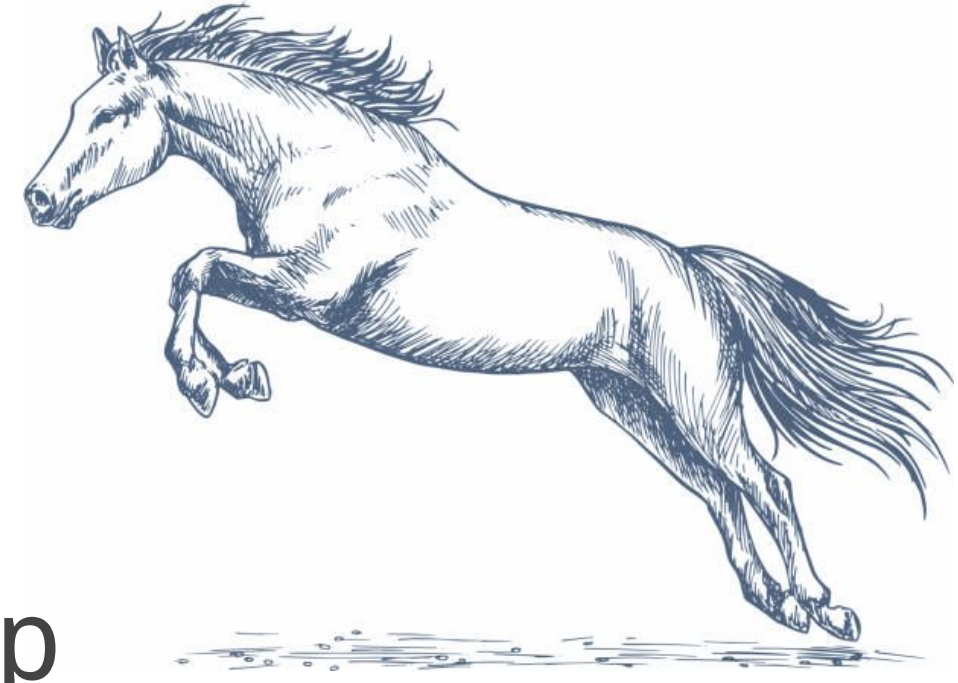
O'REILLY®



*Live Online Training*

# KCNA Exam Prep

**April 4, 2022**

Adrian Gonzalez Sanchez

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Your (friendly) instructor today

## Adrian Gonzalez Sanchez

*Telco & IT Engineer, MSc. + MBA*
*Cloud, Data & AI Specialist*
*+ University Lecturer*
*Montreal, Canada _*

**www.linkedin.com/in/adriangs86**
https://adrian-gonzalezsanchez.medium.com

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Industry trends

## Accessible entry-level cloud training and certifications for all kinds of professionals…

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Industry trends

… as an answer to the cloud shortage companies are facing today = **opportunities!**

Forbes

Mar 2, 2020, 09:30am EST | 18,724 views

## The Cloud Talent Drought Continues (And Is Even Larger Than You Thought)

Emil Sayegh Contributor
Cloud
*CEO of Ntirety. Cover all things cloud, cybersecurity & tech.*
Follow

NETWORKWORLD UNITED STATES INSIDER

## Gartner: IT skills shortage hobbles cloud, edge, automation growth

IT executives have plans to invest in emerging technologies despite the challenge of finding IT pros with the right training.

By Michael Cooney
Senior Editor, Network World | SEP 13, 2021 9:03 AM PDT

S&P Global Market Intelligence
Who We Serve Solutions

RESEARCH — 13 Jan, 2022

IN THIS LIST
RESEARCH
Major cloud providers have a talent shortage too
CASE STUDY
A Telecom Company Hones Internal Capabilities to Supercharge its Growth Strategy
BLOG

## Major cloud providers have a talent shortage too

Author Jean Atelsek

Toptal. Top 3% Why Clients Enterprise Community Blog Abou

Toptal Insights What are you looking for?

FUTURE OF WORK 10 MINUTE READ

## How to Weather the Cloud Skills Shortage

If your company isn't focused on digital transformation, you're falling behind—and putting your organization at a tremendous disadvantage. Here's what you need to know about the cloud skills shortage and how you can survive (and thrive) despite this labor shortfall.

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Industry trends

## Accessible entry-level cloud training and certifications for all kinds of professionals…

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick poll (1)

## What's your professional background?



a. Team manager

b. Software developer

c. Technology executive

d. Cloud professional

e. Student / entry-level career

f. Human resources

g. Consultant

h. Lecturer

i. Others

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick poll (2)

## What's your main goal today?



a. Obtain the KCNA certification

b. Learn about general cloud topics

c. Develop a new technical specialization

d. Apply this to my day-to-day activity

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# **Our Plan Today -** *KCNA Exam Prep*

1. Introduction to the KCNA Exam and Cloud-Native Ecosystem
2. Understanding the Lingo and Kubernetes Fundamentals
3. Overview of external resources and Linux Foundation exam process

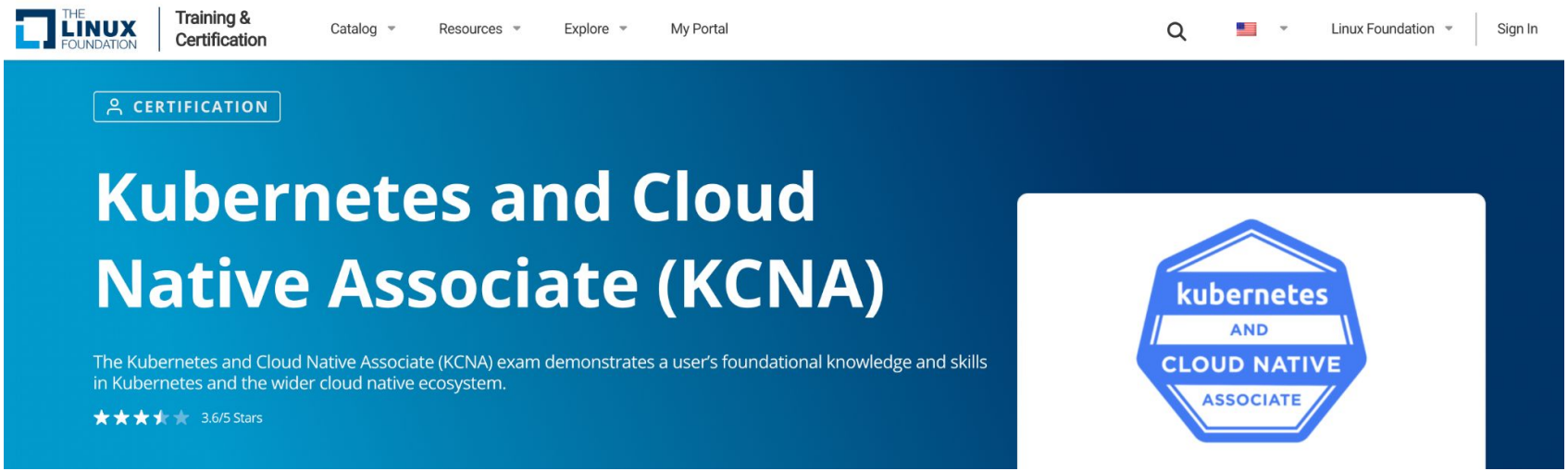
**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# 1 *Introduction to the KCNA Exam and Cloud-Native Ecosystem*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Introduction to “KCNA Exam Prep”

## Training scope, and value for Linux Foundations’ KCNA exam preparation




https://training.linuxfoundation.org/certification/kubernetes-cloud-native-associate

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Introduction to “KCNA Exam Prep”

## Training scope, and value for Linux Foundations’ KCNA exam preparation

THENEWSTACK Podcasts Events Ebooks Newsletter Sponsorship

Architecture Development Operations

cloud native professionals, which joins the Certified Kubernetes Administrator (CKA) and other certifications already available for their more experienced peers.

According to Seepersad, there has been a 455% increase in Kubernetes-certified professionals during the past three years, but that’s still not enough to meet demand.

The Linux Foundation launched KCNA “to help a new generation of cloud and Kubernetes professionals get their foot in the door,” Seeperad said. “At this point the resources are there for anyone who wants to work in this field to make a start of it, but we need even more folks to become interested in pursuing this kind of career in the first place.”

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Introduction to "KCNA Exam Prep"

## Training scope, and value for Linux Foundations' KCNA exam preparation

| Exam topics | Weight |
|---|---|
| Kubernetes Fundamentals | 46% |
| Container Orchestration | 22% |
| Cloud Native Architecture | 16% |
| Cloud Native Observability | 8% |
| Cloud Native Application Delivery | 8% |

Online, proctored, multiple-choice exam.

https://www.cncf.io/certification/kcna

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Introduction to "KCNA Exam Prep"

## Training scope, and value for Linux Foundations' KCNA exam preparation

- ✓ For all audiences (e.g., managers, devs)
- ✓ Entry-level approach, curated content
- ✓ **Adapted to the KCNA exam**

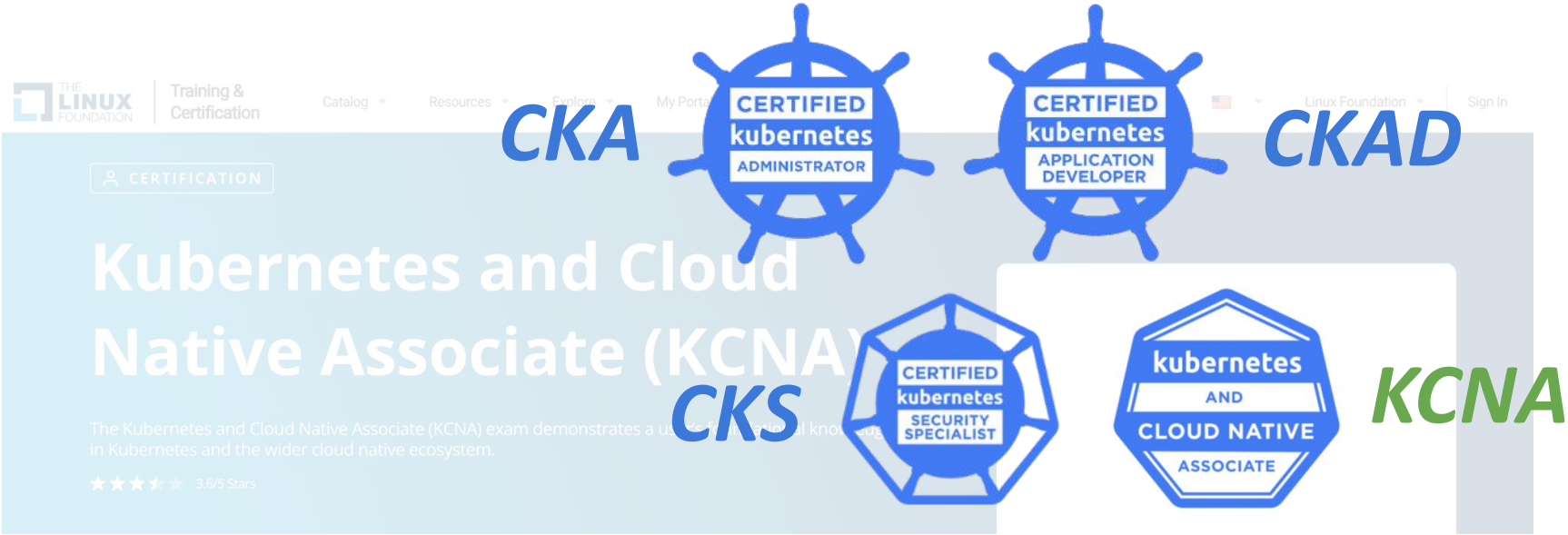
- ✗ Not an admin / expert-level session
- ✗ "Less" relevant for expert folks
- ✗ Not enough for CKA or others

