

# Getting Started with EKS (Elastic Kubernetes Service)

---

Running Your Application on EKS



**Craig Golightly**  
Senior Software Consultant

@seethatgo [www.seethatgo.com](http://www.seethatgo.com)



# Identify Your Use Case



Amazon Elastic Kubernetes Service

## Common components

- Http server
- Web / mobile APIs
- Business logic
- Workers

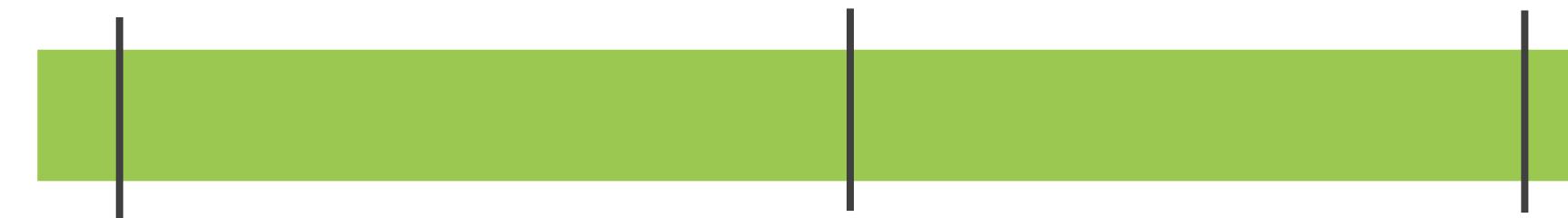
Orchestrating multiple containers

Scaling containers up and down



# When to Start

**Containerized application  
NOT using Kubernetes**



**New (greenfield)**

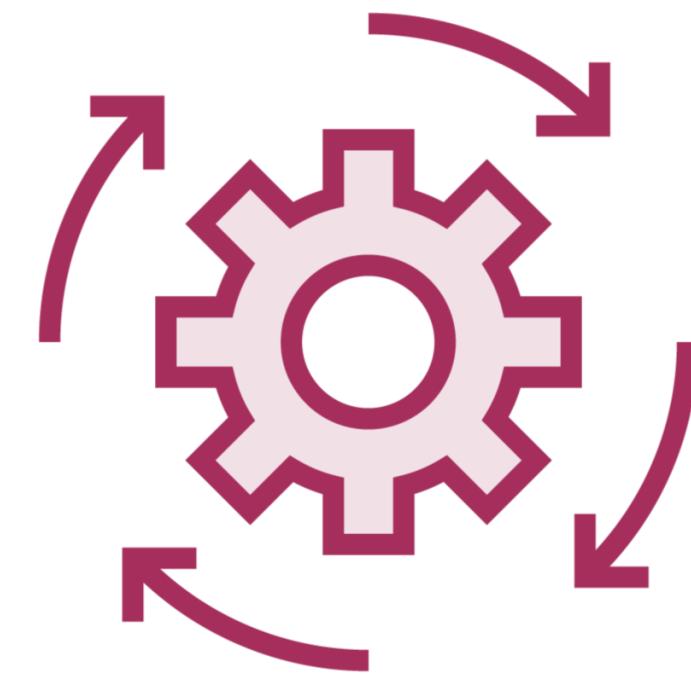
**Containerize  
existing application**



# Designed for Failure



**Pods are expendable**  
**Create a new one**



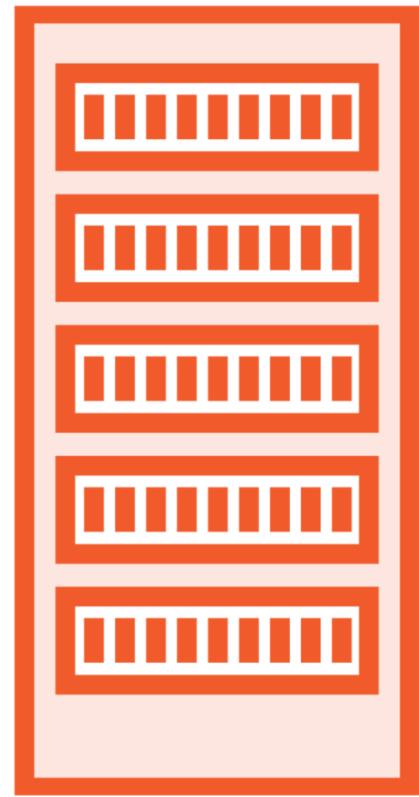
**Automated recovery**  
**Architect to scale**



