KEYSIGHT

Coffee & Kubernetes

Angela Grigoroaia

Tools

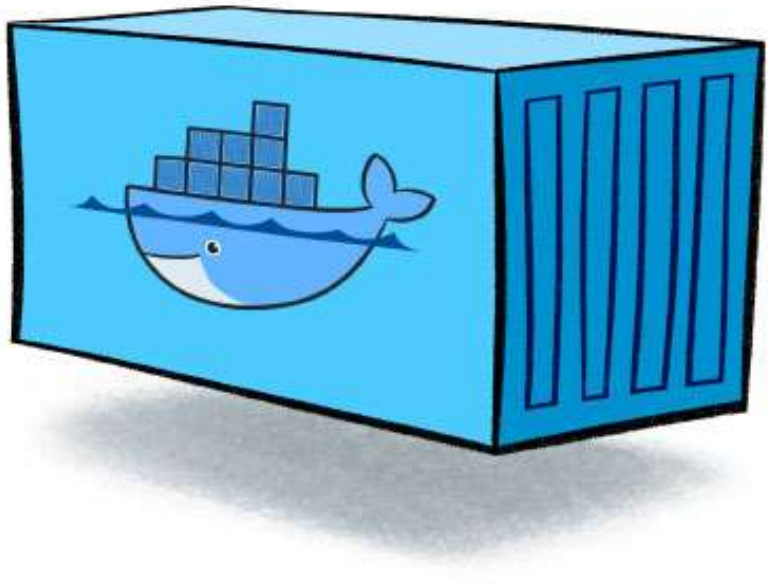
ready – set – go

- VSCode
- Rancher for Desktop (and WSL)
- Linux shell scripting
- Golang
- Make (for Windows)
- REST Client VSCode Extension
- kubectl
- K9s
- **Coffee**

Containers

you ship stuff in them

- **GOAL**
  Pack your code and all its dependencies in a bundle that is ready to be run **anywhere**.
- **ISSUE**
  Other apps can mess up your dependencies and shared resources (ex: TCP ports).
- **SOLUTION**
  A VM isolates your code at the cost of including a lot of redundant stuff (i.e. the whole OS).
- **BETTER SOLUTION**
  Isolate your code, its dependencies and resources. Reuse the common pieces of the underlying OS.



Kubernetes

the good Borg

- **GOAL**
  Monitor and manage running containers.
- **ISSUE**
  We love containers so much that everything is running in its own container now and we have too many to manage.
- **SOLUTION**
  Write code to automate starting, stopping and monitoring containers.
- **BETTER SOLUTION**
  Use industry standard container orchestration solutions (ex: Kubernetes).

Kubernetes Resources

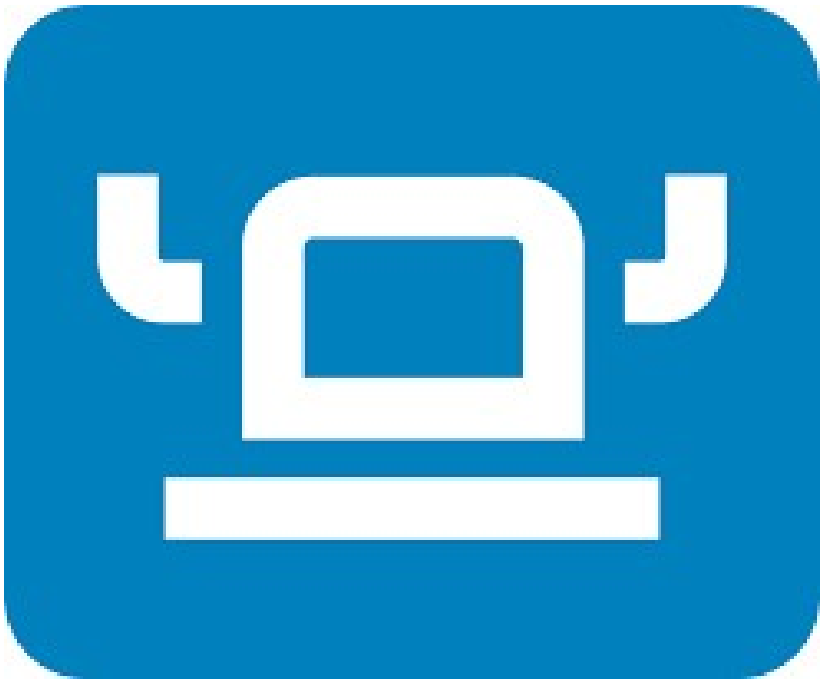
LEGO for building your app

- **CLUSTER**
  nodes, namespaces
- **WORKLOADS**
  pod, job, deployment
- **CONFIG & STORAGE**
  config map, secret, volume
- **SERVICE**
  service, ingress, endpoint
- **METADATA**
  event, custom resource definition

Rancher for Desktop

cluster 2go

- **GOAL**
  Get access to a cluster to try out things.
- **ISSUE**
  Configuring K8s from scratch is kinda hard when you're just getting started.
- **SOLUTION**
  Get someone else to do it for you.
- **BETTER SOLUTION**
  Get a someone who does this for a living to do it for you.



Go

cause you missed pointers … not

- **GOAL**
  Create containers that are as efficient as possible in terms of memory, startup time and response latency for external requests – at any scale.
- **ISSUE**
  All managed languages need to include the runtime in each container.
- **SOLUTION**
  Write unmanaged code and link it statically.
- **BETTER SOLUTION**
  Use the expertise of the K8s ecosystem and pick the language used by the majority.

Chaos Monkey

breaking stuff is ~~fun~~ useful

- **GOAL**
  Be resilient and recover from failure gracefully – without external intervention.
- **ISSUE**
  You can design and write your code with this goal in mind but **untested code doesn't work**.
- **SOLUTION**
  Write unit/integration tests to test the code that manages your recovery.
- **BETTER SOLUTION**
  Make failure a **normal** part of everyday operations in your cluster.

Prometheus

monitoring for dummies

- **GOAL**
  Minimize resources needed to serve all the current users at any given moment.
- **ISSUE**
  It’s hard to know how many users will need to be served at a future point and manually reacting to changes in usage patterns is slow.
- **SOLUTION**
  Write your own scripts to do monitoring and automated scaling.
- **BETTER SOLUTION**
  Use an existing solution that does it for you.

Operators

this is where you need the coffee

- **GOAL**
  You want your customers to easily deploy your application within their clusters.
- **ISSUE**
  Deployment via YAMLs requires a thorough understanding of the internals of the application.
- **SOLUTION**
  Create scripts that automate the deployment.
- **BETTER SOLUTION**
  Write code that manages every aspect of your application in a cluster.

KEYSIGHT

Thank you

KEYSIGHT