Excellent entry door to the cloud native ecosystem, regardless of the technical ability

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Introduction to “KCNA Exam Prep”

## KCNA as the starting point for your Kubernetes certification journey




https://training.linuxfoundation.org/certification/kubernetes-cloud-native-associate

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick poll (3)

## What's you current level of cloud native and Kubernetes knowledge



a. Total newbie

b. Entry-level practitioner

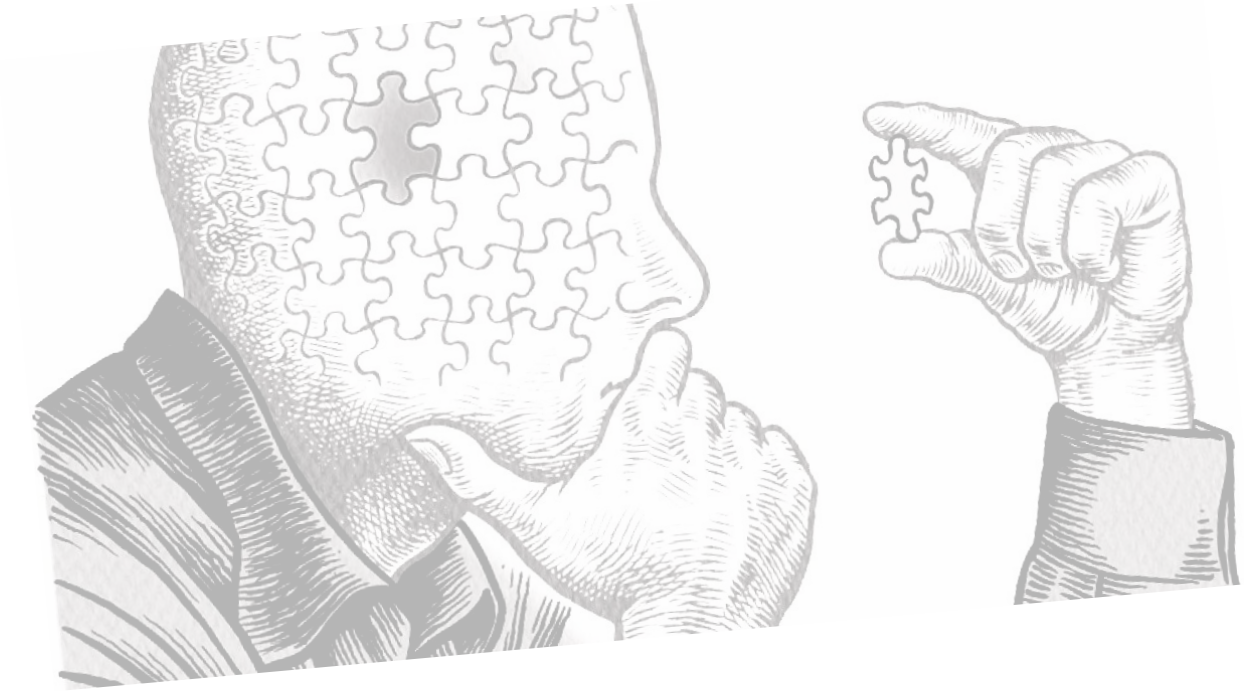
c. Currently developing advanced skills

d. Super expert / Master of the K8s universe

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# My personal journey

## Why cloud native, and how I got into it



Data & AI

↓

Cloud Computing

↓

Cloud Native

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# My personal journey

## Learnings and motivation for this type of accelerated training

Your score for the KCNA-BETA Exam Recibidos ×

**training-noreply@linuxfoundation.org** a través de amazonses.com sáb, 20 nov 2021, 5:50

para mí

Hello Adrian,

Below, you will find the exam results for the KCNA-BETA Exam that you took on Mon, Nov 01, 2021, at 10:00 AM UTC.

You scored 73% for this Exam. A score of at least 75% was required to pass.

While you did not receive a passing score, you have the option to take the exam again by selecting the exam from our Certification Catalog and registering and paying the exam fee. You can also go to www.training.linuxfoundation.org to sign up for the training newsletter, and stay informed of regular promotions and exclusive offers on our Certification Products.

To review certification requirements, go to our Certification FAQ page.

If you have additional questions, please login to trainingsupport.linuxfoundation.org and use the search bar to find an answer to your question, or select your request type from the categories provided.

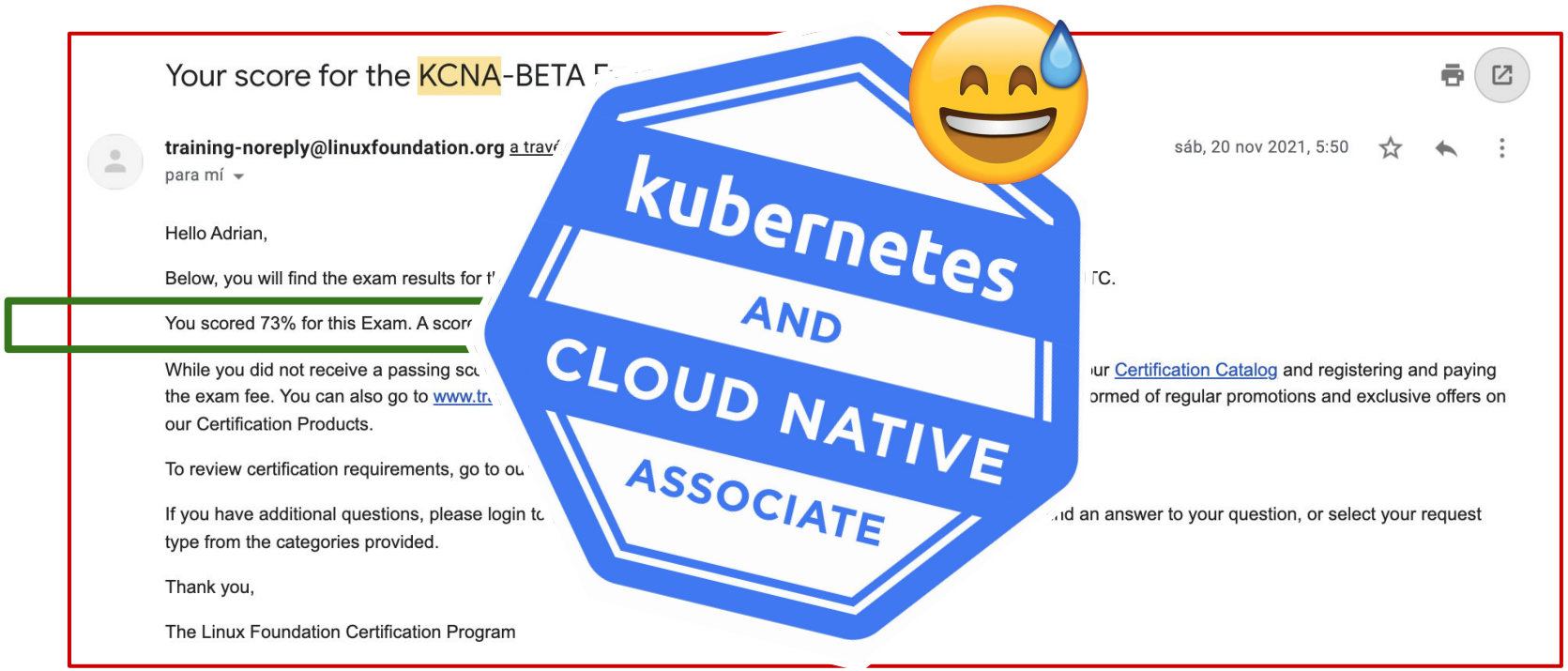
Thank you,

The Linux Foundation Certification Program

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# My personal journey

## Learnings and motivation for this type of accelerated training




**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# KCNA exam

## Official curriculum and detailed areas

### 46% - Kubernetes Fundamentals

- Kubernetes Resources
- Kubernetes Architecture
- Kubernetes API
- Containers
- Scheduling

### 22% - Container Orchestration

- Container Orchestration Fundamentals
- Runtime
- Security
- Networking
- Service Mesh
- Storage

### 16% - Cloud Native Architecture

- Cloud Native Architecture Fundamentals
- Autoscaling
- Serverless
- Community and Governance
- Personas
- Open Standards

### 8% - Cloud Native Observability

- Telemetry & Observability
- Prometheus
- Cost Management

### 8% - Cloud Native Application Delivery

- Application Delivery Fundamentals
- GitOps
- CI/CD

**Source:** https://github.com/cncf/curriculum/blob/master/KCNA_Curriculum.pdf

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# KCNA exam

## Official curriculum and detailed areas

### 46% - Kubernetes Fundamentals

- Kubernetes Resources
- Kubernetes Architecture
- Kubernetes API
- Containers
- Scheduling

### 22% - Container Orchestration

- Container Orchestration Fundamentals
- Runtime
- Security
- Networking
- Service Mesh
- Storage

### 16% - Cloud Native Architecture

- Cloud Native Architecture Fundamentals
- Autoscaling
- Serverless
- Community and Governance
- Personas
- Open Standards