**Indispensable apps**  
**Emergency fire drill**  
**Manual intervention**



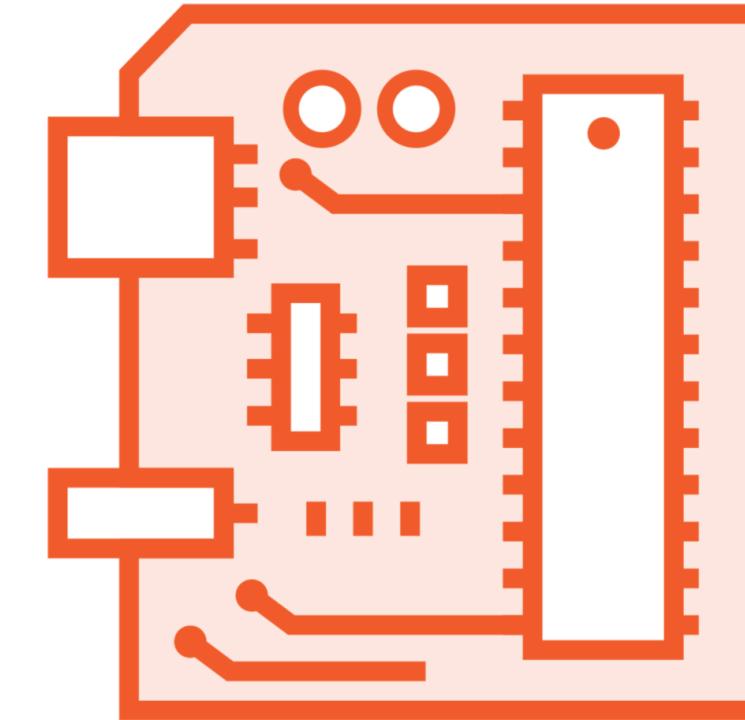
# Application Hardware Requirements



## Commodity hardware

Change CPU, RAM, disk

Some GPU access



## Specialized hardware

Verify supported node types

Kubernetes driver support



