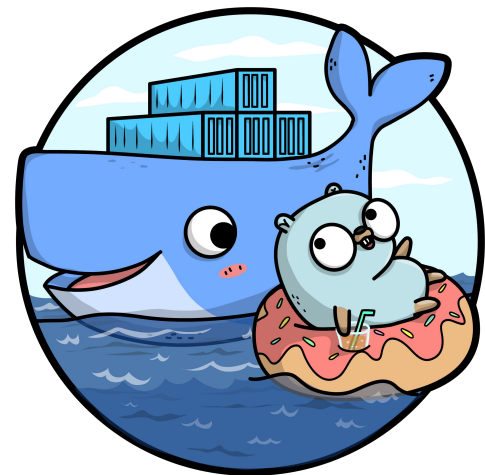


Distributing DevOps Tools for Fun and Profit!

Featuring: Go and Containers!

Lucy Davinhart
@lucydavinhart



What I'm Going To Say

- Where did this thing come from?
- Initial footsteps into containerising the tools
- New CLI made by gophers
- Adventures in open sourcing it!

The Past: Vagrant

Workstation VM, a big ol' Vagrant image

Content:

- ChefDK
- Docker
- Git
- Terraform
- Vault CLI
- AWS CLI
- Several other things



The Past: Vagrant

Problems:

- Supporting multiple versions of tools
- VMs are big!
- They take a long time to build
- And the startup script took a long time



The Past: Vagrant

Problems:

- Supporting multiple versions of tools
- VMs are big!
- They take a long time to build
- And the startup script took a long time
- So people very infrequently updated!



The Past: Docker run ...

