

An aerial photograph of a large container yard filled with rows of shipping containers. A semi-transparent blue horizontal band is overlaid across the middle of the image. The text "Demystifying Containers" is written in white, sans-serif font across this band. A thin white horizontal line is positioned directly beneath the text.

Demystifying Containers

About Me



Kunal Babre

Sr. Cloud Solution Architect,
EMEA Strategic Global Downstream at Microsoft





Agenda

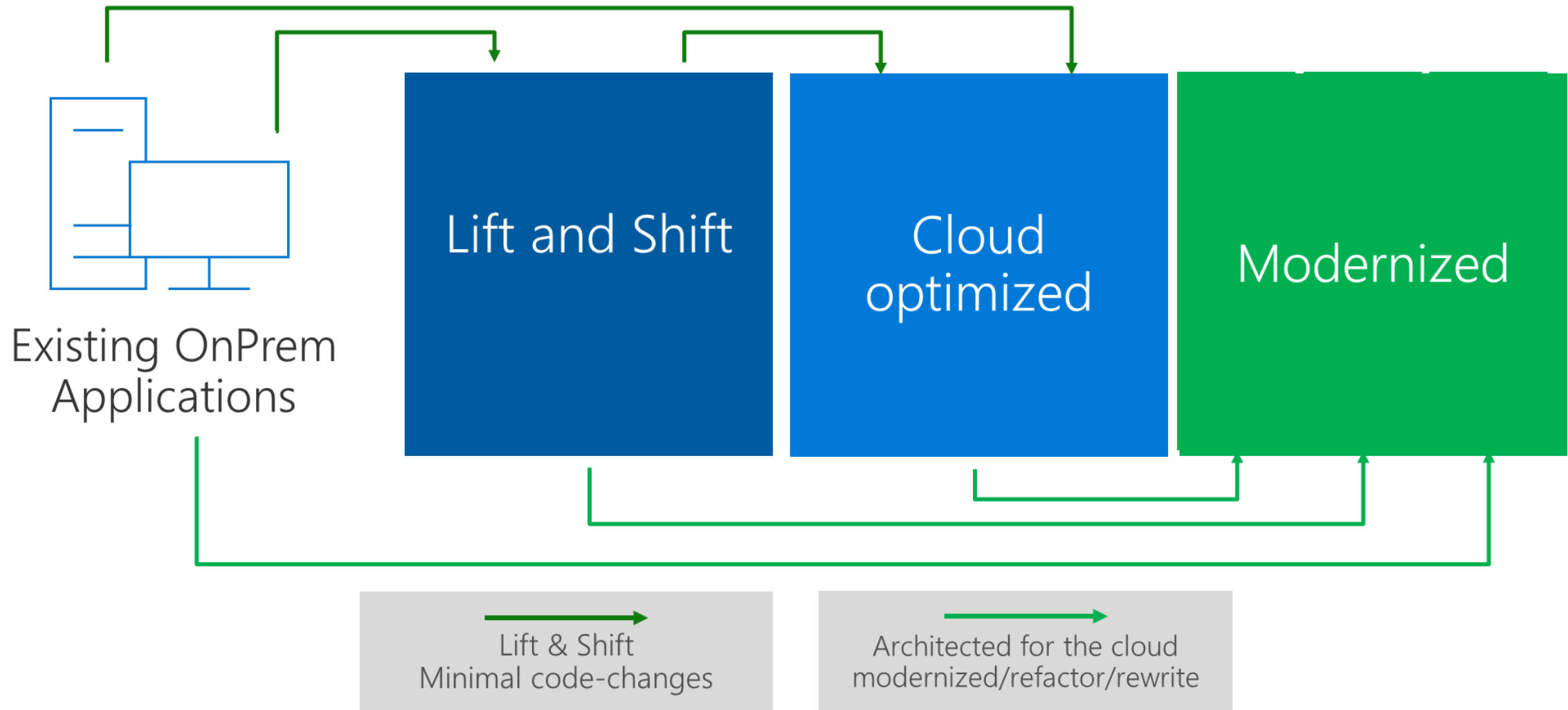
1. Containers Overview
2. Container Orchestration
3. Containers on Azure