# Automating the Process

# GitHub

**GitHub Actions**

**GitHub flow**

- Branching
- Pull requests
- Merging

**Natural fit into existing process**

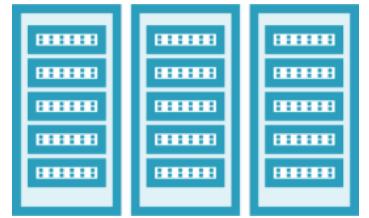
- Consider using Git



# Amazon Web Services



**Physical building, power, cooling**



**Machines, network infrastructure**



**Physical and virtual security**



**Staff to monitor and maintain**

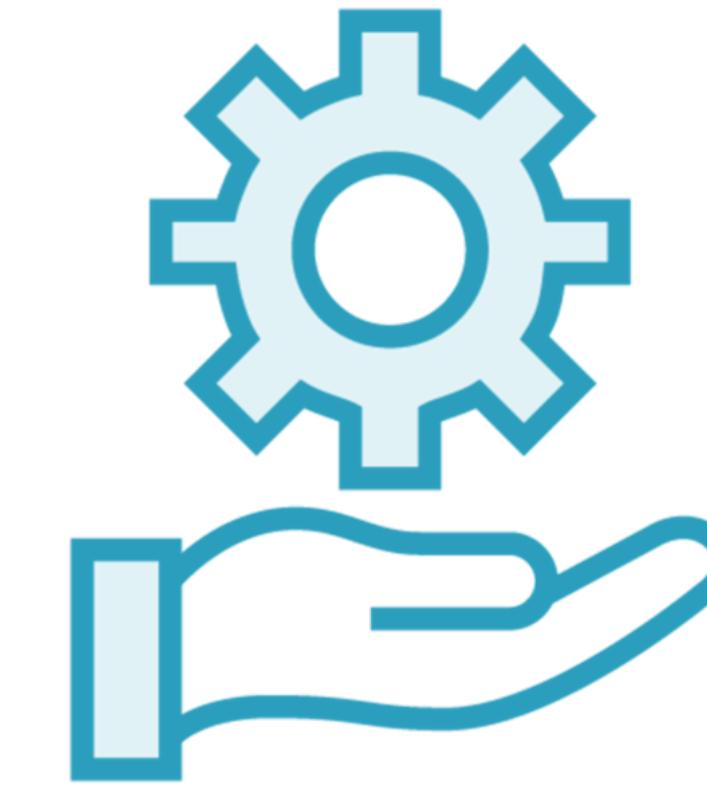


# Kubernetes in the Cloud



## Kubernetes on Cloud Machines

**Set up and manage the  
Kubernetes infrastructure yourself**



## Managed Kubernetes

**Cloud provider manages  
Kubernetes infrastructure  
You simply use it**