### 8% - Cloud Native Observability

- Telemetry & Observability
- Prometheus
- Cost Management

### 8% - Cloud Native Application Delivery

- Application Delivery Fundamentals
- GitOps
- CI/CD

awareness

**Source:** https://github.com/cncf/curriculum/blob/master/KCNA_Curriculum.pdf

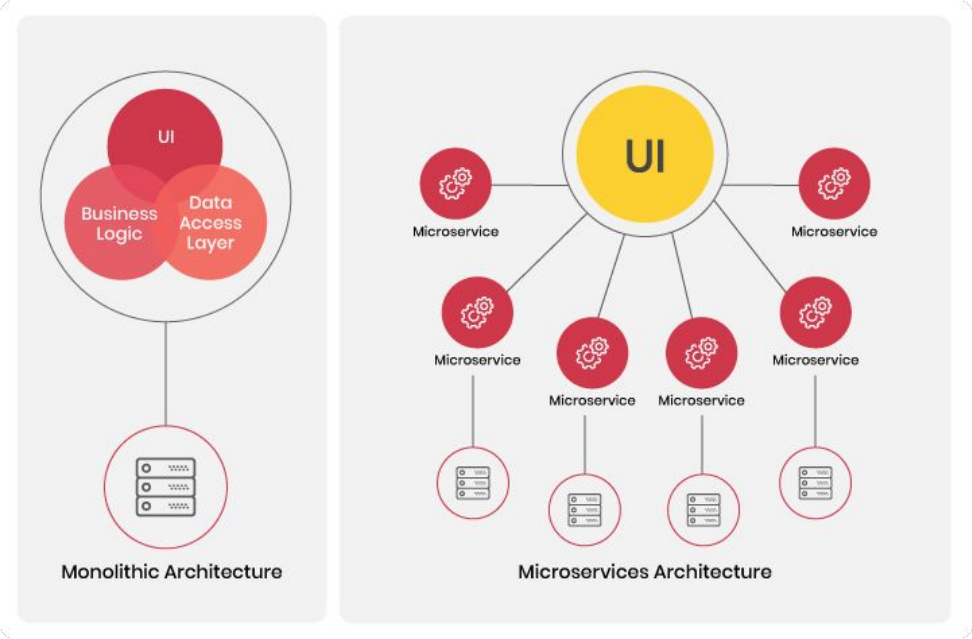
**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# 1 *Introduction to the KCNA Exam and Cloud-Native Ecosystem*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## The notion of microservices

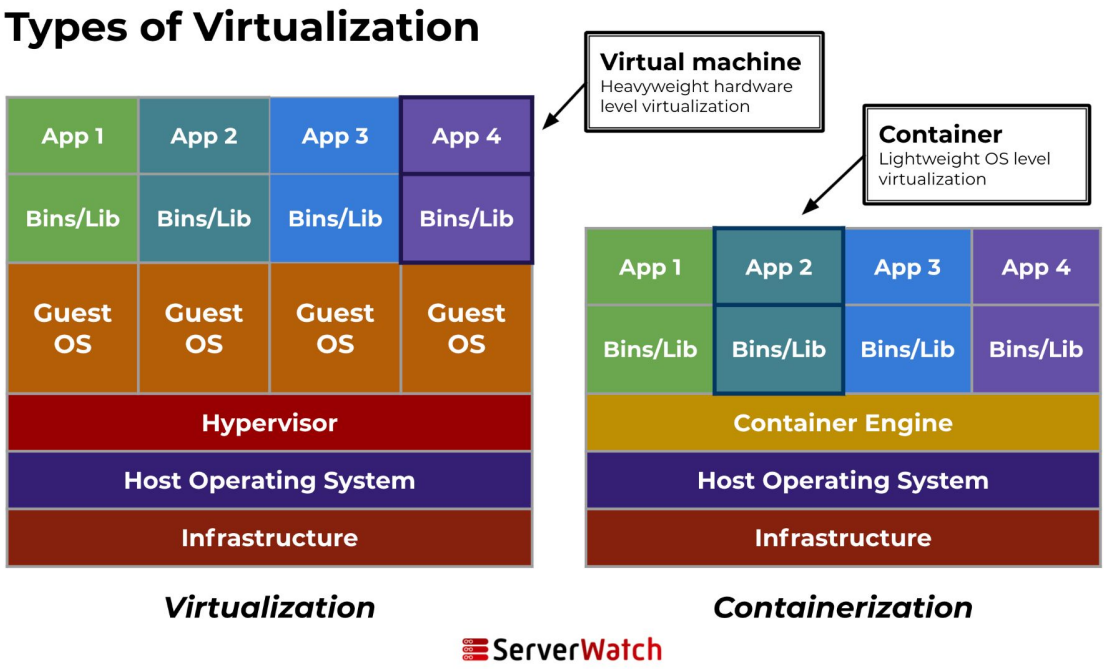



**Source:** Hyscale

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## Differences between virtualization and containerization




**Source:** ServerWatch

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## What's this and how it is different vs. general cloud computing

*Exam tip*

**Source:** Amulya Bhatia, Reddit

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## Open source as the main trend vs. vendor tools

| | ✓ PROS | ✗ CONS |
|---|---|---|
| OPEN SOURCE SOFTWARE | Ability to customize it<br>Ability to make fixes right away<br>Moves at a higher pace, specially in recent open-source projects<br>"Cheaper"<br>Auditable source-code | General lack features (i.e. more strict feature set)<br>Features and bug fixes may take a while to be implemented (or might never be)<br>Lack of effective support (support provided by the community)<br>Some projects become abandoned<br>Lack of detailed documentation<br>Inability to find trainings (on that topic, or near your location) |
| PROPRIETARY SOFTWARE | Access to support with SLAs<br>"Enterprise" level features<br>Extensive, up-to-date documentation<br>Access to official training and related materials | Unable to perform deep levels of customization<br>Dependent on vendor to make fixes and implement new features<br>Some don't have open APIs<br>Moves at a slower pace<br>Expensive<br>Non-auditable source-code |

**Source:** Xray

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## History, main actors, and key events

### Origins

- 1999-2000 → FreeBSD launches “jails”, an equivalent of what we call today “containers” in Linux
- 2003-2004: Birth of the Borg System (precursor of Kubernetes, secretly created by Google)
- 2008 → Linux Containers (LXC) are officially released
- 2012 → Heroku creates the 12-Factor app methodology
- 2013 → Docker is released. There is a new standard for containers
- 2013 → Google releases Omega (evolution from Borg)
- 2014 → Google introduces Kubernetes as an open-source version of Borg
- 2015 → Kubernetes 1.0 is released. Several companies start to join the Kubernetes community

### The CNCF era (2015+)

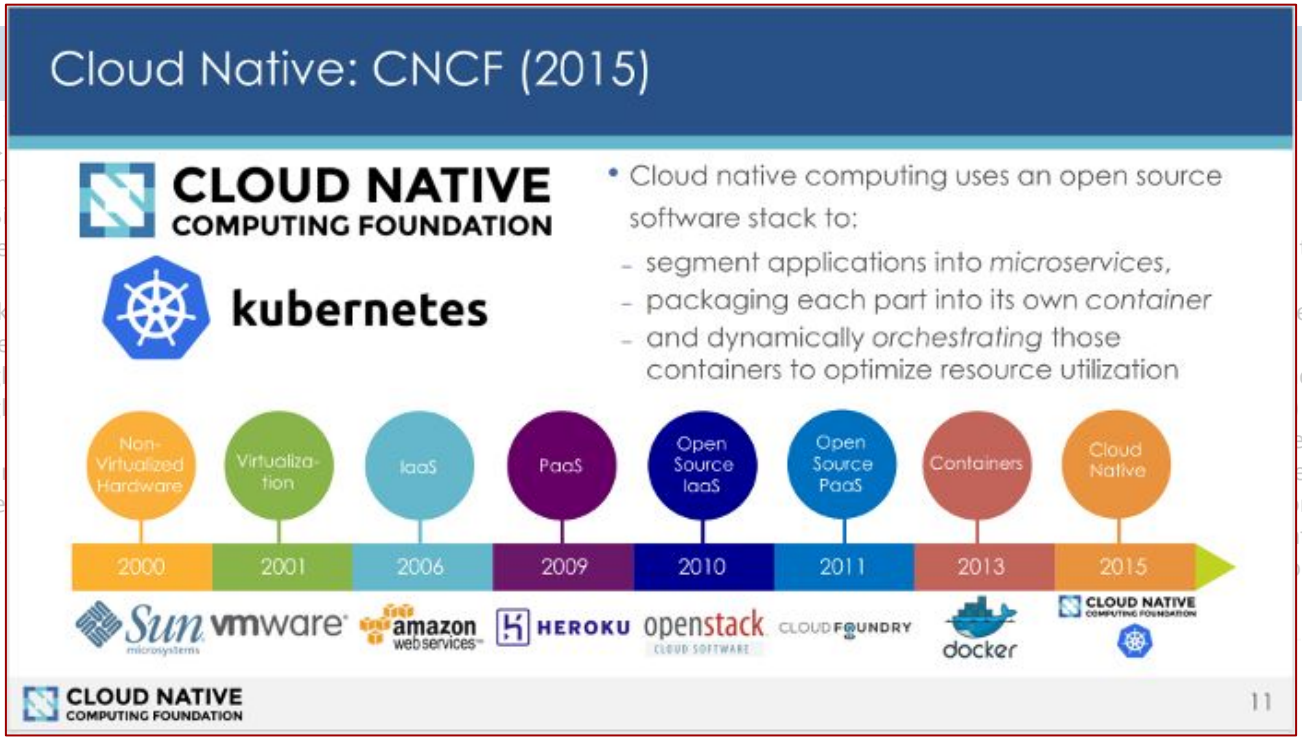
- 2015 → Docker (via Linux Foundation) creates the Open Container Initiative (OCI, initially called OCP) to get together companies and define open containerization standards
- 2015 → Google and the Linux Foundation create together Cloud Native Computing Foundation (CNCF)
- 2015 → Google donates the Kubernetes code to the CNCF
- 2016-2018 → Kubernetes goes mainstream.
  - Several Kubernetes-related projects are launched: Helm (package manager), Minikube (for local implementations)
  - “Pokemon GO” video game is launched, and it leverages Kubernetes
  - New support for OpenAPI (standard for API-based communications)
  - Apache Mesos and Docker announce support for Kubernetes
- 2018 → Kubernetes becomes the first CNCF graduated project
- 2018 → Google, Amazon, and Microsoft release cloud-enabled services to facilitate Kubernetes implementations

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## History, main actors, and key events

- 1999-2000 → the Open Container
  call today "cor companies and define
- 2003-2004: B
  secretly create together Cloud Native
- 2008 → Linux
- 2012 → Herok e CNCF
- 2013 → Docke
- 2013 → Googl ched: Helm (package
- 2014 → Googl
  Borg everages Kubernetes
- 2015 → Kube ed communications)
  the Kubernete r Kubernetes
  ted project
  ud-enabled services to




**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## The role of the Cloud Native Computing Foundation (CNCF)

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## The role of the Cloud Native Computing Foundation (CNCF)