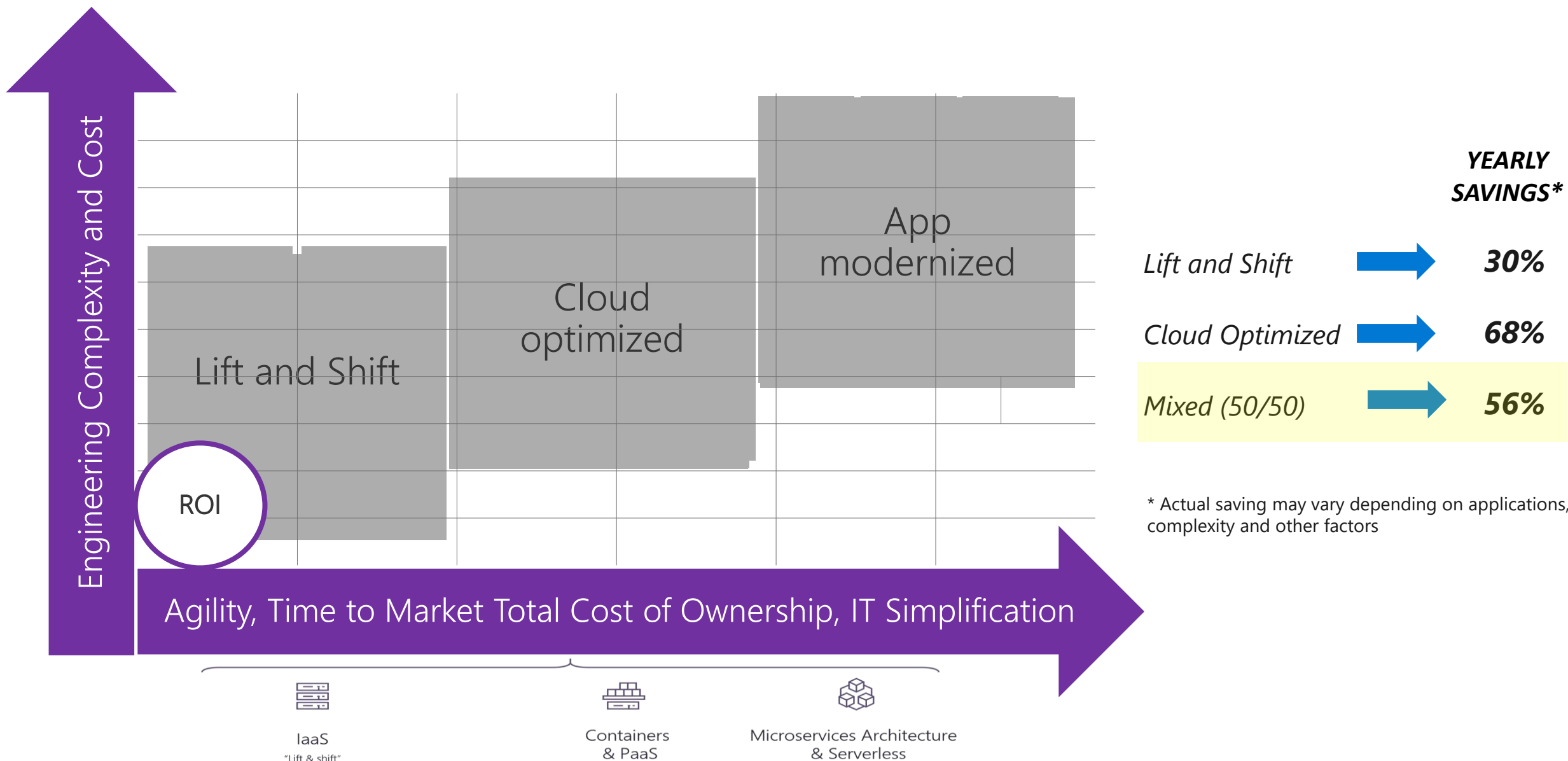
Overview

Gartner predicts that by 2022, more than 75% of global organizations will be running containerized applications in production.

Cloud Maturity Model



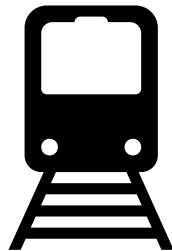
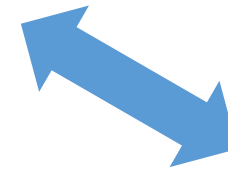
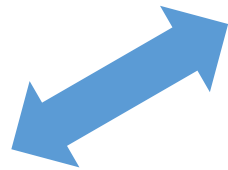
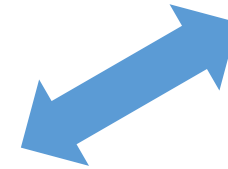
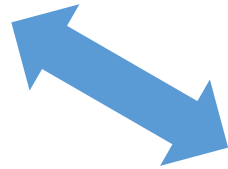
Cloud Maturity Model - Benefits