# Why Managed Kubernetes?



**Speed**



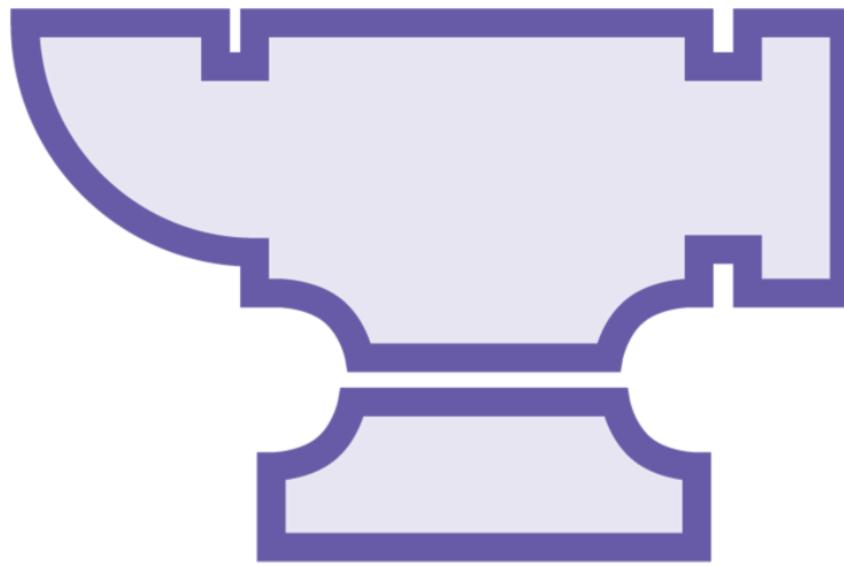
**Simplicity**



**Scale**



# Providing Value



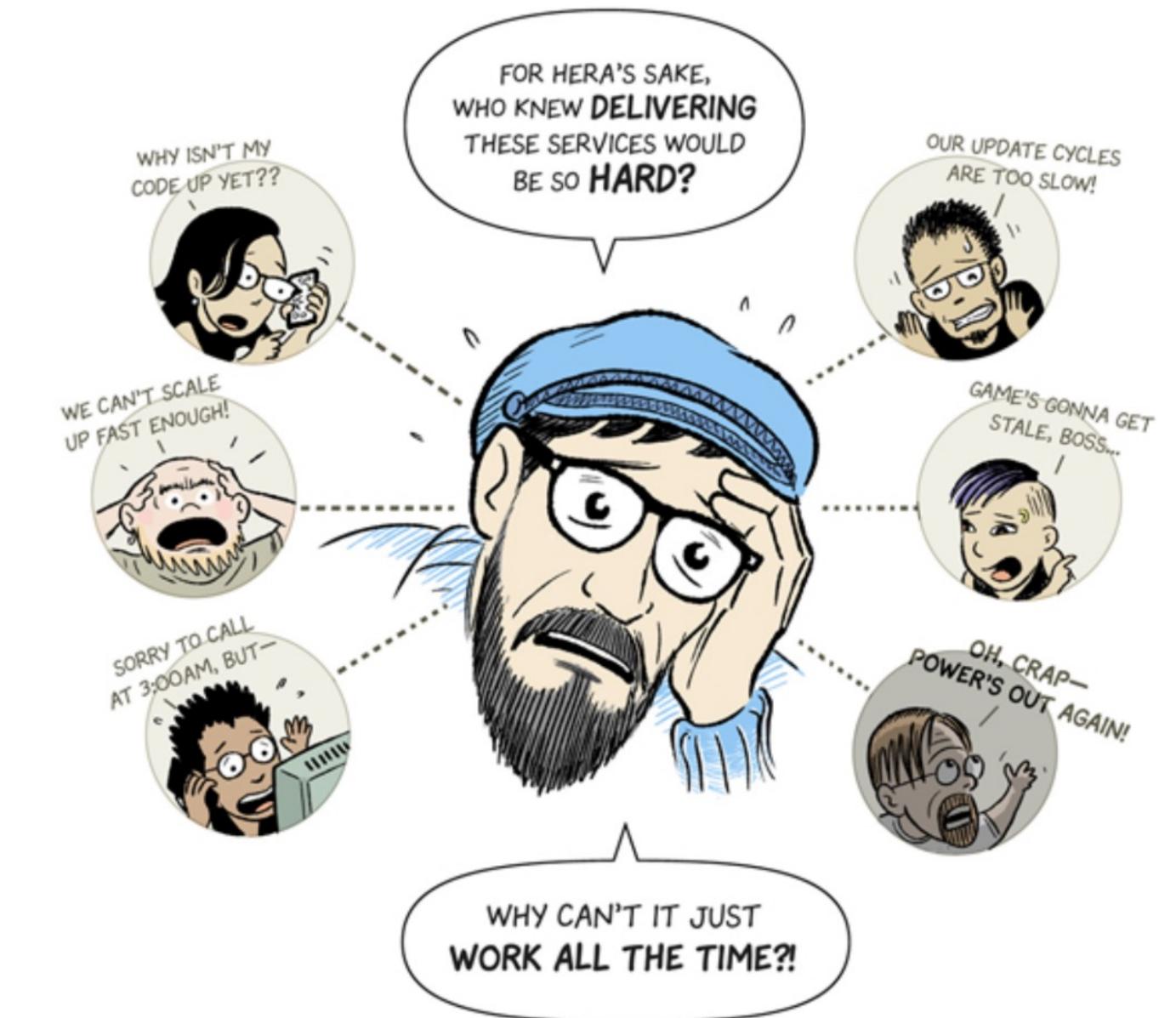
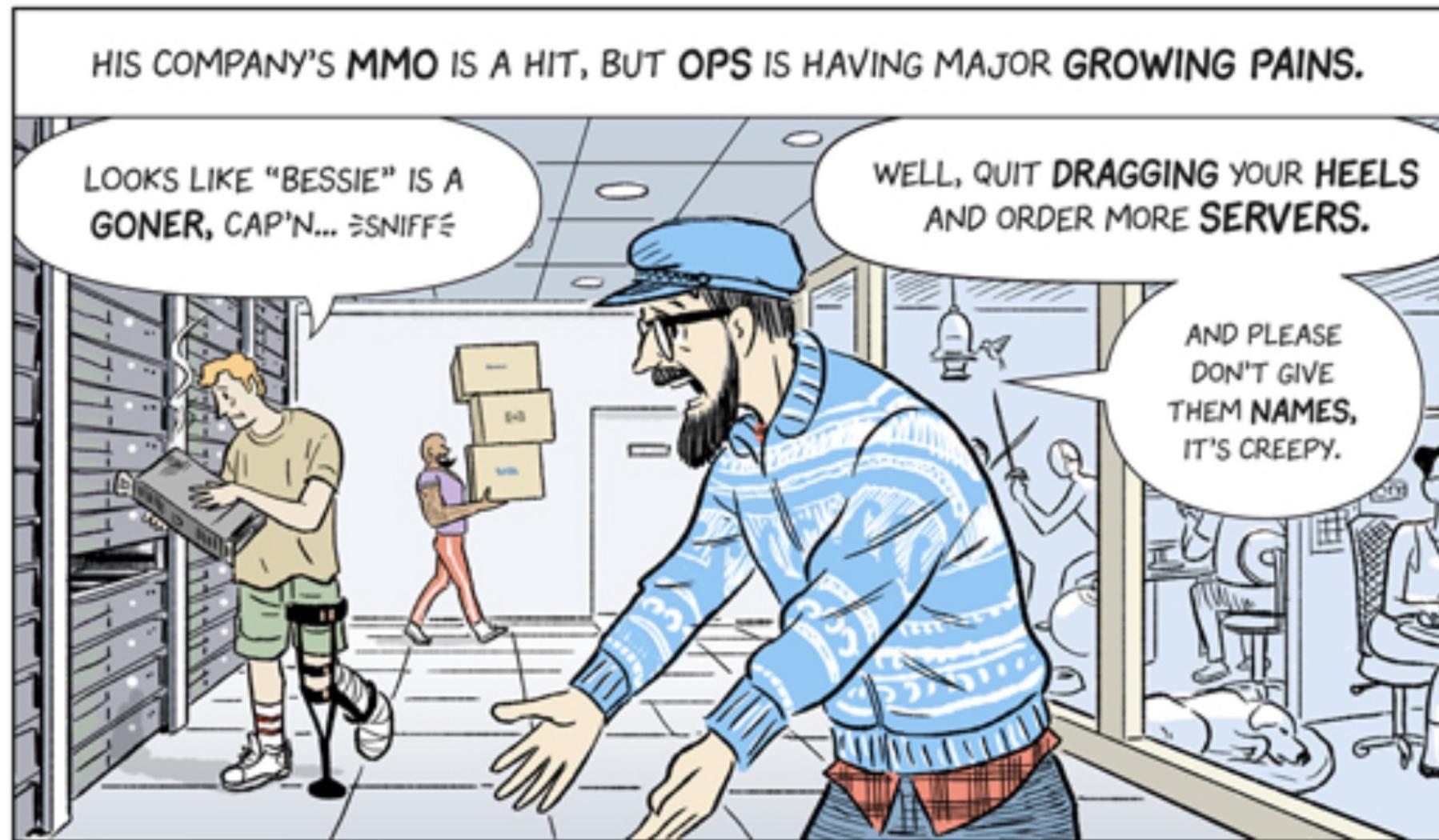
**Undifferentiated Heavy Lifting**  
**Lots of work that doesn't add value**  
**Reduce as much as possible**