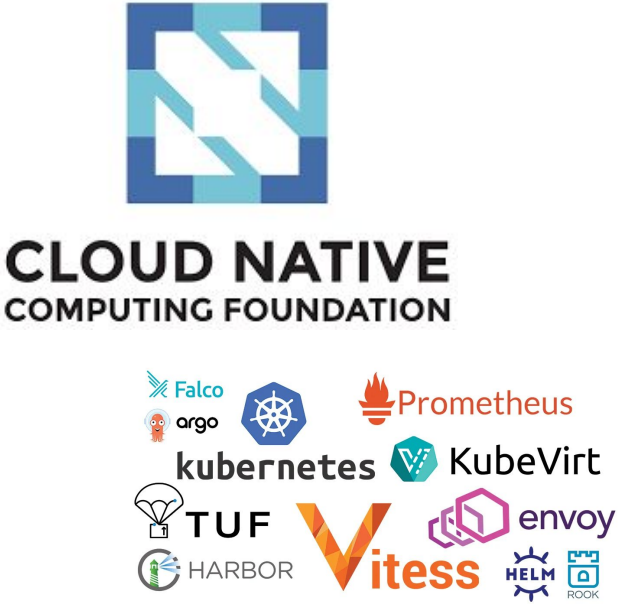
CLOUD NATIVE
COMPUTING FOUNDATION

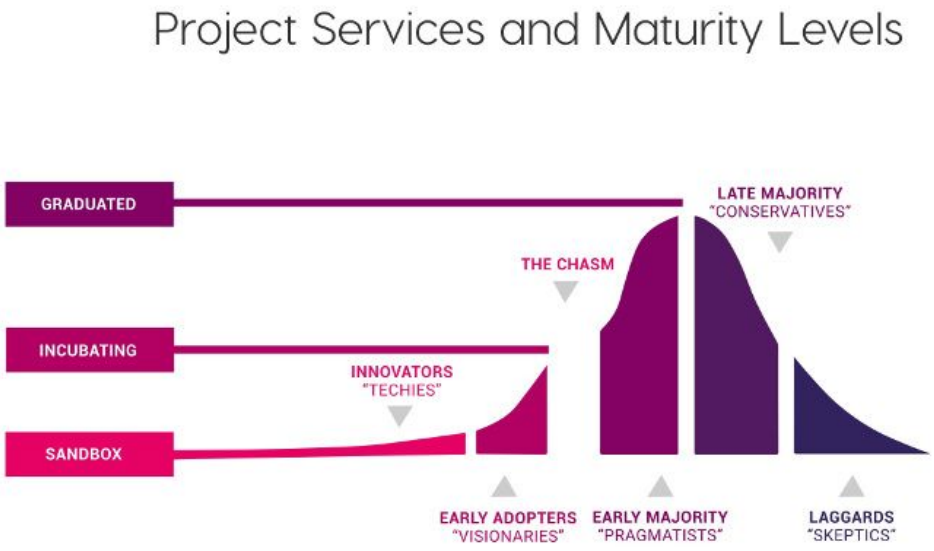
***"CNCF is an open source software foundation dedicated to making cloud native computing ubiquitous"***

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## The role of the Cloud Native Computing Foundation (CNCF)




**Source:** CNCF - https://www.cncf.io/projects

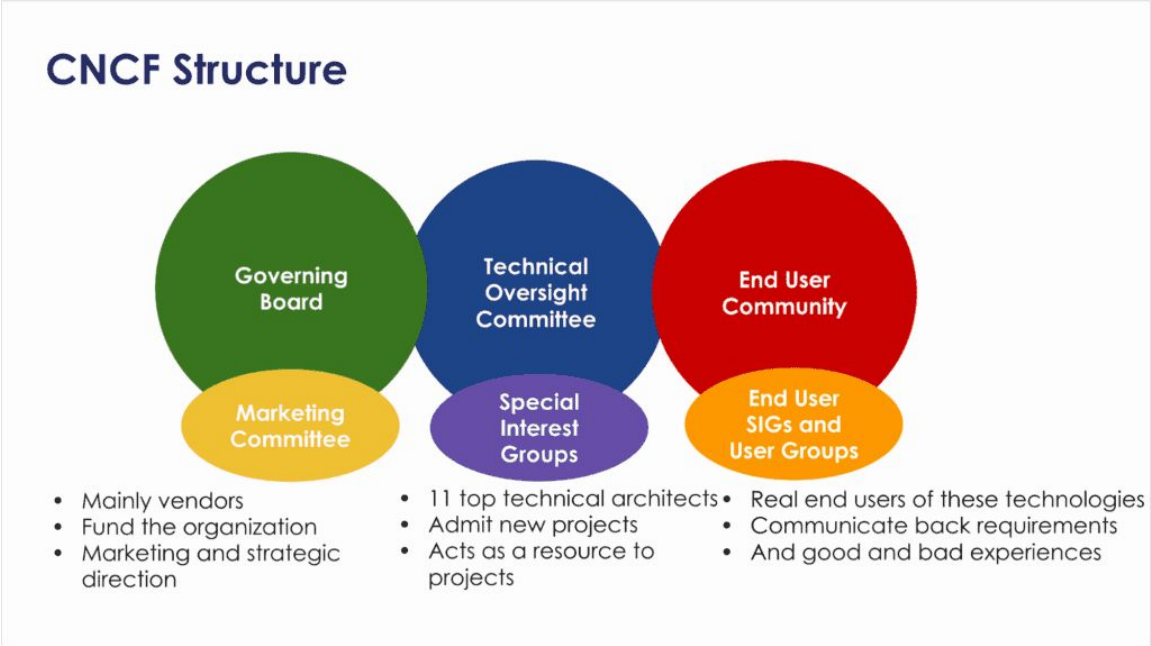
**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## The role of the Cloud Native Computing Foundation (CNCF)

*Exam tip*

CLOUD NATIVE
COMPUTING FOUNDATION




**Source:** CNCF - https://www.cncf.io/blog/2019/12/06/cncf-toc-governance-structure-elections-2020

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (1)

## How many projects have graduated at CNCF?

a. 1 (Kubernetes)

b. 5

c. 10

d. 16

e. 51

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

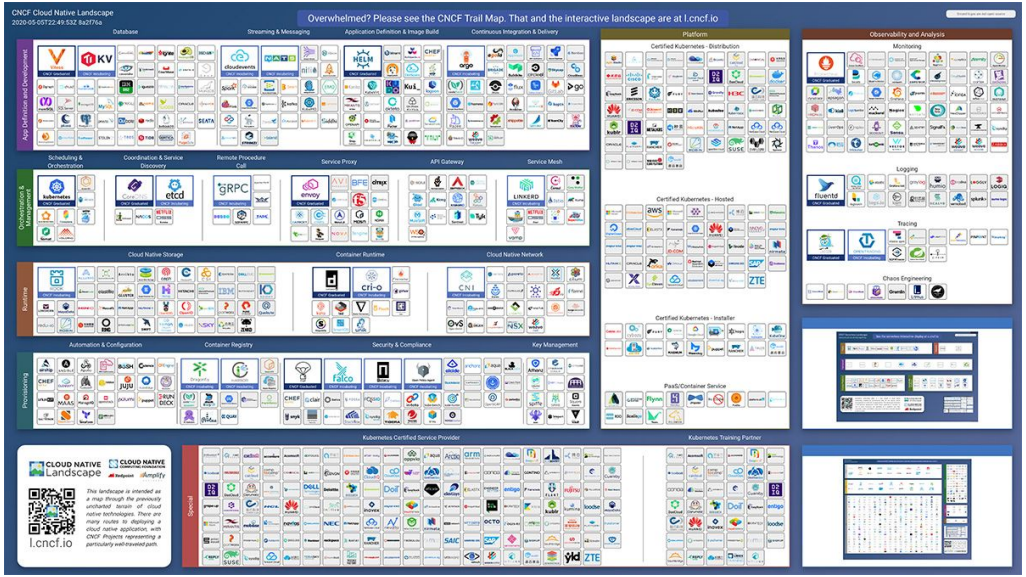
## CNCF's key resources

- Interactive Landscape
- Glossary
- Community
- Blog
- …

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's key resources - Interactive Landscape

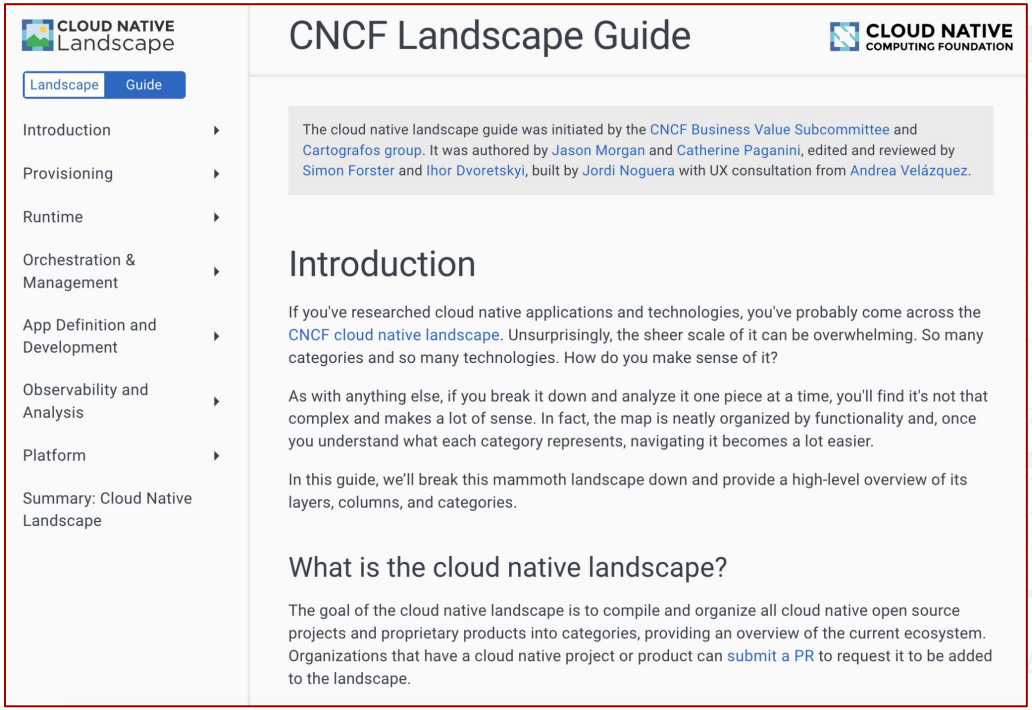


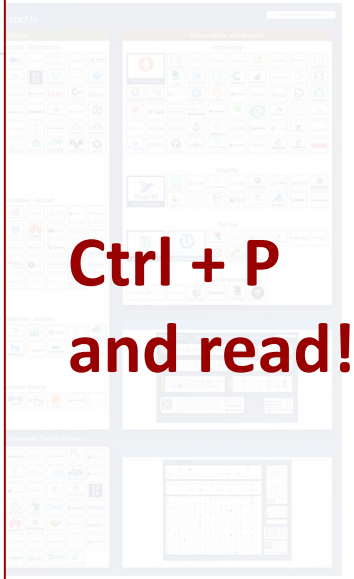

https://landscape.cncf.io + Detailed guide https://landscape.cncf.io/guide

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's key resources - Interactive Landscape