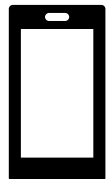
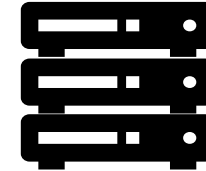
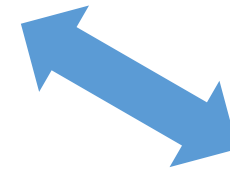
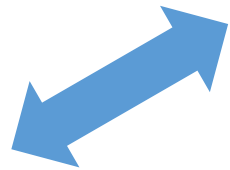
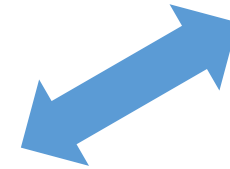
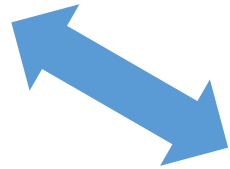
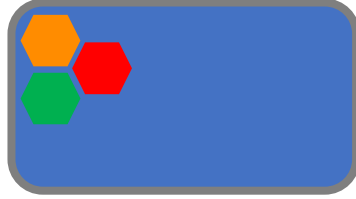
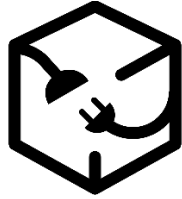
An aerial photograph of a large container yard, showing rows of stacked shipping containers in various colors. A semi-transparent blue horizontal band is overlaid across the middle of the image. The word "Container?" is written in white, sans-serif font within this band, underlined with a white wavy line.

Container?

What is a Container?



What is a Container?



Consistent Environment

Run Anywhere

Isolation

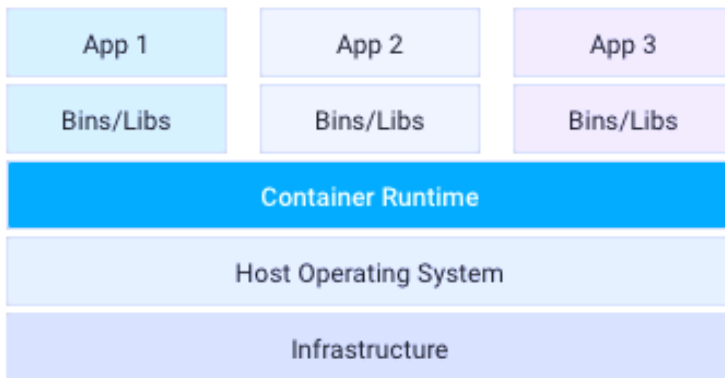
Containers vs Virtual Machines?

Containers

Shared host OS kernel

Portability

Faster scalability

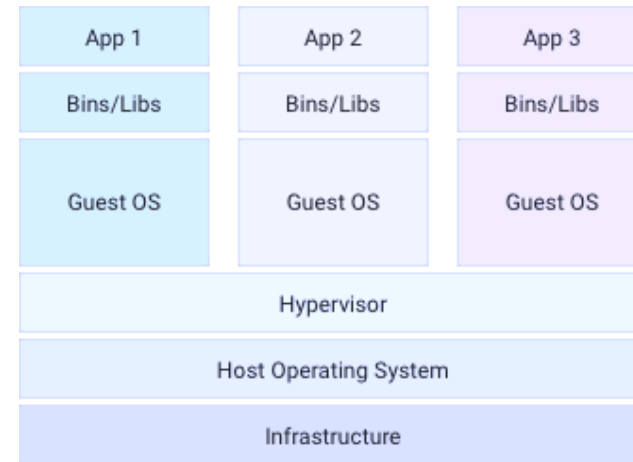


Virtual machines

Separate OS per instance

Large footprint

Slower startup



Three things about **Container**

1.Runtime Isolation

Namespaces

- Mount
- Network

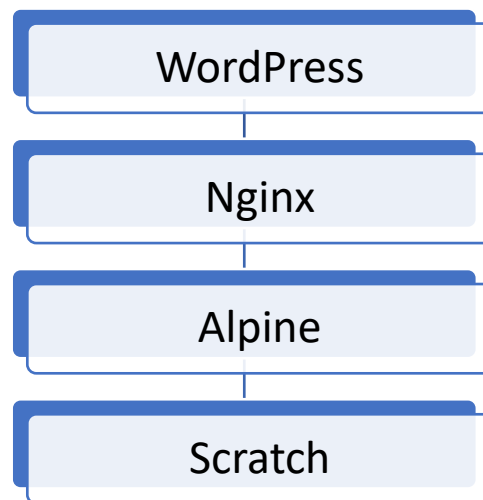
Cgroups

- Memory
- CPU

2.Image

Layer Capability

- Union Filesystem



3.Container Engines



Definition Image - Dockerfile



```
FROM          mcr.microsoft.com/dotnet/aspnet:5.0

WORKDIR       /app

COPY          Contoso/publish .

EXPOSE        80

ENTRYPOINT    ["dotnet", "Contoso.dll"]
```


