從魷魚遊戲看容器開發

Shawn Ho, Kubernetes Evangelist

Google Cloud

魷魚遊戲得要跨過六關才能拿到456億

201

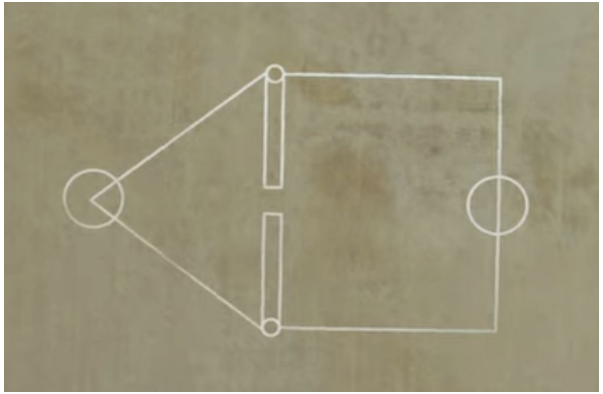
108

40

17

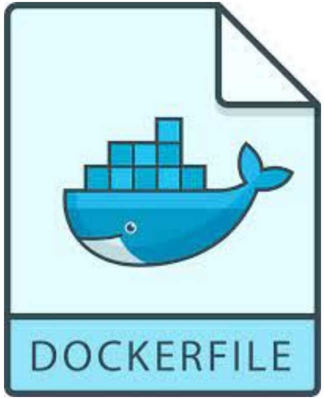
3

1



K8S對於開發者呢？

201



108

docker
REGISTRY
TM & © 2015 Docker, Inc.

40

17

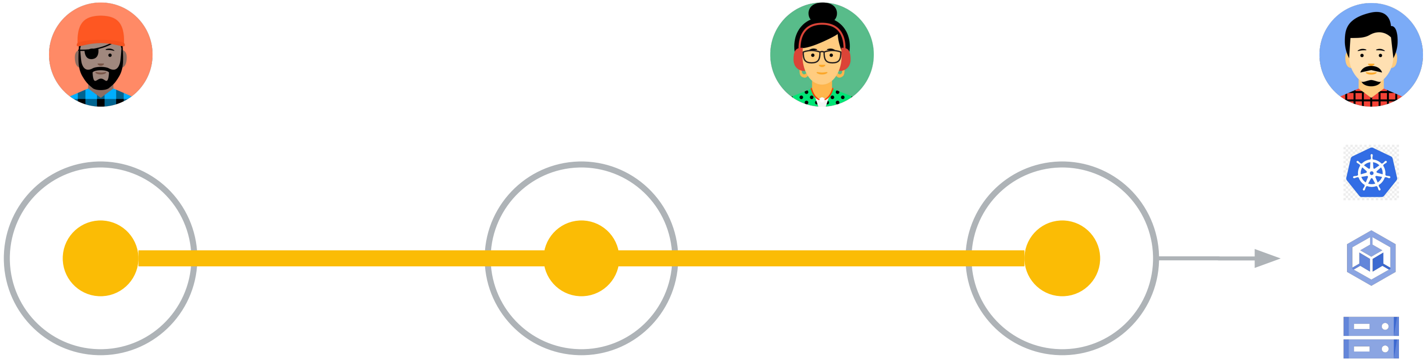
3

1

호떡 놀이
糖饼游戏
00:12
천천히 하세요!
6+ 호떡 놀이
O.102 78PCS
THE PICTURE IS FOR REFERENCE ONLY PLEASE REFER TO THE REAL OBJECT
천천
369