Docker for Mac

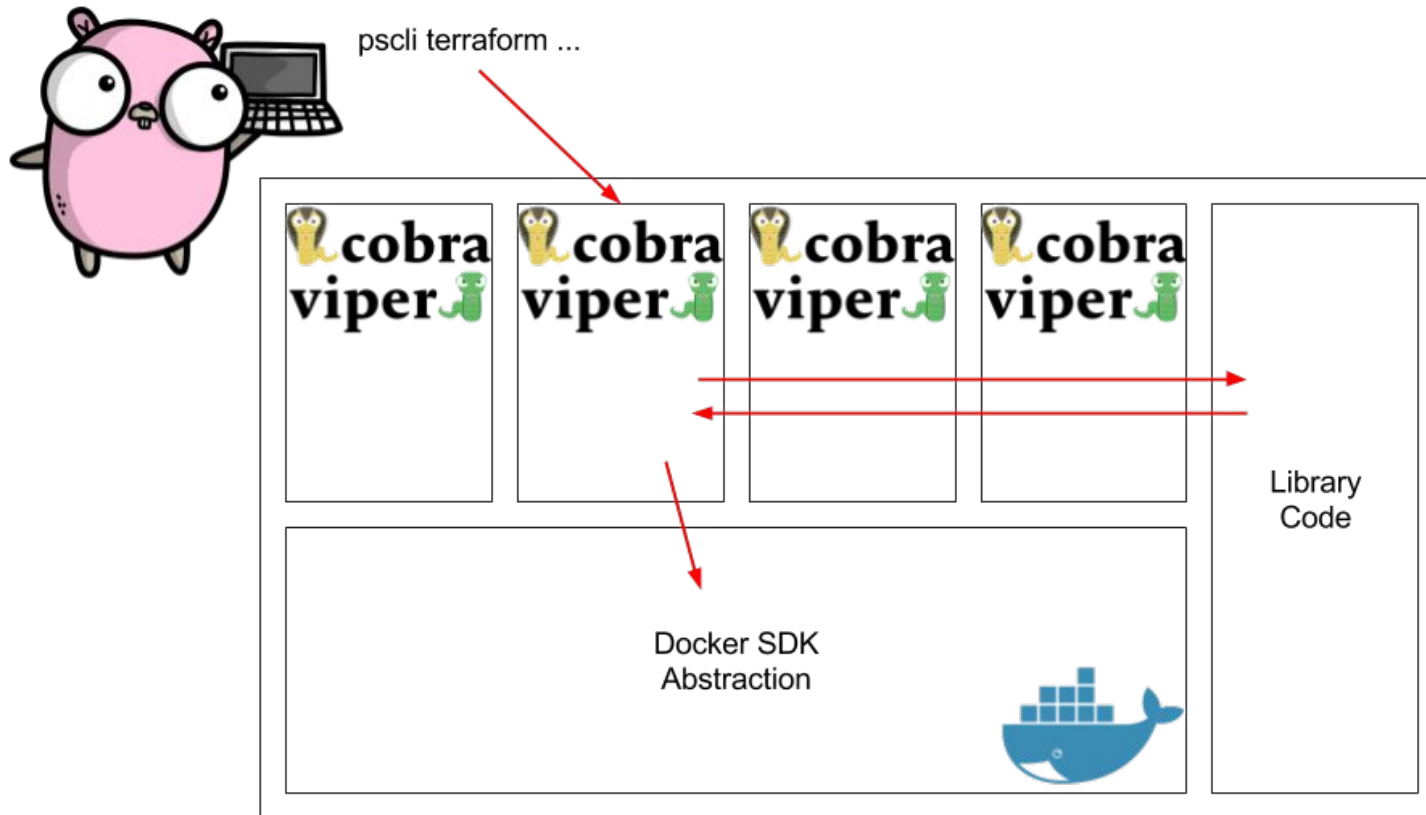


The Past: Docker run ...

```
alias buildtools="docker run --rm -it
  -v \${PWD}:/root/build
  -v /var/run/docker.sock:/var/run/docker.sock
  -v \${HOME}/.aws:/root/.aws
  -e \"BUILD_ROOT=\${PWD}\"
  -v ~/.gitconfig:/root/.gitconfig
  hub.fancycorp.io/illuminati/buildtools:stable"
```

The Present: PSCLI

Docker + Cobra/Viper + lots of code = pscli



The Present: PSCLI

Docker + Cobra/Viper + lots of code = pscli

- Self-updating
- Commands bind \$PWD or start with a git checkout
- Can run with local or remote docker
- Run arbitrary go code before launching a container, e.g.
 - Authenticate with Vault
 - Generate access credentials for AWS
 - Check if user has configured git, and prompt if not
- Some subcommands aren't even containers!
- Anonymous usage statistics



Demoing PSCLI

For those of you following the slides at home,
Asciinema recording of this demo:

<https://goo.gl/wqWPmH>



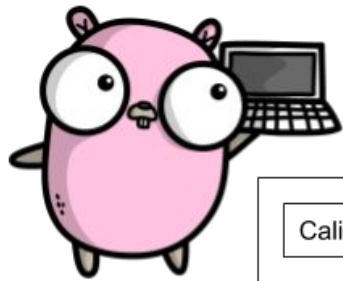
Open Source: Cali

Cali + less code = pscli?