Docker

Commands



Build

Build an image from the Dockerfile in the current directory and tag the image

```
docker build -t myimage:1.0 .
```

List all images that are locally stored with the Docker Engine

```
docker image ls
```

Delete an image from the local image store

```
docker image rm alpine:3.4
```



Share

Pull an image from a registry

```
docker pull myimage:1.0
```

Retag a local image with a new image name and tag

```
docker tag myimage:1.0 myrepo/myimage:2.0
```

Push an image to a registry

```
docker push myrepo/myimage:2.0
```



Run

Run a container from the Alpine version 3.9 image, name the running container "web" and expose port 5000 externally, mapped to port 80 inside the container.

```
docker container run --name web -p 5000:80 alpine:3.9
```

Stop a running container through SIGTERM

```
docker container stop web
```

Stop a running container through SIGKILL

```
docker container kill web
```

List the networks

```
docker network ls
```

List the running containers (add `--all` to include stopped containers)

```
docker container ls
```

Delete all running and stopped containers

```
docker container rm -f $(docker ps -aq)
```

Print the last 100 lines of a container's logs

```
docker container logs --tail 100 web
```



Docker Demo

Run Commands

Run latest version of nginx in detached mode, name the running container “web” and expose port 8080 externally, mapped to port 80 inside the container.

```
$ docker run --name web -d -p 8080:80 nginx:latest
```

Stop a running container (SIGTERM or use **kill** to terminate immediately SIGKILL)

```
$ docker stop web
```

List the running containers including stopped ones (or docker ps --all)

```
$ docker ls --all
```

Delete containers

```
$ docker rm -f web
```


Container Registry

Repositories of built container

Public

- hub.docker.com
- ...

Private

- **Azure Container Registry (Azure ACR)**
- Amazon Elastic Container Registry (Amazon ECR)
- Google Container Registry
- Docker
- ...

Container Registry Demo

Docker Hub & ACR

1. Create new dotnet MVC App (in folder demo)

```
$ dotnet new mvc
```

2. Add Docker File

```
Dockerfile
```

3. Build

```
$ docker build -t kunalbabre/demo:latest .
```

4. Run

```
$ docker run --name web -d -p 8080:80 kunalbabre/demo:latest
```

5. Publish (make sure you have repo created)

```
$ docker push kunalbabre/demo:latest
```

An aerial photograph of a large container yard, showing rows of stacked shipping containers in various colors. A semi-transparent blue horizontal band is overlaid across the middle of the image, serving as a background for the title text.

Container Orchestrators

Container Orchestrators

- Kubernetes
- Docker Swarm
- Service Fabric
- Mesos DC/OS
- Google Borg (proprietary)
- And more...



