Building Great CLI Experiences in Node

Node+JS Interactive 2018

Jeff Dickey
CLIs are powerful

- Repeatable or composable tasks
- Creating machine-readable output
- Scripting
- Interacting with an API
Building for a CLI takes a fraction of the effort of the effort of web development, which suits...

- Internal or admin tools
- Prototyping functionality
- Testing a new feature with customers
oclif

Open CLI Framework. Built with 💜 by Heroku.

🔗 https://github.com/oclif/oclif
CLIs built using oclif
Today:

12 Factor CLI Apps

oclif Demo
The Twelve-Factor App

I. Codebase
One codebase tracked in revision control, many deploys

II. Dependencies
Explicitly declare and isolate dependencies

III. Config
Store config in the environment

IV. Backing services
Treat backing services as attached resources

V. Build, release, run
Strictly separate build and run stages

VI. Processes
Execute the app as one or more stateless processes

VII. Port binding
Export services via port binding

VIII. Concurrency
Scale out via the process model

IX. Disposability
Maximize robustness with fast startup and graceful shutdown

X. Dev/prod parity
Keep development, staging, and production as similar as possible

XI. Logs
Treat logs as event streams

XII. Admin processes
Run admin/management tasks as one-off processes

...
6 of the 12 CLI Factors

medium.com/@jdxcode
Great help is essential
Great help is essential

Make help accessible through every possible avenue

```
# list all commands
$ mycli
$ mycli --help
$ mycli help
$ mycli -h

# get help for subcommand
$ mycli subcommand --help
$ mycli subcommand -h
```
Factor 1

Great help is essential
Factor 2

Prefer flags to args

$ heroku fork FROMAPP --app TOAPP

$ heroku fork --from FROMAPP --to TOAPP
Factor 3
What version am I on?

```
~/.config/oclif

heroku/7.16.6-beta.fbd11fd darwin-x64 node-v10.11.0
```

```
$ mycli version # multi only
$ mycli --version
$ mycli -V
```
$ myapp > foo.txt
Warning: something went wrong
Factor 4

Mind the streams

```
tar^C
curl https://cli-assets.heroku.com/heroku-darwin-x64.tar.xz > heroku.tar.xz
```
Factor 5: Handle things going wrong

1. Error code
2. Error title
3. Error description (Optional)
4. How to fix the error
5. URL for more information

```bash
$ myapp dump -o myfile.out
Error: EPERM - Invalid permissions on myfile.out
Cannot write to myfile.out, file does not have write permissions.
Fix with: chmod +w myfile.out
https://github.com/jdxcode/myapp
```
Factor 6

Be Fancy!
Factor 6

Be Fancy!
Disable if:

- Not a tty (usually redirecting to file)
- TERM=dumb
- NO_COLOR=1
- --no-color
yarn add v1.3.1
[2/5] Resolving packages...
[3/6] Fetching packages...
warning Pattern ("@oclif/config@1.3.59") is trying to unpack in the same destination "/Users/dickeyxxx/Library/Caches/Yarn/v1/npm-@oclif/config@1.3.59-10cc39757654850458b2d3f8991655af2e869deb5" as pattern ("@oclif/config@1.3.59"). This could result in non-deterministic behavior, skipping.
warning Pattern ("@oclif/command@1.4.2") is trying to unpack in the same destination "/Users/dickeyxxx/Library/Caches/Yarn/v1/npm-@oclif/command@1.4.2-f20ead470e87f629812ce21dd72b42c88517a5d" as pattern ("@oclif/command@1.4.2", "@oclif/command@1.4.1"). This could result in non-deterministic behavior, skipping.
warning Pattern ("@oclif/errors@1.0.2") is trying to unpack in the same destination "/Users/dickeyxxx/Library/Caches/Yarn/v1/npm-@oclif/errors@1.0.2-0a3f773d673a3cd8dc26bf2e3c49588af9fe3e" as pattern ("@oclif/errors@1.0.2"). This could result in non-deterministic behavior, skipping.
warning Pattern ("@oclif/plugin-help@1.1.6") is trying to unpack in the same destination "/Users/dickeyxxx/Library/Caches/Yarn/v1/npm-@oclif/plugin-help@1.1.6-2f92e5705ce656e26e75867af74bd3a3cc4bd607" as pattern ("@oclif/plugin-help@1.1.6"). This could result in non-deterministic behavior, skipping.
[4/5] Linking dependencies...
| Factor 7 | Prompt if you can |
Factor 7

Prompt if you can
Factor 7

Prompt if you can
Factor 8

Use tables
| Factor 8 | Use tables |
Factor 9  
Be Speedy

- <100ms: very fast (sadly, not feasible for scripting languages)
- 100ms–500ms: fast enough, aim here
- 500ms-2s: usable, but not going to impress anyone
- 2s+: languid, users will prefer to avoid your CLI at this point
Encourage Contributions

- Choose a License
- Code of Conduct
- Contribution Guidelines
- Plugins
Factor 11

Be Clear About Subcommands

$ git submodule add git@github.com:oclif/command

$ heroku domains:add www.myapp.com
Follow XDG spec

$XDG_DATA_HOME/myapp  ||  ~/.local/share/myapp
$XDG_CACHE_HOME/myapp  ||  ~/.cache/myapp
$XDG_CONFIG_HOME/myapp  ||  ~/.config/myapp