**Generic Kubernetes Implementation**  
**Day 1 / Hello World - incomplete**  
**Day 2 / Production - complete**



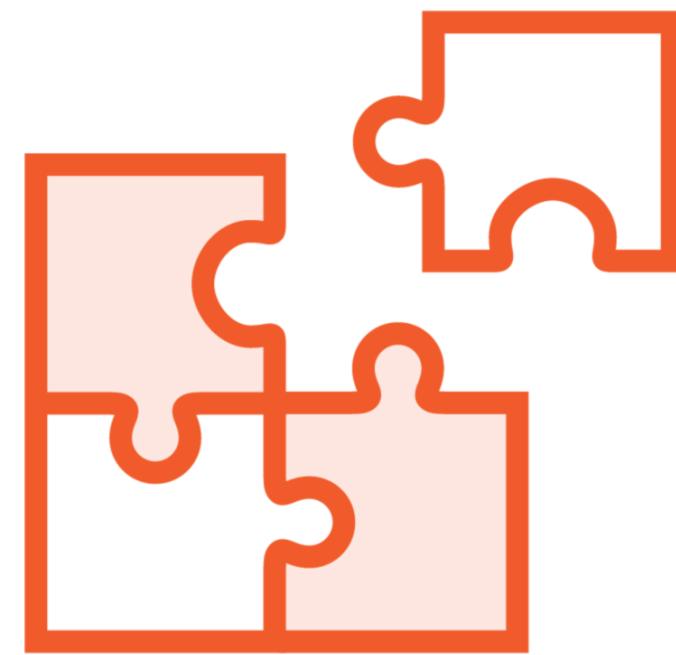
# The Kubernetes Comic



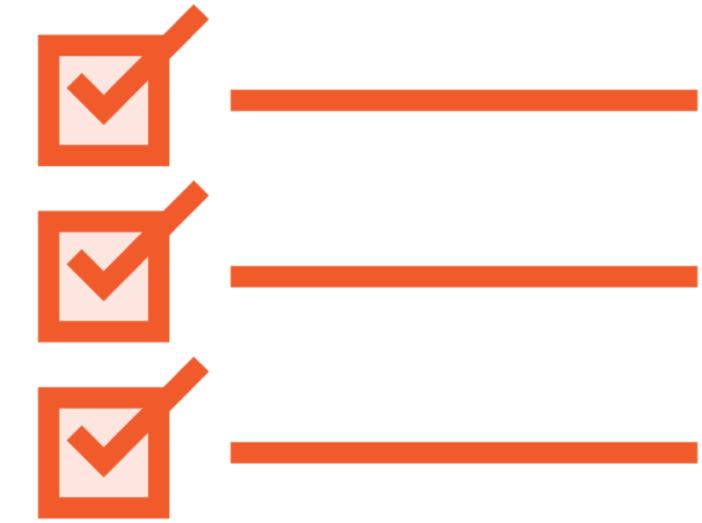
<https://cloud.google.com/kubernetes-engine/kubernetes-comic>