**Ctrl + P and read!**

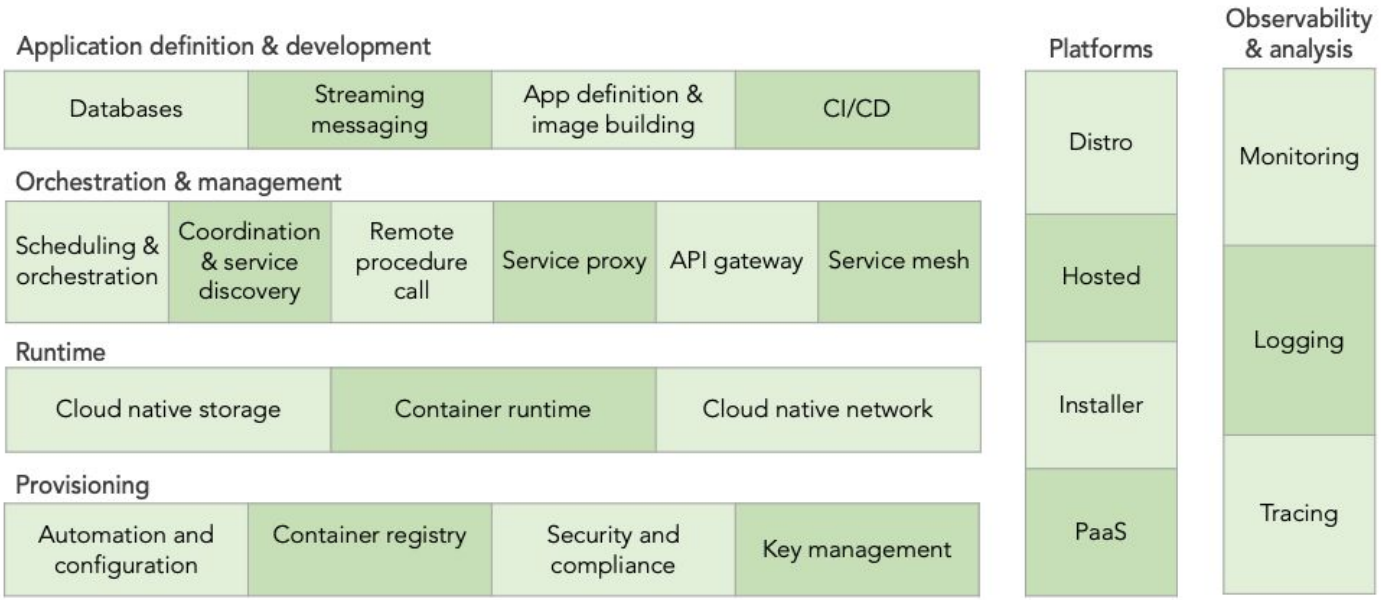
//landscape.cncf.io/guide

*Exam tip*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's key resources - Interactive Landscape




**Source:** The New Stack https://thenewstack.io/an-introduction-to-the-cloud-native-landscape

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's key resources - Glossary

*Exam tip*

CLOUD NATIVE GLOSSARY

Contributor Ladder

Search this site...

- Abstraction
- Agile Software Development
- API Gateway
- Application Programming Interface
- Autoscaling
- Bare Metal Machine
- Blue Green Deployment
- Canary Deployment
- Chaos Engineering
- Client-Server Architecture
- Cloud Computing
- Cloud Native Apps
- Cloud Native Security
- Cloud Native Technology
- Cluster

### Cloud Native Glossary

The Cloud Native Glossary is a project led by the CNCF Business Value Subcommittee (BVS). Its goal is to explain cloud native concepts in clear and simple language without requiring any previous technical knowledge.

### Contributing

Everybody is invited to suggest changes, additions, and improvements to the Cloud Native Glossary. We employ a community-driven process governed by the CNCF to develop and improve upon this shared lexicon. This Glossary provides a vendor-neutral platform to organize a shared vocabulary around cloud native technologies. Contributions are welcome from all participants who abide by the project's purpose and charter.

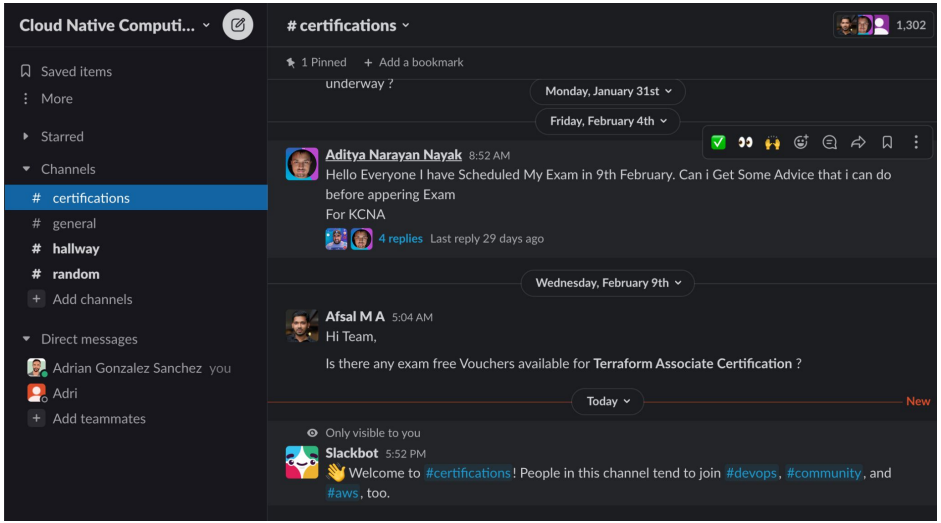
Anyone who wishes to make a contribution may submit a GitHub issue or create a pull request. Please ensure you follow the Style Guide, read the How To Contribute doc, and join the #glossary channel on the CNCF Slack. There is also a #glossary-localizations channel for those who want to help translate the glossary into their native language.

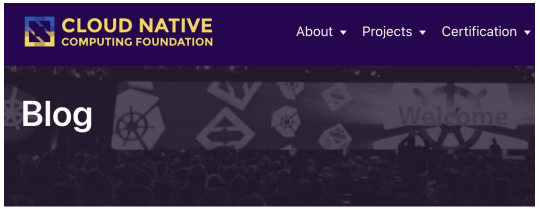
https://glossary.cncf.io

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's key resources - Community & Blog



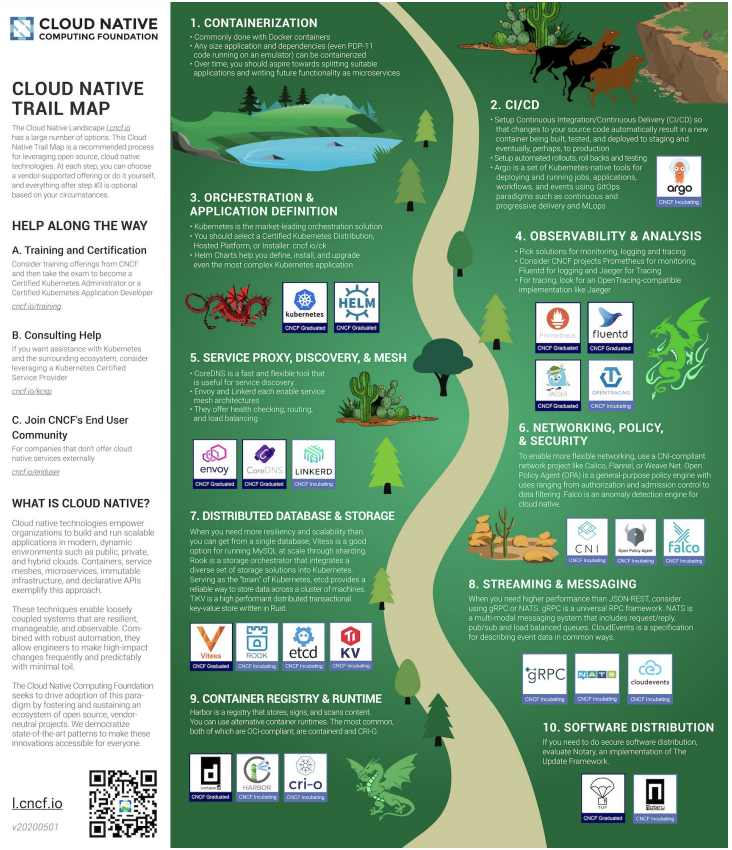



Cloud-native.slack.com + https://www.cncf.io/blog

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's Trail Map

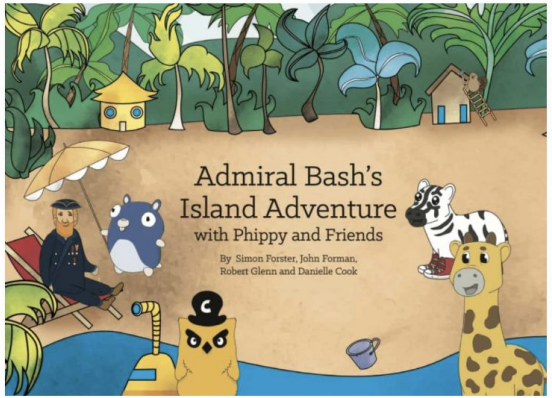



https://github.com/cncf/trailmap

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's adventure books to "to demystify cloud native computing and explain complicated concepts"




**Admiral Bash's**
**Island Adventure**
By the Cartografos Group

https://www.cncf.io/wp-content/uploads/2021/10/Admiral-Bash.pdf

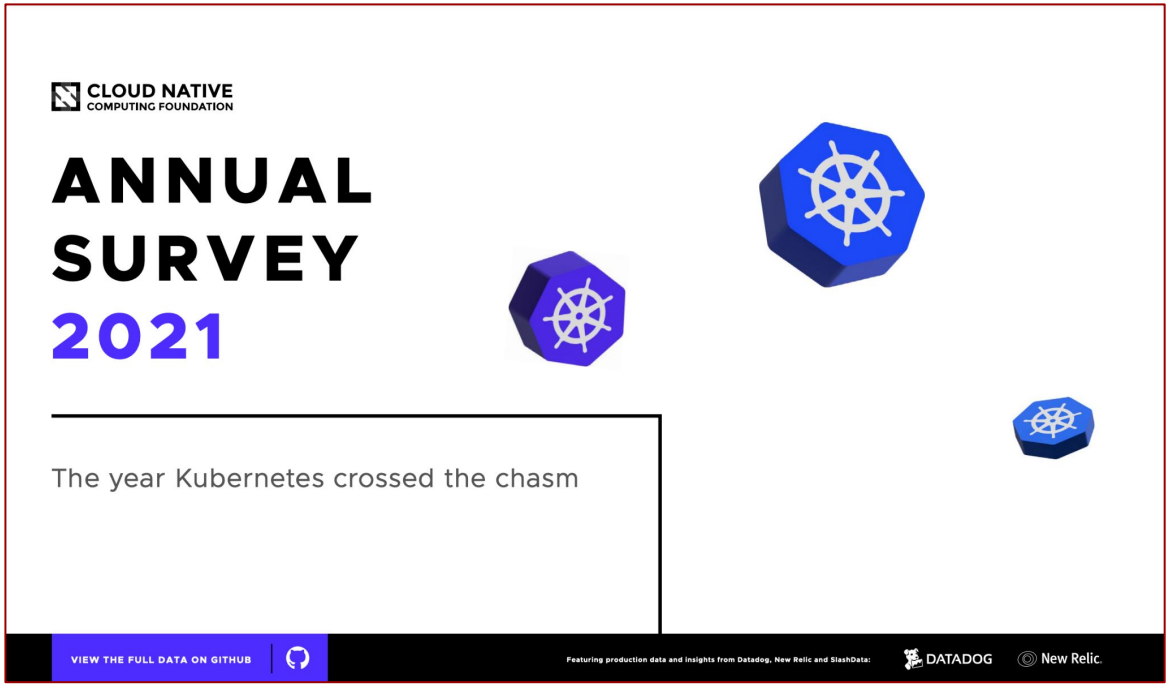
**From 00-K8s, with Love**
Donated by Microsoft Azure

https://www.cncf.io/wp-content/uploads/2021/10/From-00-K8s-with-Love_Digital.pdf

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## CNCF's Annual Report & Survey to know what's going on (trends, news, etc.)




