapez \ /dev pour l'expression régulière.

Les différentes expressions régulières sont :

- `^ début de ligne`
- `. un caractère quelconque`
- **`$`** `fin de ligne`
- **`x*`** `zéro ou plus d'occurrences du caractère` **`x`**
- **`x+`** `une ou plus occurrences du caractère` **`x`**
- **`x?`** `une occurrence unique du caractère` **`x`**
- **`[...]`** `plage de caractères permis`
- **`[^...]`** `plage de caractères interdits`
- **`\ {n\}`** `pour définir le nombre de répétition` **`n`** `du caractère placé devant`

On utilise aussi des métas caractères pour indiquer des remplacements de caractères :

Le méta caractère .

Le méta caractère**.** remplace dans une expression régulière un caractère unique. Par exemple

toto**.** va identifier toutes les lignes contenant la chaîne toto suivi d'un caractère quelconque unique. Si vous voulez identifier les lignes contenant la chaîne **.cshrc**, l'expression régulière correspondante est \ **.cshrc**

Les méta caractères [ ]

Les métacaractères **[]** permettent de désigner des caractères compris dans un certain intervalle de valeur à une position déterminée d'une chaîne de caractères. Par exemple **[Ff]raise** va identifier les chaînes **Fraise** ou **fraise**, **[a-z]toto** va identifier une chaîne de caractère commençant par une lettre minuscule (intervalle de valeur de **a** à **z**) et suivi de la chaîne **toto** (**atoto**, **btoto**, ..., **ztoto**).

D'une manière plus générale voici comment **[]** peuvent être utilisés.

[A-D] intervalle de **A** à **D** (**A**, **B**, **C**, **D**) **par exemple bof[A-D] donne bofA**, **bofB**, **bofC**, **bofD**

[2-5] intervalle de **2** à **5** (**2**, **3**, **4**, **5**) par exemple **12[2-5]2 donne 1222**, **1232**, **1242**, **1252**

[2-56] intervalle de **2** à **5** et **6** (et non pas **56**) (**2**, **3**, **4**, **5**, **6**) **par exemple 12[2-56]2 donne 1222**, **1232**, **1242**, **1252**, **1262**

[a-dA-D] intervalle de **a** à **d** et **A** à **D** (**a**, **b**, **c**, **d**, **A**, **B**, **C**, **D**) **par exemple z[a-dA-D]y donne zay**, **zby**, **zcy**, **zdy**, **zAy**, **zBy**, **zCy**, **zDy**

[1-3-] intervalle de **1** à **3** et **-** (**1**, **2**, **3**, **-**) **par exemple [1-3-]3 donne 13**, **23**, **33**, **-3**

[a-cI-K1-3] intervalle de **a** à **c**, **I** à **K** et **1** à **3** (**a**, **b**, **c**, **I**, **J**, **K**, **1**, **2**, **3**)

On peut utiliser **[]** avec un ^ pour identifier l'opposé de l'expression régulière. Voici un exemple: **[^0-9]toto** identifie les lignes contenant une chaîne **toto**, le caractère juste avant ne doit pas être un chiffre (exemple **atoto**, **gtoto** mais pas **1toto**, **5toto**). Autre exemple **[^a-zA-Z]** n'importe quel caractère sauf une lettre minuscule ou majuscule. Attention à la place de ^, si vous tapez **[1-3^]**, c'est équivalent aux caractères **1**, **2**, **3** et ^.

Les méta caractères ^ et $

Le méta caractère ^ identifie un début de ligne. Par exemple l'expression régulière ^**a** va identifier les lignes commençant par le caractère **a**.

Le méta caractère **$** identifie une fin de ligne. Par exemple l'expression régulière **a$** va identifier les lignes se terminant par le caractère **a**.

L'expression régulière ^**toto$** identifie les lignes qui contiennent strictement la chaîne **chaine**.

L'expression régulière ^**$** identifie une ligne vide.

Le méta caractère *

Le méta caractère ***** est le caractère de répétition.

L'expression régulière **a*** correspond aux lignes comportant 0 ou plusieurs caractère **a**. Son utilisation est à proscrire, car toutes les lignes, même celles ne contenant pas le caractère **a**,

répondent aux critères de recherche. **x*** est une source de problèmes, il vaut mieux éviter de l'employer.

L'expression régulière **aa*** correspond aux lignes comportant 1 ou plusieurs caractères **a**.

L'expression régulière **[a-z][a-z]*** va chercher les chaînes de caractères contenant 1 ou plusieurs lettres minuscules (de **a** à **z**).

L'expression régulière **[^][^]*** est équivalent à tout sauf un blanc.

Les méta caractères \ (\)

Pour le traitement complexe de fichier, il est utile parfois d'identifier un certain type de chaîne pour pouvoir s'en servir dans la suite du traitement comme un sous programme. C'est le principe des sous chaînes, pour mémoriser une sous chaîne, on utilise la syntaxe \ **(expression régulière)**\, cette sous chaîne sera identifié par un chiffre compris par 1 et 9 (suivant l'ordre de définition).

Par exemple \ **([a-z][a-z]*)**\ est une sous chaîne identifiant les lignes contenant une ou plusieurs lettres minuscules, pour faire appel à cette sous chaîne, on pourra utiliser \ **1**.

Exemples

unix : cherche la chaine de caractère unix et aucune autre chaine
unix* cherche la chaîne de caractère uni, unix, unixxx ….
(unix)* cherche la chaîne unix ou unixunix ou rien mais pas unixx
^unix cherche la chaîne unix en début de ligne
L'expression **[a-z][a-z] *** cherche les lignes contenant au minimum un caractère en minuscule.
[a-z] caractère permis, **[a-z]*** recherche d'occurrence des lettres permises.