Self-healing

Load balancing

Scheduling

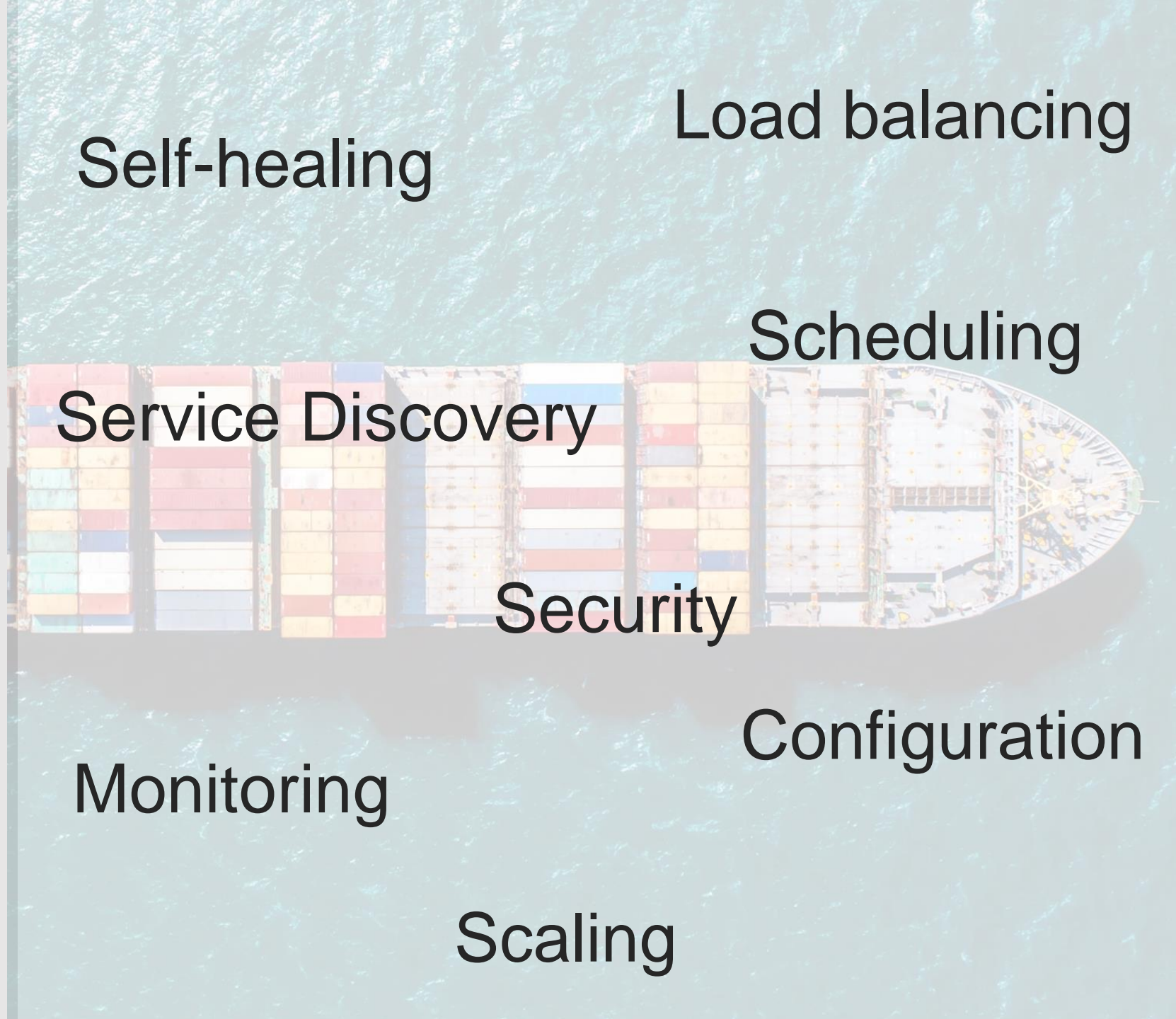
Service Discovery

Security

Configuration

Monitoring

Scaling



What is **Kubernetes**?

Open-source container orchestrator. Designed to **automate deployment, scaling, and management** of applications.