https://www.cncf.io/wp-content/uploads/2022/01/CNCF_Annual_Report_2021.pdf
https://www.cncf.io/reports/cncf-annual-survey-2021

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud native fundamentals

## Related initiatives / projects

- Open Container Initiative (OCI) & specs - https://opencontainers.org
- CloudEvents - https://cloudevents.io
- Container Network Interface (CNI) - https://www.cni.dev
- Container Runtime Interface (CRI) - https://kubernetes.io/blog/2016/12/container-runtime-interface-cri-in-kubernetes
- Container Storage Interface (CSI) - https://github.com/container-storage-interface/spec
- Service Mesh Interface (SMI) - https://smi-spec.io

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (2)

## Which cloud model would you choose if your company wants to minimize maintenance costs?

a. Infra as a Service

b. Bare metal

c. Infra as Code

d. Platform as a Service

e. Software as a Service

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Q&A

1 of 3



**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

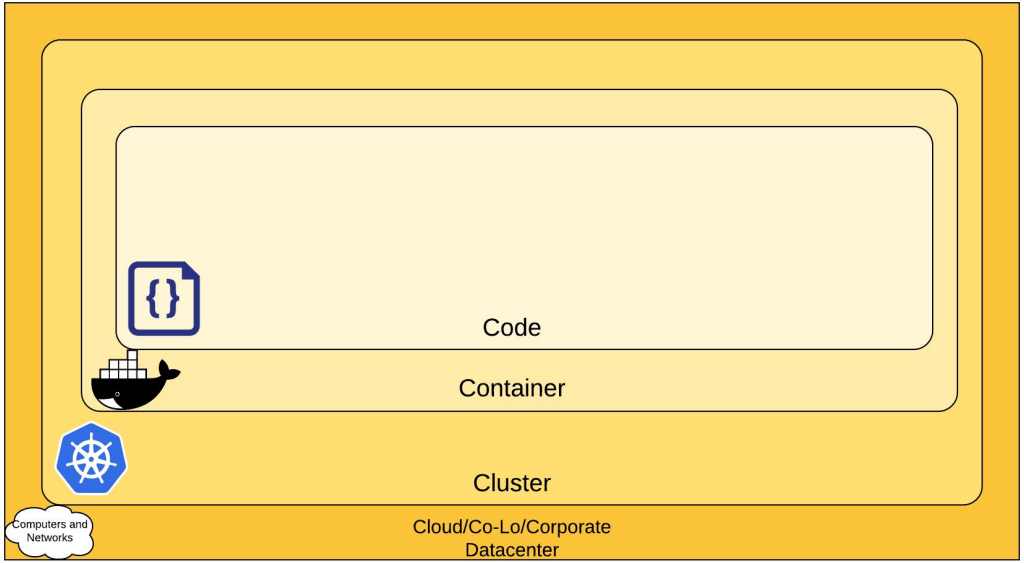
# Break

*See you in 10 minutes*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud Native Security

## The four "Cs"




https://kubernetes.io/docs/concepts/security/overview

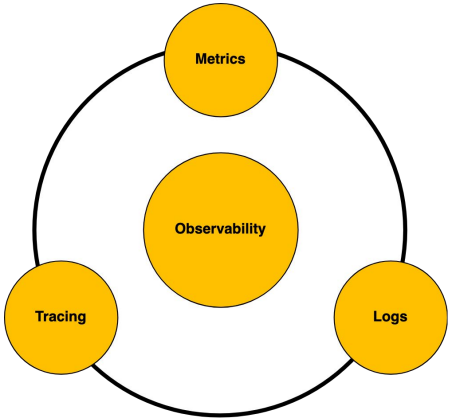
*"60 percent of organizations experienced a container security incident and in this study, 94 percent of participating companies said that they had container security concerns"*

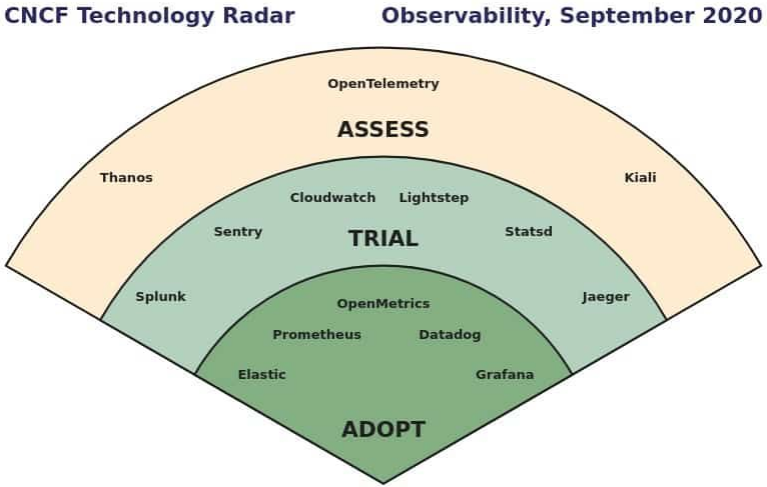
*Tripwire - the State of Security*
*https://www.tripwire.com/state-of-security/devops/organizations-container-security-incident*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud Native Observability

## Solutions for monitoring, logging, and tracing





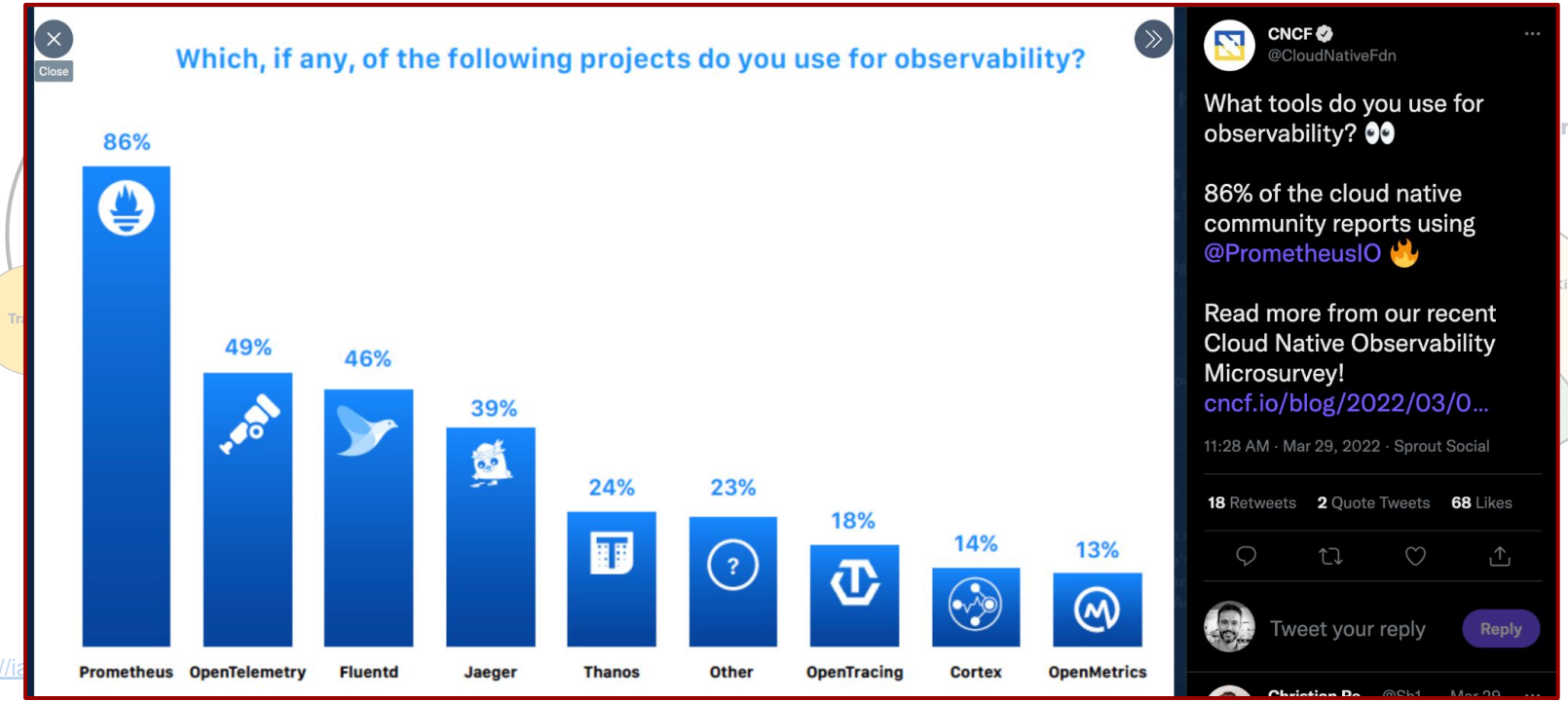

- Monitoring is capturing and displaying data
- Observability can discern system health by analyzing its inputs and outputs

https://iamondemand.com/blog/the-3-pillars-of-system-observability-logs-metrics-and-tracing

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud Native Observability

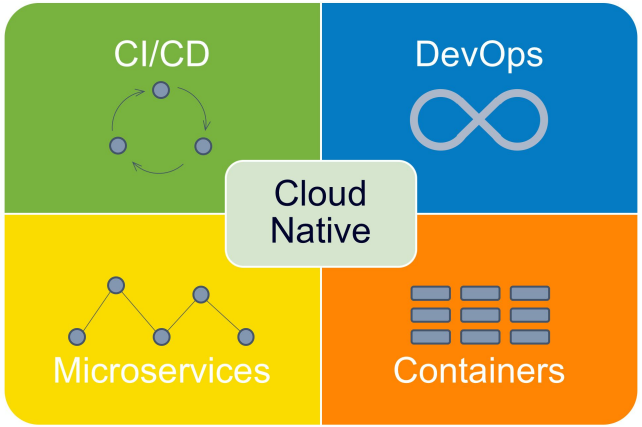
## Solutions for monitoring, logging, and tracing

