Construire un outil de CLI userfriendly

Inventons la Command Line eXperience !

Nicolas Lepage - @njblepage

⚠ Disclaimer ⚠

```
$ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.0 165992 11536 ?        Ss   12:00   0:01 /sbin/init
root           2  0.0  0.0      0     0 ?        S    12:00   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   12:00   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   12:00   0:00 [rcu_par_gp]
root           6  0.0  0.0      0     0 ?        I<   12:00   0:00 [kworker/0:0H-events_highpri]
root           8  0.0  0.0      0     0 ?        I<   12:00   0:00 [mm_percpu_wq]
...
```

a : Les process de tout les utilisateurs

x : Même les process sans tty

u : Format orienté utilisateur

Conventions

Options

Options longues

```
$ grep \
    --ignore-case \
    --invert-match \
    --count \
    needle haystack.txt
21
```

Options courtes

```
$ grep -i -v -c needle haystack.txt
21
$ grep -ivc needle haystack.txt
21
```

Options usuelles

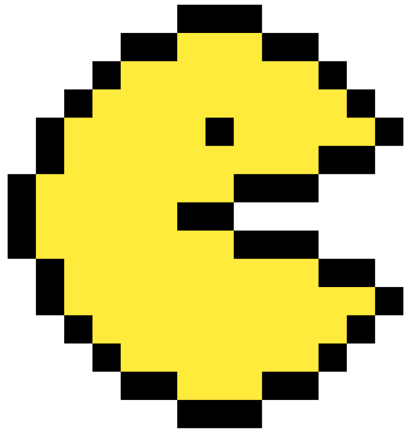
```
$ grep --version
grep (GNU grep) 3.6
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Written by Mike Haertel and others; see
<https://git.sv.gnu.org/cgit/grep.git/tree/AUTHORS>.
```

Options usuelles

```
$ grep --help
Usage: grep [OPTION]... PATTERNS [FILE]...
Search for PATTERNS in each FILE.
Example: grep -i 'hello world' menu.h main.c
PATTERNS can contain multiple patterns separated by newlines.

Pattern selection and interpretation:
  -E, --extended-regexp     PATTERNS are extended regular expressions
  -F, --fixed-strings       PATTERNS are strings
...
```



pacman

pacman

```
$ pacman -h
usage:  pacman <operation> [...]
operations:
    pacman {-h --help}
    pacman {-V --version}
    pacman {-D --database} <options> <package(s)>
    pacman {-F --files}    [options] [file(s)]
    pacman {-Q --query}    [options] [package(s)]
    pacman {-R --remove}   [options] <package(s)>
    pacman {-S --sync}     [options] [package(s)]
    pacman {-T --deptest}  [options] [package(s)]
    pacman {-U --upgrade}  [options] <file(s)>

use 'pacman {-h --help}' with an operation for available options
```

git

Commandes git

```
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
    modified:   slides.md

no changes added to commit
$
$ git commit -am "Add slides about git"
[main 536e1f3] Add slides about git
 1 file changed, 25 insertions(+), 2 deletions(-)
```

Groupes de commandes

```
$ docker image

Usage:  docker image COMMAND

Manage images

Commands:
  build       Build an image from a Dockerfile
  history     Show the history of an image
  import      Import the contents from a tarball to create a filesystem image
  inspect     Display detailed information on one or more images
  load        Load an image from a tar archive or STDIN
  ls          List images
  prune       Remove unused images
  pull        Pull an image or a repository from a registry
  push        Push an image or a repository to a registry
  rm          Remove one or more images
  save        Save one or more images to a tar archive (streamed to STDOUT by default)
  tag         Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE

Run 'docker image COMMAND --help' for more information on a command.
```

git checkout ?