Golden Path

Unification, Insulation, Flexible Responsibility



Dev Inner Loop

Code

Try

Commit

CI

Build

Test

Security

CD

Promote

Roll-out

Roll-back

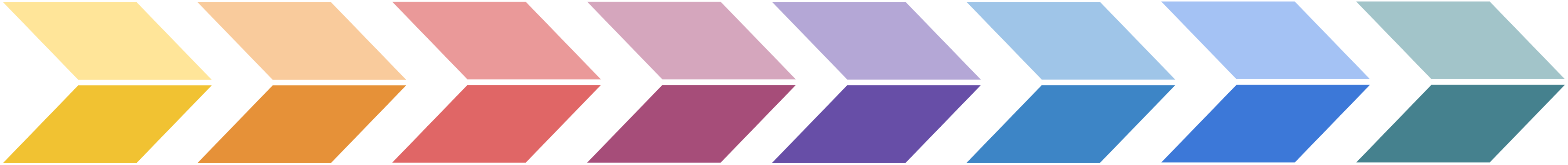
Metrics

Developer Challenges

- Many tools
- Long feedback loops between changes
- Tedious manual steps and ordering
- Development across services and repositories becomes challenging

```
$ docker build

$ docker tag

$ docker push

$ vim deploy.yaml

$ kubectl apply -f deploy.yaml

$ kubectl logs

$ kubectl port-forward
```

Kubernetes Development Lifecycle



Build

Build container images for each artifact

Tag

Ensure the built items are identifiable

Push

Push images to the registry

Test

Verify the built images

Render manifests

Replace image references in manifests with the pushed images

Deploy Manifests

Deploy resources to cluster and wait for them to become healthy

Forward Ports and Logs

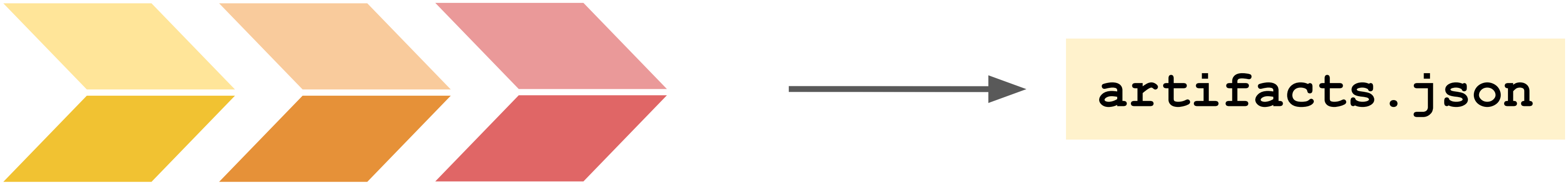
Forward notable ports and ensure logs are streaming to console