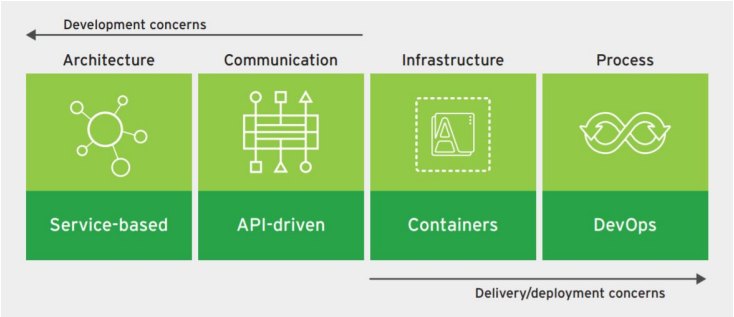



**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud Native Apps

## Dev and maturity




Cloud Native Maturity Model

*Exam tip*

| | |
|---|---|
| Cloud Native | • Micro-service Architecture and Principles<br>• API first design |
| Cloud Resilient | • Design for failure<br>• Apps are unaffected by dependant service failure<br>• Proactive testing for failure<br>• Metrics and monitoring baked in<br>• Cloud agnostic runtime implementation |
| Cloud Friendly | • Twelve Factor App<br>• Horizontally scalable<br>• Leverage platform for HA |
| Cloud Ready | • No file-system requirements or uses S3 API<br>• Self contained application<br>• Platform managed ports and addressing<br>• Consume off platform services using platform semantics |

https://blogs.sap.com/2017/03/28/developing-cloud-native-apps-on-sap-cloud-platform
https://www.instana.com/blog/cloud-native-seeing-through-hype
https://www.redhat.com/en/resources/eight-steps-cloud-native-application-development-brief

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# 2 *Understanding the Lingo and Kubernetes Fundamentals*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Cloud Native Lingo

## What are (some of) the main terms?

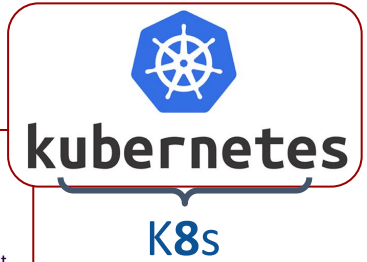
| | |
|---|---|
| **Containers** | - A **container** is a running process with resource and capability constraints managed by a computer's OS.<br>- Containerization is the process of bundling an application and its dependencies into a container image.<br>- A container image is an immutable, static file containing the dependencies for the creation of a container. |
| **Orchestration** | - Container **orchestration** automates the deployment, management, scaling, and networking of containers.<br>- Kubernetes is like a data center operating system that manages applications running across a distributed system. It schedules containers across nodes in a cluster. |
| **Pods** | - **Pods** are the smallest deployable units of computing that you can create and manage in Kubernetes.<br>- A Pod (as in a pod of whales or pea pod) is a group of one or more containers, with shared storage and network resources, and a specification for how to run the containers. |
| **Others** | - **DevOps** is a methodology in which teams own the entire process from application dev to production operations.<br>- **Stateful** Apps, wherestate refers to any data the app needs to store to function as designed.<br>- A **stateless** application doesn't save any client session (state) data on the server where the application lives.<br>- **Infrastructure as code** (IaC) is the practice of storing the definition of infrastructure as one or more files.<br>- **Continuous delivery** (CD), is a set of practices in which code changes are automatically deployed into an acceptance environment. Continuous integration (CI), is the practice of integrating code changes as regularly as possible. |

**Sources:** CNCF's cloud native glossary, Red Hat, Kubernetes.io

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## What's K8s?




### What it is

Kubernetes, often abbreviated as K8s, is a popular open-source tool for modern infrastructure automation. It's like a data center operating system that manages applications running across a distributed system (just like the OS on your laptop that manages your apps).

Kubernetes schedules containers across nodes in a cluster. It bundles several infrastructure constructs, sometimes referred to as "primitives," like an instance of an app, load balancers, persistent storage, and others together in a way that they can be composed into applications.

Kubernetes enables automation and extensibility, allowing users to deploy applications declaratively in a reproducible way. Software products and projects in the Kubernetes ecosystem take advantage of that automation and extensibility to extend the Kubernetes API. This enables them to leverage Kubernetes' automation and make their tools more accessible to experienced Kubernetes practitioners.

### Certified Kubernetes Hosted platforms

- AKS ENGINE FOR AZURE STACK
- BAIDU CLOUD CONTAINER ENGINE
- ALIBABA CLOUD CONTAINER SERVICE
- BIZMICRO
- AMAZON EKS
- BOCLOUD BEYOND CONTAINER
- AZURE (AKS) ENGINE
- GOOGLE KUBERNETES ENGINE (GKE)
- AZURE KUBERNETES SERVICE (AKS)
- RED HAT OPENSHIFT

### Serverless Platforms

- AWS LAMBDA
- AZURE FUNCTIONS
- CLOUDFLARE WORKERS
- GOOGLE CLOUD FUNCTIONS
- IBM CLOUD CODE ENGINE
- IBM CLOUD FUNCTIONS
- NETLIFY FUNCTIONS
- ORACLE FUNCTIONS
- TWILIO FUNCTIONS
- VERCEL
- OTHER

### KUBERNETES HAS CROSSED THE ADOPTION CHASM TO BECOME A MAINSTREAM GLOBAL TECHNOLOGY

**According to CNCF's respondents, 96% of organizations are either using or evaluating Kubernetes – a record high since our surveys began in 2016.** Particularly interesting is the regional adoption of Kubernetes in production, with emerging technology hub Africa (73%) jumping ahead of other more established tech centers including Europe (69%) and North America (55%). Additionally, 93% of respondents are currently using, or planning to use, containers in production, echoing 92% in our 2020 survey.

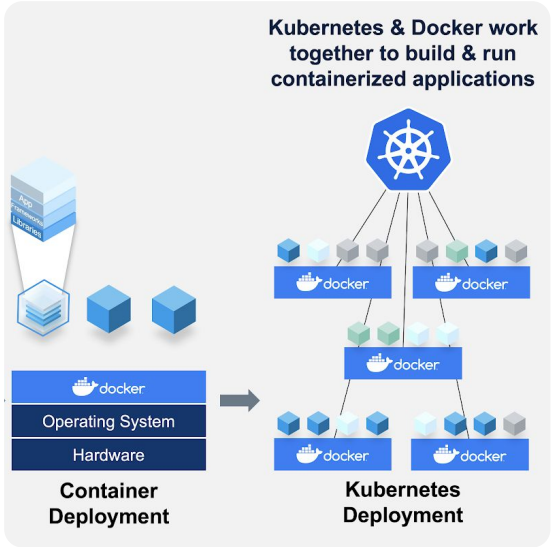
**96%** OF ORGANIZATIONS ARE EITHER USING OR EVALUATING KUBERNETES

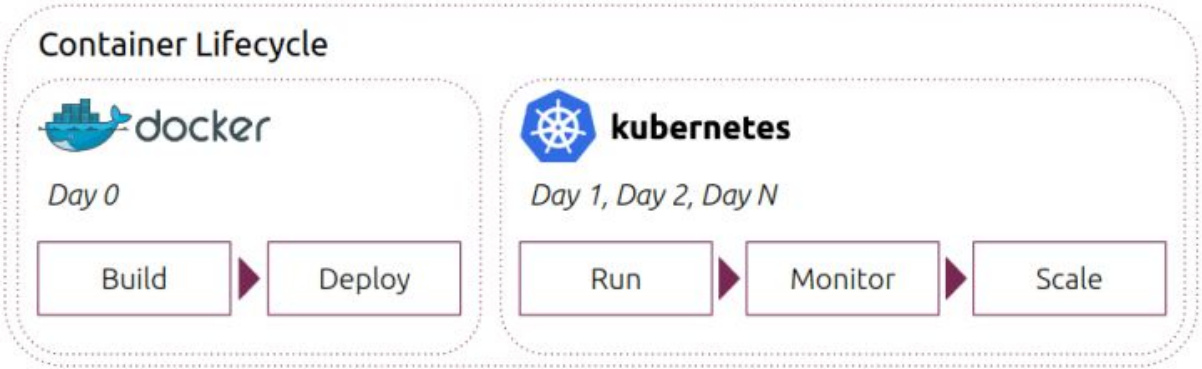
**Source:** CNCF

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes and Dockers

## How are they related? Are they the same?




**Source:** Docker, Ubuntu

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (3)