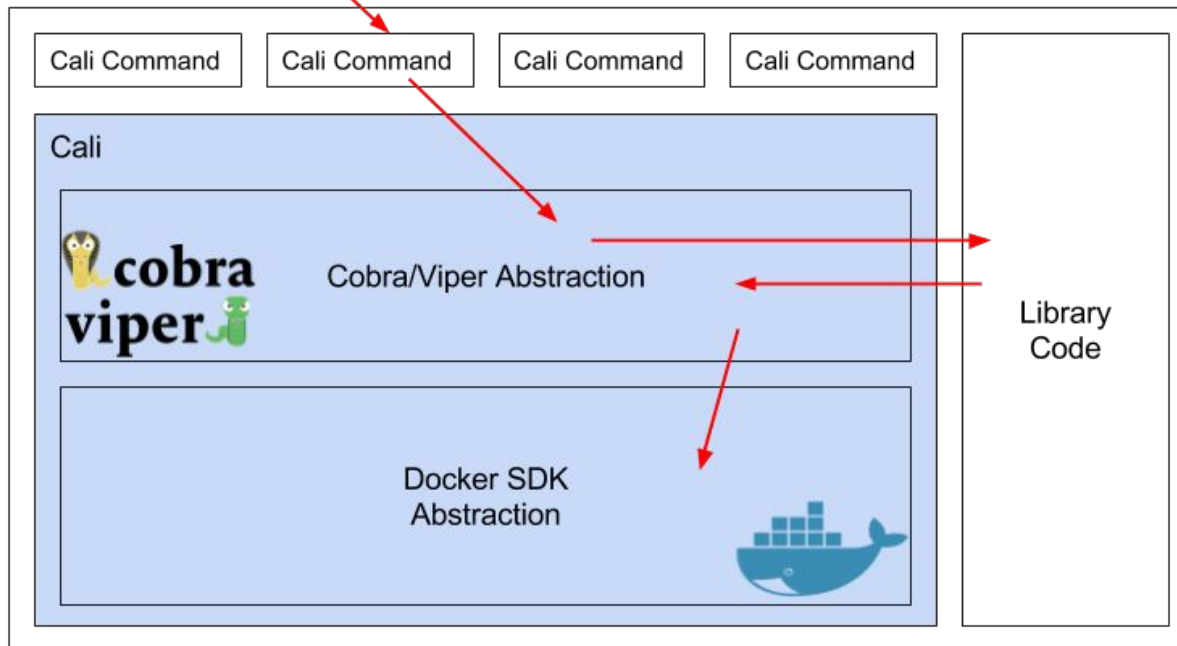


Open Source: Cali

Cali + less code = pscli?



cali-app terraform ...



Open Source: Cali

Cali + less code = pscli?

- Wrapper around Docker library and Cobra/Viper
- Sets sensible defaults
- Allows access to Docker and Cobra/Viper if needed



Demoing a Cali app

For those of you following the slides at home,
Asciinema recording of this demo:

<https://goo.gl/J2JpBR>



The Future

- Refactoring pscli to use Cali
- Experimenting with better ways to build these apps
- Figure out best practices for testing them
- Start releasing dev/beta versions
- Look into auto-updating where appropriate
- Cross-platform considerations
- More examples and documentation!



What I Said

The past:

Big VMs

The present:

Docker + Cobra/Viper + lots of code = pscli

The future:

Cali + less code = pscli (or your own tools here)

Links / Sources

Cali:

github.com/skybet/cali

LU(cy) CLI:

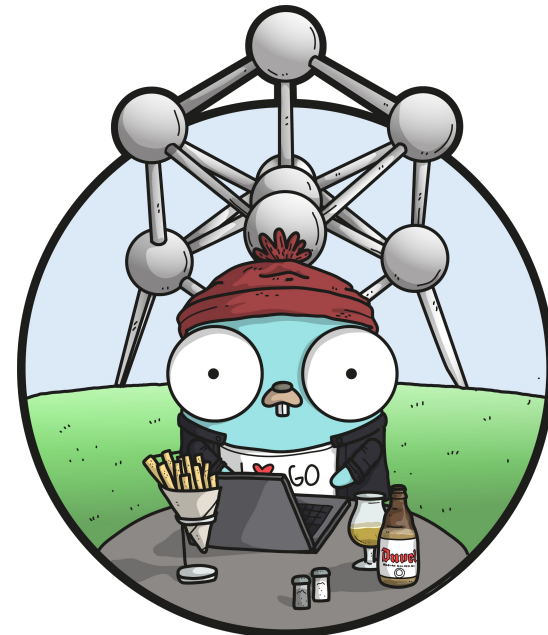
github.com/LMHD/luccli

Staticli (another tool we've made with Cali):

github.com/staticli/staticli | staticli.github.io

Gopher images: thanks to Ashley McNamara

github.com/ashleymcnamara/gophers



Questions?