# Why Managed Kubernetes?



**Default Settings**  
May not be compatible



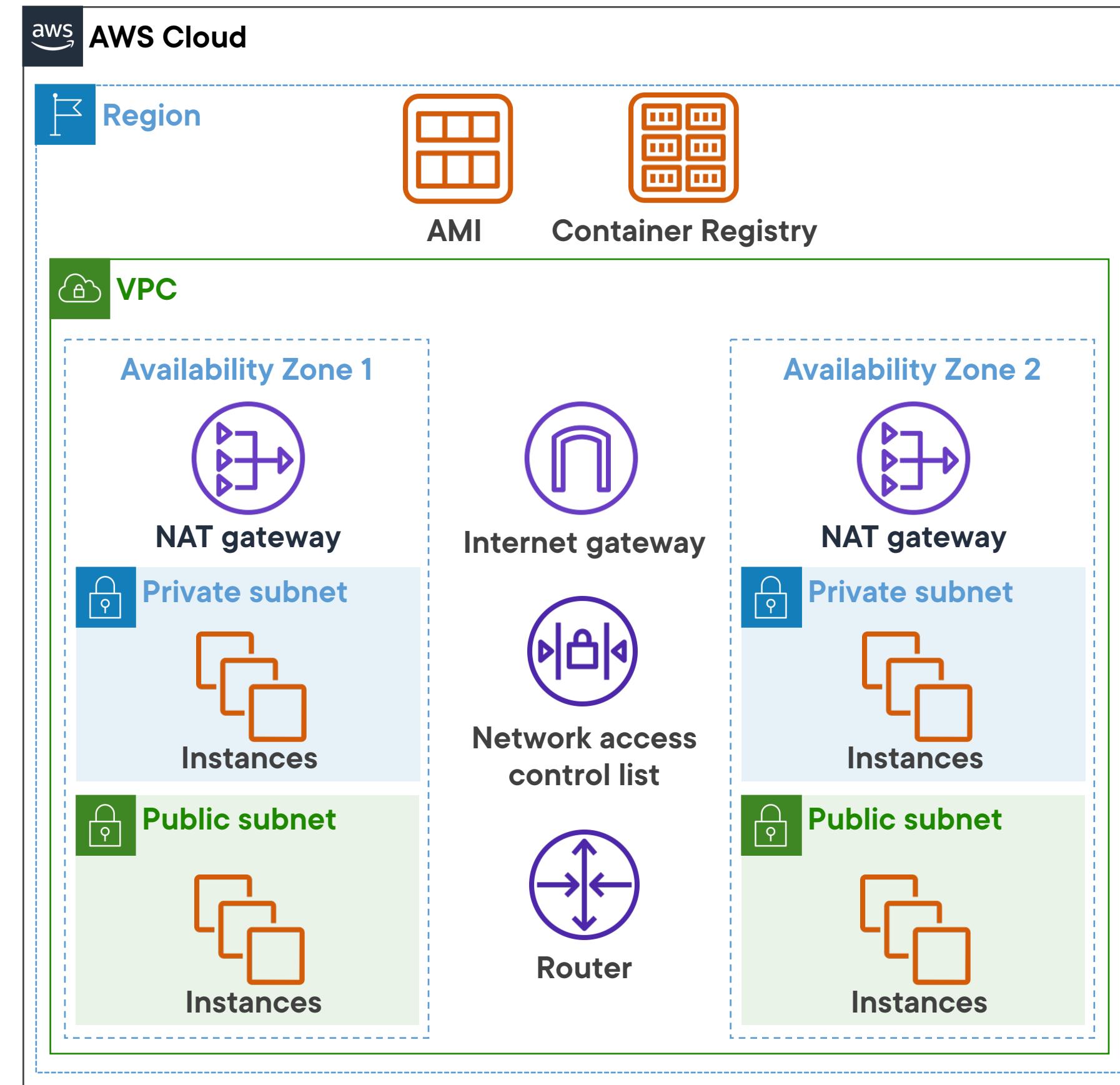
**Leverage framework**  
Production-grade



**Cloud infrastructure**  
Decision points



# Cloud Infrastructure



# Environments

## Development

Most recent code

Rapidly changing

Internal use

## Staging

Stable environment

Pre-production

Release candidate

## Production

Live application

Minimize downtime

Closely monitored



# Additional Services

**Cluster auto-scaling**

**Monitoring and metric display**

**Log management**

**Ingress controller**

**Certificate manager**

**External DNS**





## Additional services for EKS cluster

- Some are mandatory for app
- Others optional (but helpful)

Several choices for each service

Don't get stuck in analysis paralysis



# Supporting Tools for EKS Cluster

## Opinionated toolset

Widely used

Tested

Secure

Easy to get running

## Open-source options

Have necessary functionality

Simple management

Quick to set up

Can modify components



# Versioning



**Many different tools**

- Constantly changing

**Automation**

- Test and rollback

**Open-source updates**

**Know which versions you are using**

**Manage continual maintenance**

- Framework supports you



## Summary



### What EKS provides

- Undifferentiated heavy lifting

### Infrastructure for "Day 2" (production)

### Additional services

- Cluster autoscaler
- Monitoring
- Log management

### Avoid analysis paralysis

- Use established starting point
- Easy to modify



Up Next:  
Automating Your EKS Setup and Management

---

