



I106B

7. Globbing

[HTTPS://WEB.ARCHIVE.ORG/WEB/20161027105732/HTTP://R
YANSTUTORIALS.NET:80/LINUXTUTORIAL/WILDCARDS.PHP](https://web.archive.org/web/20161027105732/http://ryanstutorials.net:80/linuxtutorial/wildcards.php)

Globbering

2

- ▶ Une chaîne est un **motif générique** (*wildcard pattern*) si elle contient un ou plusieurs caractères parmi « ? », « * » et « [».
- ▶ Le développement (*globbing*) est l'opération qui transforme un motif générique en une liste de **noms de fichiers** correspondant à ce motif.
- ▶ Les motifs génériques ne sont pas des expressions rationnelles (regex)!
- ▶ Cf. [man 7 glob](#)

- ▶ **Avant** d'exécuter une commande, le shell traite les **méta-caractères** :
- ▶ ***** : désigne tous les chemins où * remplace 0, 1 ou plusieurs caractères
- ▶ **?** : désigne tous les chemins où ? remplace exactement 1 caractère
- ▶ **[XXX]** : désigne tous les chemins où [XXX] remplace exactement 1 des caractères entre les crochets

Exemple de globbing

4

`ls b*`

- ▶ Avant d'exécuter la commande `ls`, le shell va d'abord résoudre `b*`
- ▶ Supposons que le répertoire courant contienne exactement : `alf`, `bob`, `barry` et `charles`
- ▶ `b*` est remplacé par `bob` et `barry`
- ▶ La commande réellement exécutée sera donc :
`ls bob barry`

- ▶ Commande pour trouver des fichiers dans une arborescence.

```
find chemin -name nom
```



nom peut utiliser les méta-caractères du globbing. Dans ce cas, il faut écrire "nom*" (et pas nom*) sinon le *globbing* intervient **avant** l'exécution du `find` qui ne verra pas l'étoile

- ▶ Il existe de nombreux autres critères de recherche (cfr. `man find`)

find -exec

6

```
find chemin -name nom -exec commande {} \;
```

Ici, `find` va effectuer la recherche demandée et, pour chaque occurrence trouvée, il exécutera la *commande* fournie.

`{}` sera remplacé par le nom du fichier trouvé et prendra la place d'un argument de la commande.

ex :

```
find / -name "*.back" -exec rm {} \;
```