Watch for updates

When source files change, rebuild and redeploy app

$ skaffold build

CI



artifacts.json

Build

Build container images for each artifact

Tag

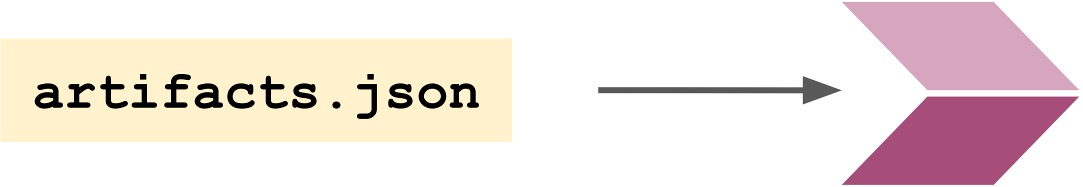
Ensure the built items are identifiable

Push

Push images to the registry

```
$ skaffold test
```

CI



artifacts.json

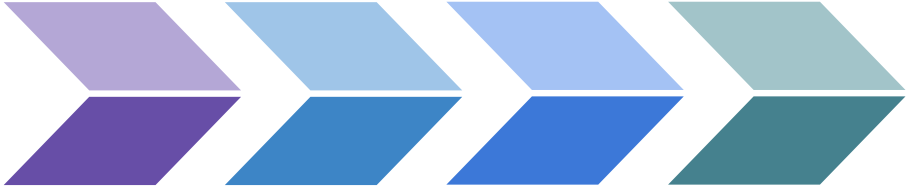
Test

Verify the
built images

```
$ skaffold deploy
```

CD

```
artifacts.json
```



Render manifests

Replace image references in manifests with the pushed image

Deploy Manifests

Deploy manifests to cluster and wait for them to become healthy

Forward Ports and Logs

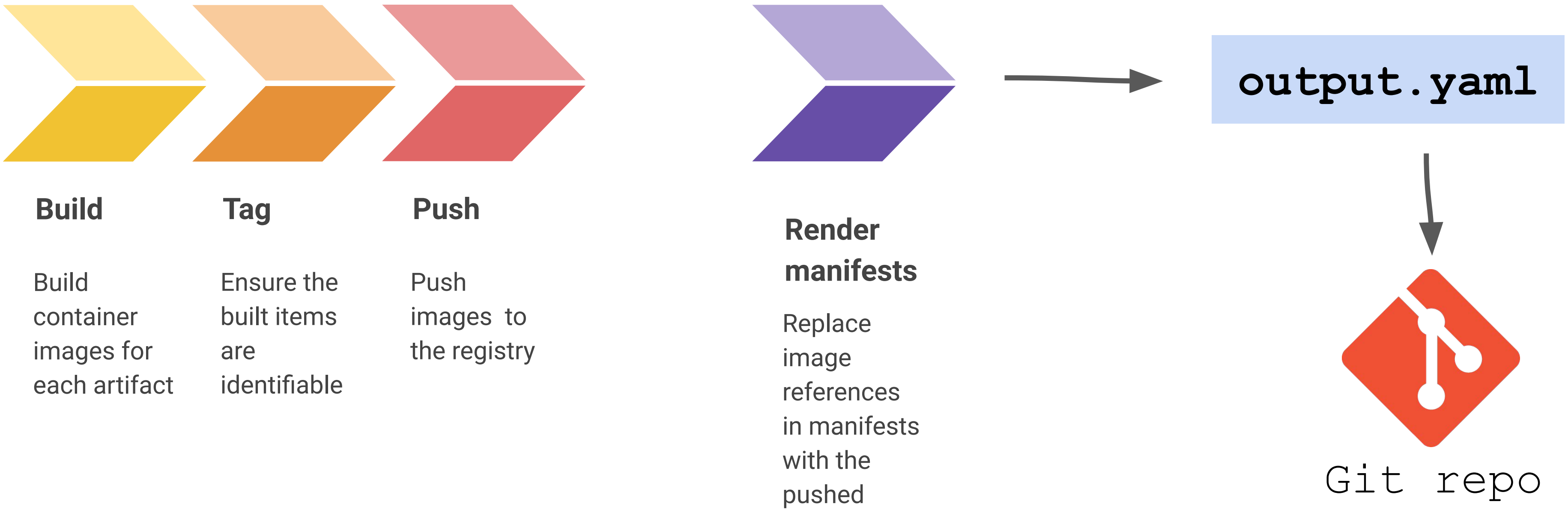
Forward notable ports and ensure logs are streaming to console

Watch for updates

If source files change, rebuild and redeploy app

```
$ skaffold render
```

CI



Build

Build container images for each artifact

Tag

Ensure the built items are identifiable

Push

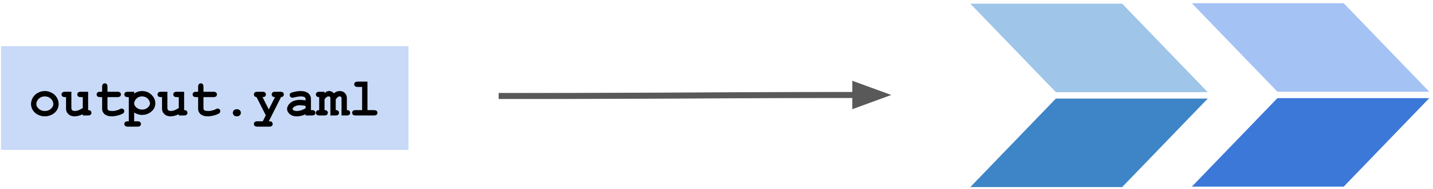
Push images to the registry

Render manifests

Replace image references in manifests with the pushed image

```
$ skaffold apply
```