Switch branches or restore working tree files catption

🐱 catption

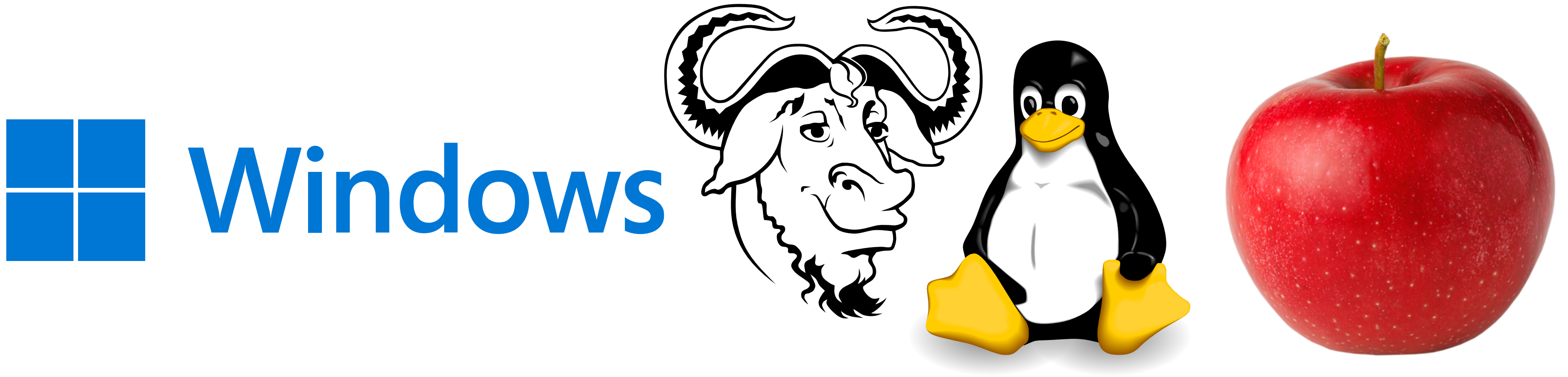
```
$ catption -h
Cat caption generator tool

Usage:
  catption [flags] <input file>

Flags:
  -b, --bottom string    Bottom text
      --fontSize float   Font in points (default 96)
  -h, --help             help for catption
      --margin float     Top/bottom text margin (default 20)
  -o, --out string       Output file (default "out.jpg")
  -s, --size float       Output image size (default 1024)
  -t, --top string       Top text
  -v, --version          version for catption
```

```
[nico Pictures]🍺 
```

Intégration avec l’OS



Ouvrir le fichier généré

Ouvrir le fichier généré

- Windows : start
- macOS : open
- Linux : xdg-open

```
[nico Pictures]🍺
```

(Ne pas) Ouvrir le fichier généré

```
$ catption -h
Cat caption generator CLI

Usage:
  catption [flags] <input file>

Flags:
  -b, --bottom string    Bottom text
      --fontSize float   Font in points (default 96)
  -h, --help             help for catption
      --margin float     Top/bottom text margin (default 20)
      --open             Open file with system viewer (default true)
  -o, --out string       Output file (default "out.jpg")
  -s, --size float       Output image size (default 1024)
  -t, --top string       Top text
  -v, --version          version for catption
```

Configurable

Configurer mes dossiers d’images de chats

Configurer mes dossiers d’images de chats

```
$ catption help dir
Manages input files directories

Usage:
  catption dir [command]

Available Commands:
  add         Adds an input files directory
  list        Lists input files directories
  remove      Removes an input files directory

Flags:
  -h, --help   help for dir

Use "catption dir [command] --help" for more information about a command.
```

Un peu d’interactivité

```
[nico Pictures]🍺 
```

Idées en plus

- Conseiller / guider l’utilisateur
- Autocomplétion personalisée
- Logs sympas (couleur, émojis, …)

🐱 github.com/nlepage/catption

cobra **viper**

github.com/mattn/go-isatty

Packages os et os/exec de la bibliothèque standard

Autres langages

- NodeJS : commander
- Java : picocli
- Python : click
- Et bien d’autres

Références

- https://man.archlinux.org/man/ps.1.en
- https://www.gnu.org/software/libc/manual/html_node/Argument-Syntax.html
- https://man.archlinux.org/man/grep.1.en
- https://man.archlinux.org/man/pacman.8.en
- https://git-scm.com/docs
- https://docs.docker.com/engine/reference/commandline/image/
- https://www.freedesktop.org/wiki/Software/xdg-utils/

Slides : nlepage.github.io/build-userfriendly-cli-tool-talk