Portable

Public, private, hybrid,
multi-cloud



Extensible

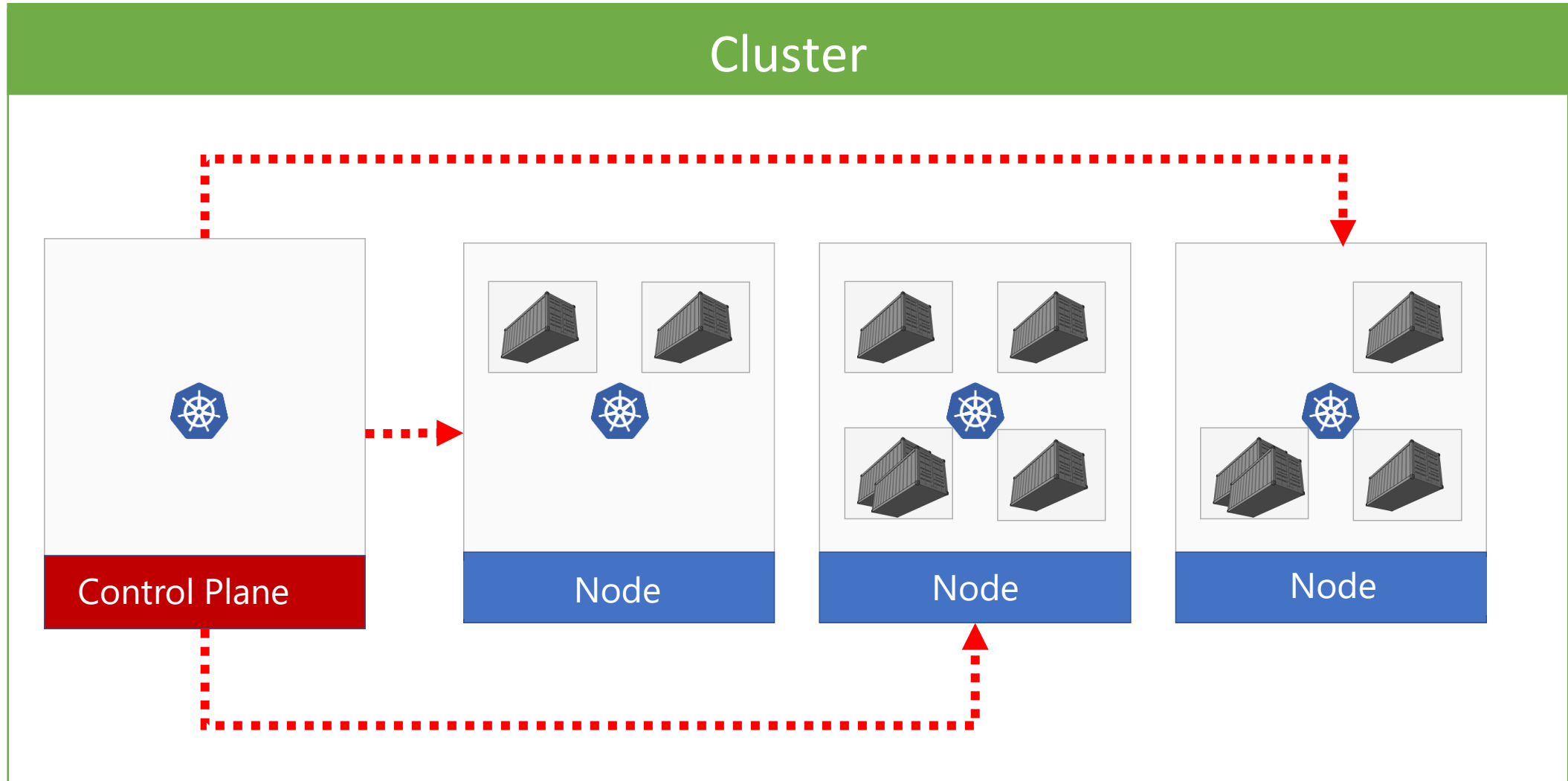
Modular, pluggable, hook
able, composable



Self-healing

Auto-placement, auto-restart,
auto-replication, auto-scaling

Kubernetes Architecture



An aerial photograph of a large container yard, showing numerous shipping containers stacked in neat rows. A semi-transparent blue horizontal band is overlaid across the middle of the image. The text 'Container on Azure' is written in white, sans-serif font across this band. A thin white horizontal line is positioned directly beneath the text.

Container on Azure

Running Containers in Azure



Azure Kubernetes Service (AKS)



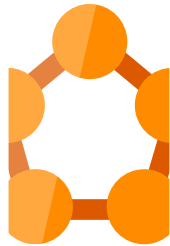
App Service
(WebApp & Functions)



Container Instances



Batch



Service Fabric



Virtual Machine



Azure Red Hat OpenShift



Google Search

I'm Feeling Lucky

1. Search **SuperMario Docker**
2. Note Image Tag
3. Port Number
4. Deploy in ACI
Azure container
Instance
5. Play the game

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Microsoft.ContainerInstances-20210418125759 | Overview

Deployment

Search (Ctrl+/) Delete Cancel Redeploy Refresh

Overview

Inputs

Outputs

Template

We'd love your feedback! →

Deployment is in progress

Deployment name: Microsoft.ContainerInstances-20210418125759 Start time: 18/04/2021, 12:58:56
Subscription: InternalConsumption_1 Correlation ID: ccd27080-49dd-4510-810b-d7da05bfa46f
Resource group: AzureDemo

Deployment details (Download)

Resource	Type	Status	Operation details
superdemokb	Microsoft.ContainerInstanc...	Created	Operation details

Security Center
Secure your app
Go to Azure sec

Free Microsoft
Start learning to

Work with an e
Azure experts a
who can help m
and be your fir
Find an Azure e

1. Search
SuperMario Docker

2. Note Image Tag
pengbai/docker-
supermario:latest

3. Port Number
8080

4. Deploy in ACI
Azure container
Instance

5. Play the game

Azure Container Instance Demo



Thank You