CD



```
output.yaml
```

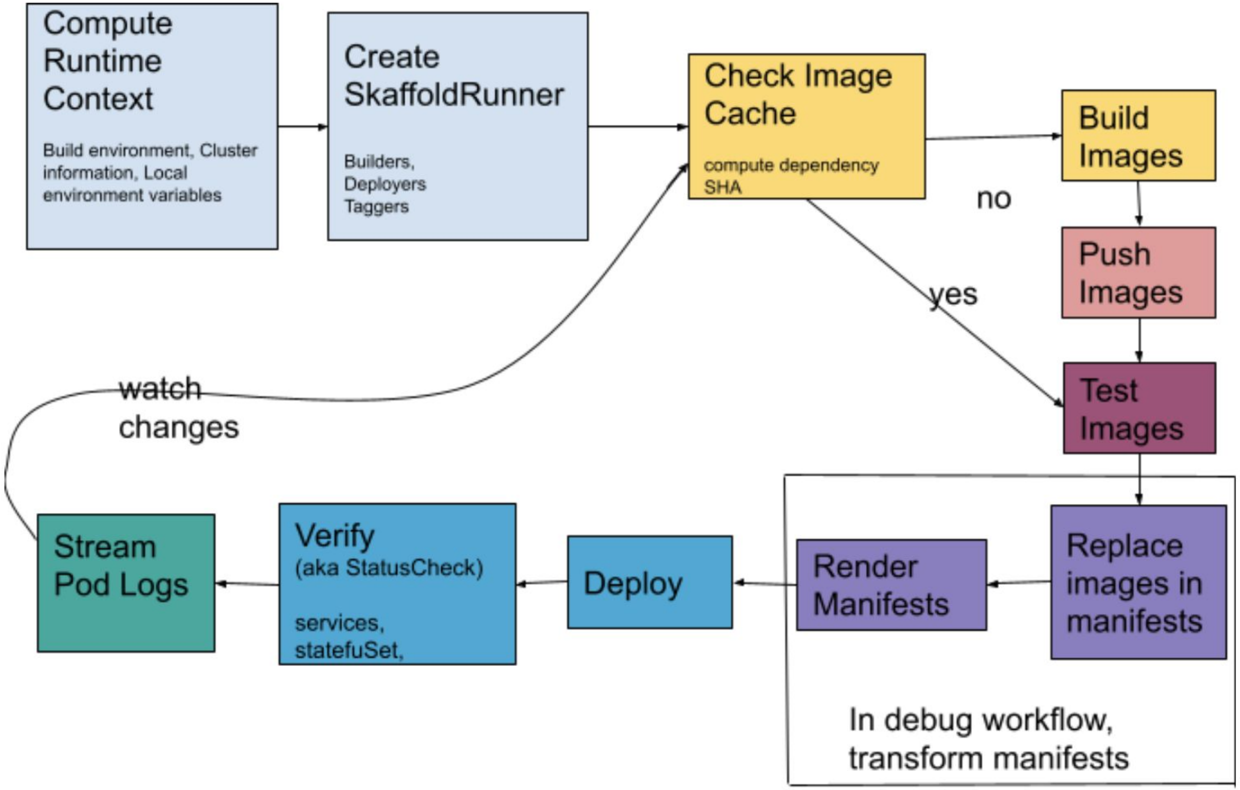
Deploy Manifests

Deploy manifests to cluster and wait for them to become healthy

Forward Ports and Logs

Forward notable ports and ensure logs are streaming to console

$ skaffold dev



`$ skaffold debug`

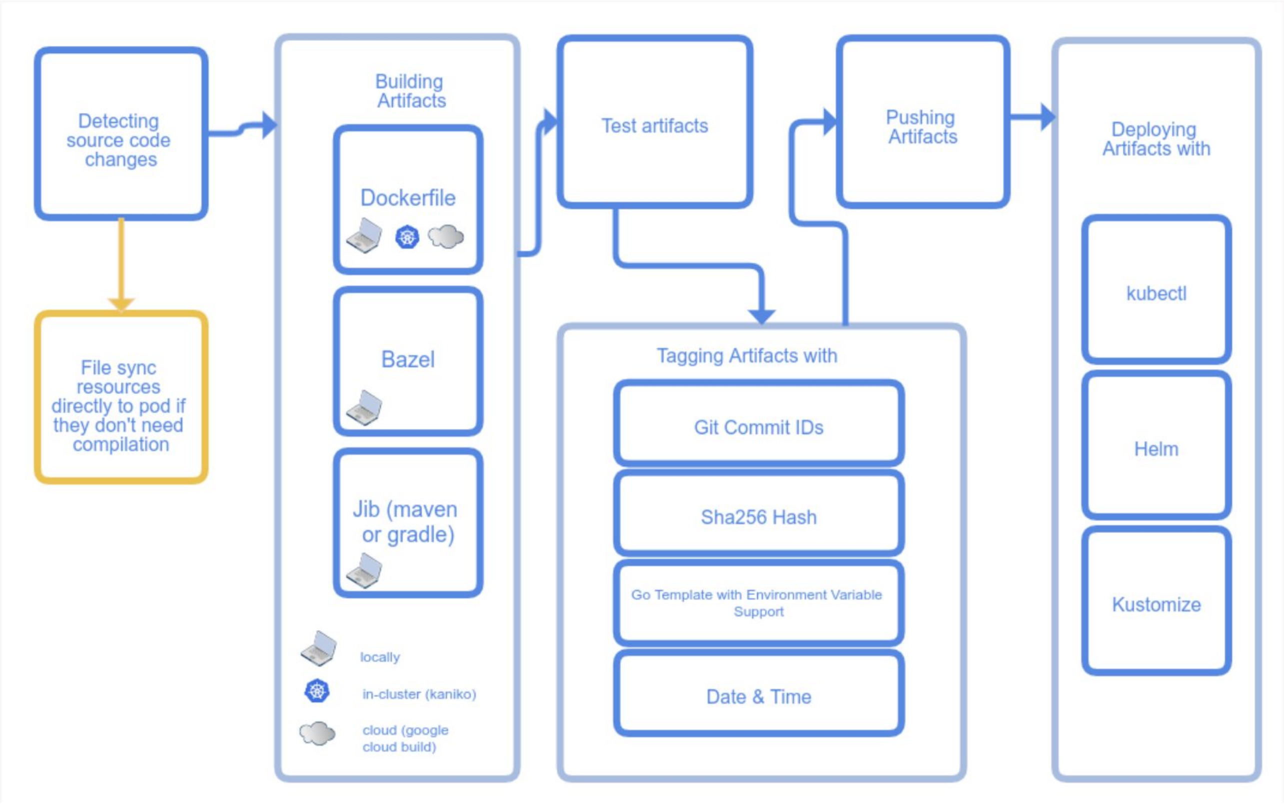
Acts like `skaffold dev`, but it additionally configures debuggers.

Debugging ports are exposed and forwarded to the local machine, allowing developers to debug remotely deployed applications from their host machines.

Supported Languages :

- Go
- NodeJS
- Python
- Java
- .NET Core

Skaffold:


Detecting source code changes
File sync resources directly to pod if they don't need compilation
Building Artifacts
Dockerfile
Bazel
Jib (maven or gradle)
locally
in-cluster (kaniko)
cloud (google cloud build)
Test artifacts
Tagging Artifacts with
Git Commit IDs
Sha256 Hash
Go Template with Environment Variable Support
Date & Time
Pushing Artifacts
Deploying Artifacts with
kubectl
Helm
Kustomize