An intro to ko, developing the knative way

Andrea Frittoli
Developer Advocate
andrea.frittoli@uk.ibm.com
@blackchip76

IBM Cloud

Microservice(s) in Go

From local...

- A few lines of code
- Build and run locally

```
package main

import (
  "flag"
  "fmt"
  "net/http"
)

func main() {
  hwPort := flag.Int("port", 8080, "Listening port
       numbers")
  flag.Parse()

  http.HandleFunc("/", func(w http.ResponseWriter,
       r *http.Request) {
    fmt.Fprintf(w, "{\"hello\": \"%s\"}", r.URL.
      Path)
  })

  http.ListenAndServe(fmt.Sprintf(":%d", *hwPort),
       nil)
}
```

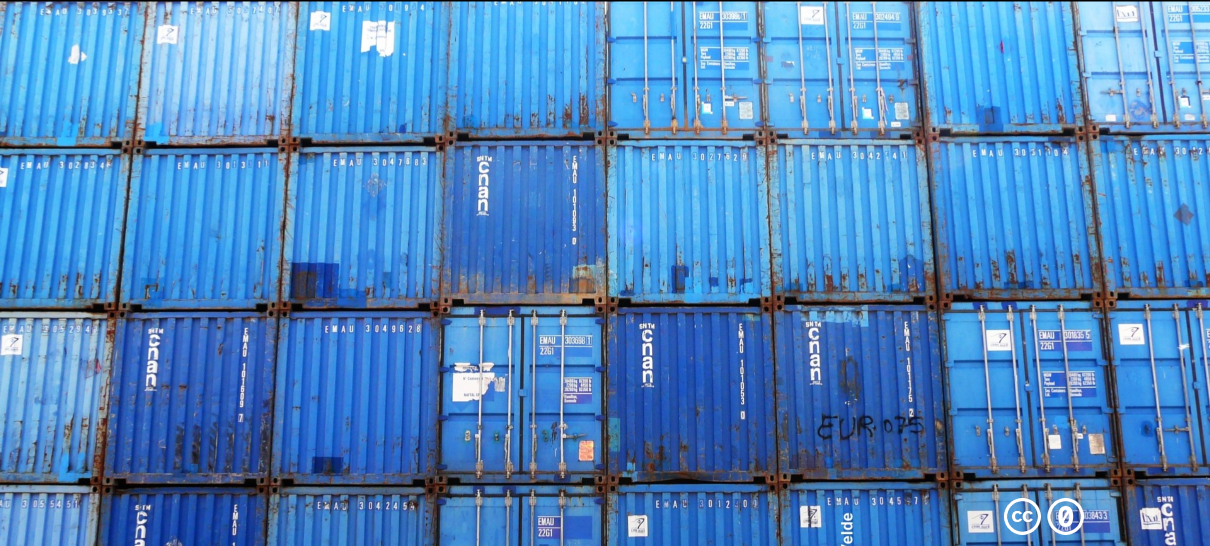
```
#!/bin/bash

go build
./helloworld --port 8080 &
```

…to the Cloud

- Scaling
- Resiliency
- Security

Something is missing to get from A to B



Define the container to build
Define the container to run

- One Dockerfile for both

```
# Start by building the
     application.
FROM golang:1.8 as build

WORKDIR /go/src/github.
     com/afrittoli/
     go_helloworld
ADD . /go/src/github.com
     /afrittoli/
     go_helloworld

RUN go-wrapper download
       # "go get -d -v
     ./..."
RUN go-wrapper install

# Now copy it into our
     base image.
FROM gcr.io/distroless/
     base
COPY --from=build /go/
     bin/go_helloworld
     /helloworld
CMD ["/helloworld"]
```

Build the image
Tag the image
Push the image to the registry
Update the image version

- Kubernetes manifests
- Helm chart values

```
#!/bin/bash

# Define variables
TAG=$(git log -1 --
     pretty=%H)
REGISTRY=registry.ng.
     bluemix.net/
     knative

# Build and push the
     image
docker build .
docker tag ${REGISTRY}/
     go_helloworld:${
     TAG}
docker push ${REGISTRY}/
     go_helloworld:${
     TAG}
```

Getting Started with Ko

Installing

Container Registry

Publish, Resolve, Apply, Delete

Publish

Resolve

Apply & Delete

Ko and Knative

Developing Knative

Knative & Ko

Q&A

Thank You! Questions?

Andrea Frittoli
Developer Advocate
andrea.frittoli@uk.ibm.com
@blackchip76

—