## Which of these are orchestration technologies?

a. Docker

b. Apache Mesos

c. Azure Kubernetes Service

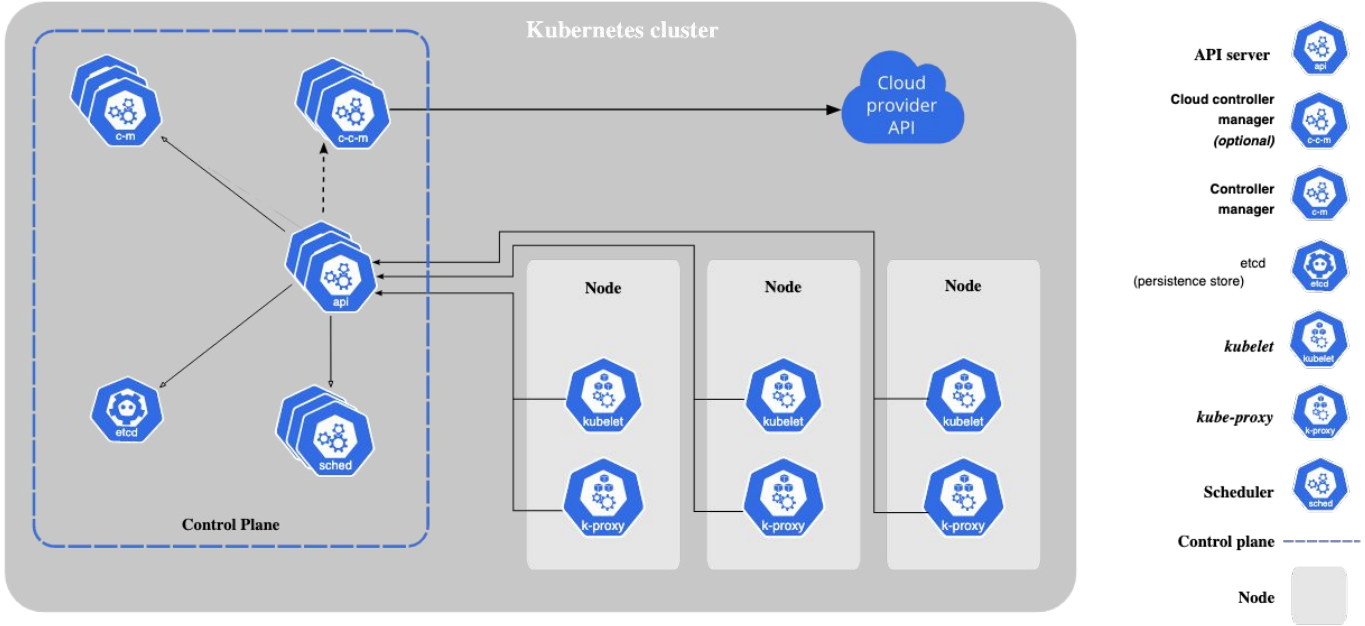
d. Kubernetes

e. Docker Swarm

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Main components

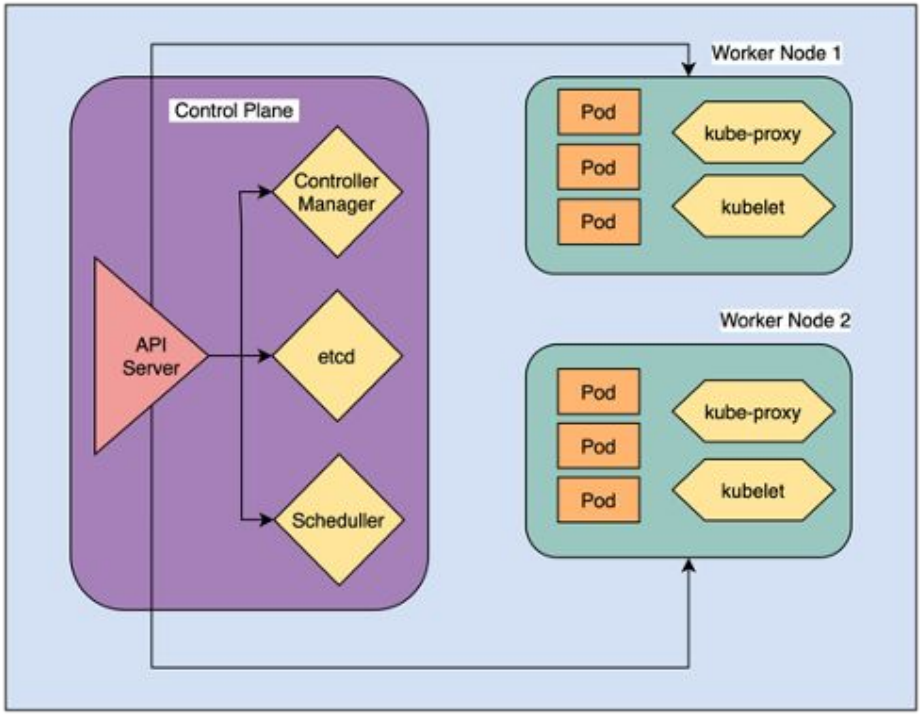



**Source:** kubernetes.io - https://kubernetes.io/docs/concepts/overview/components

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Main components

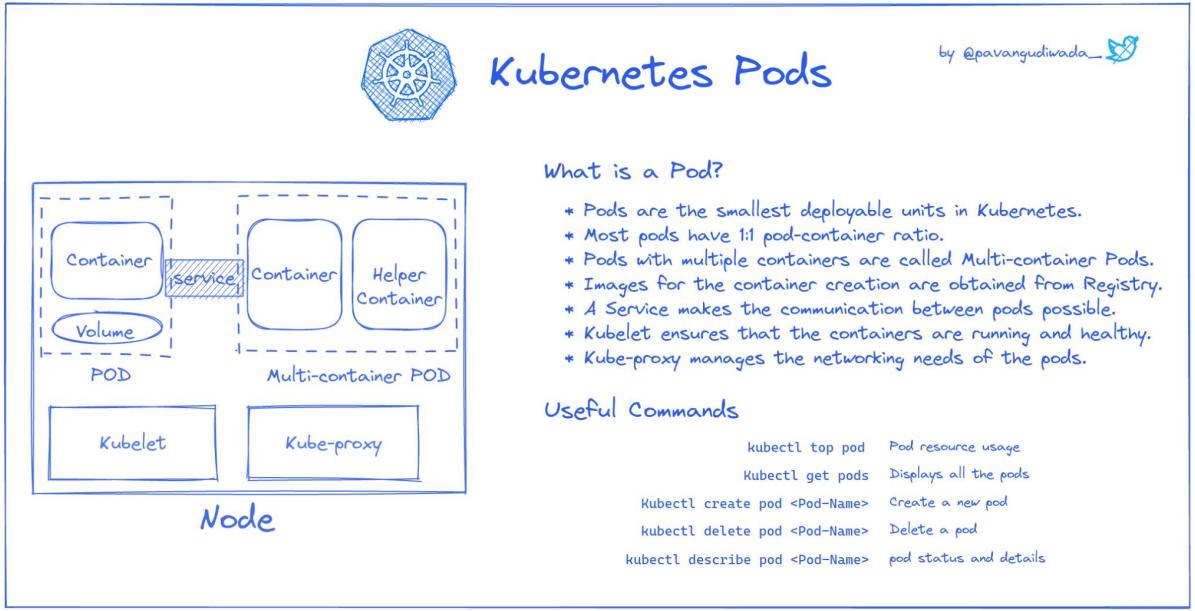



**Source:** Mohamed Abukar

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Fundamentals

## Pods

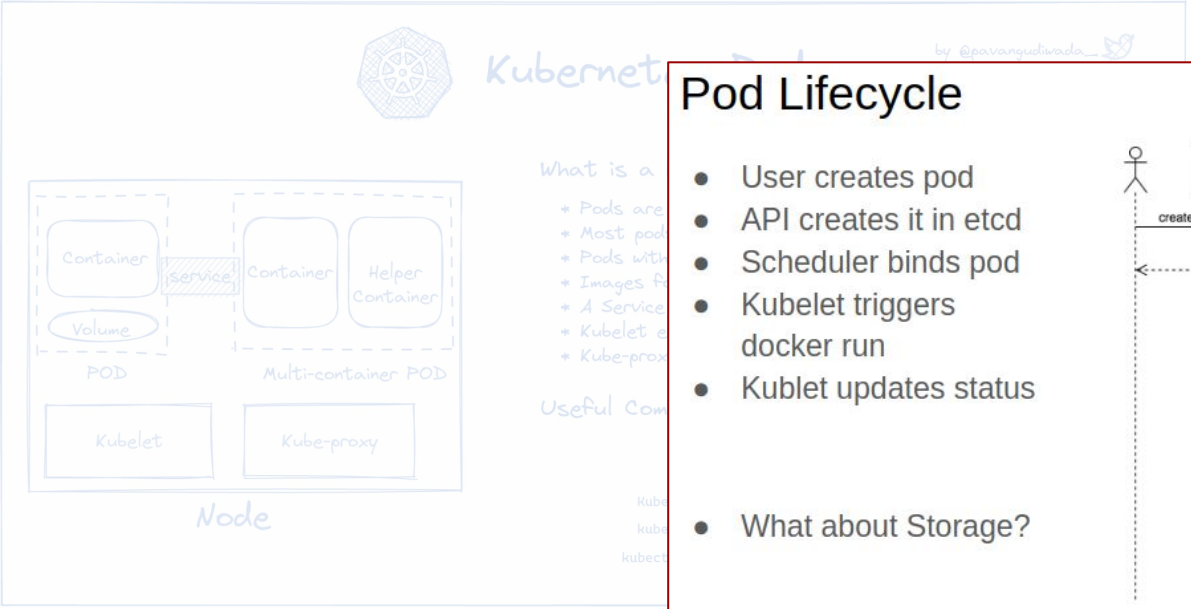



**Source:** Pavan Gudiwada, Maya Data

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Fundamentals

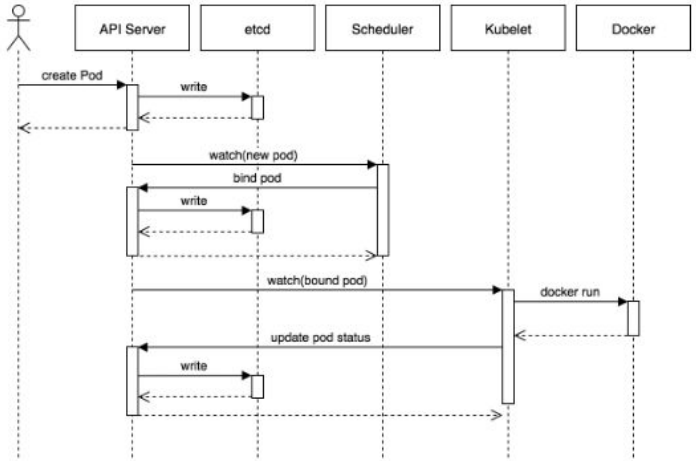
## Pods




### Pod Lifecycle

- User creates pod
- API creates it in etcd
- Scheduler binds pod
- Kubelet triggers docker run
- Kublet updates status

- What about Storage?

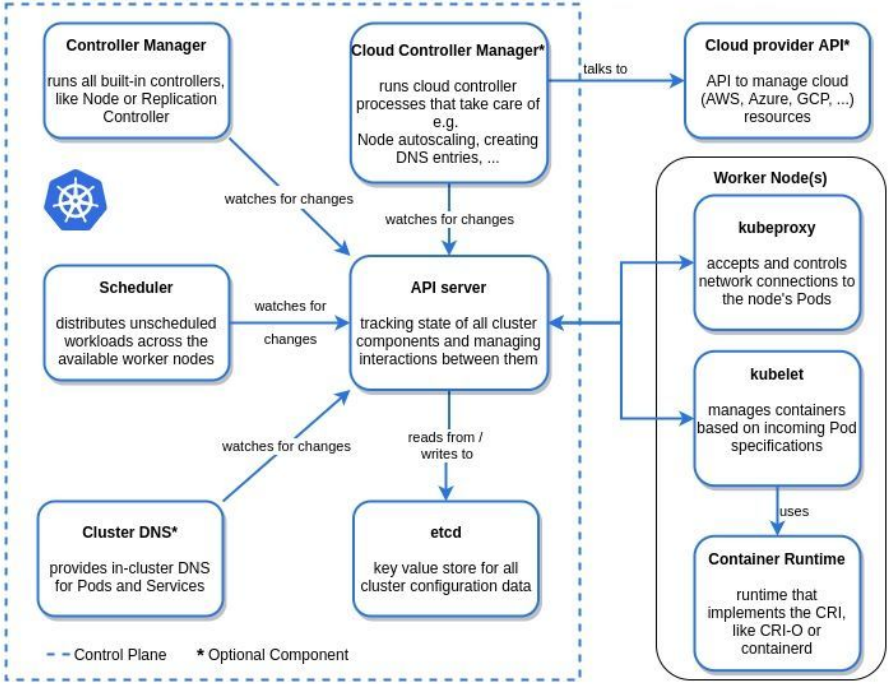



**Source:** Pavan Gudiwada, Maya Data

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Interactions between components

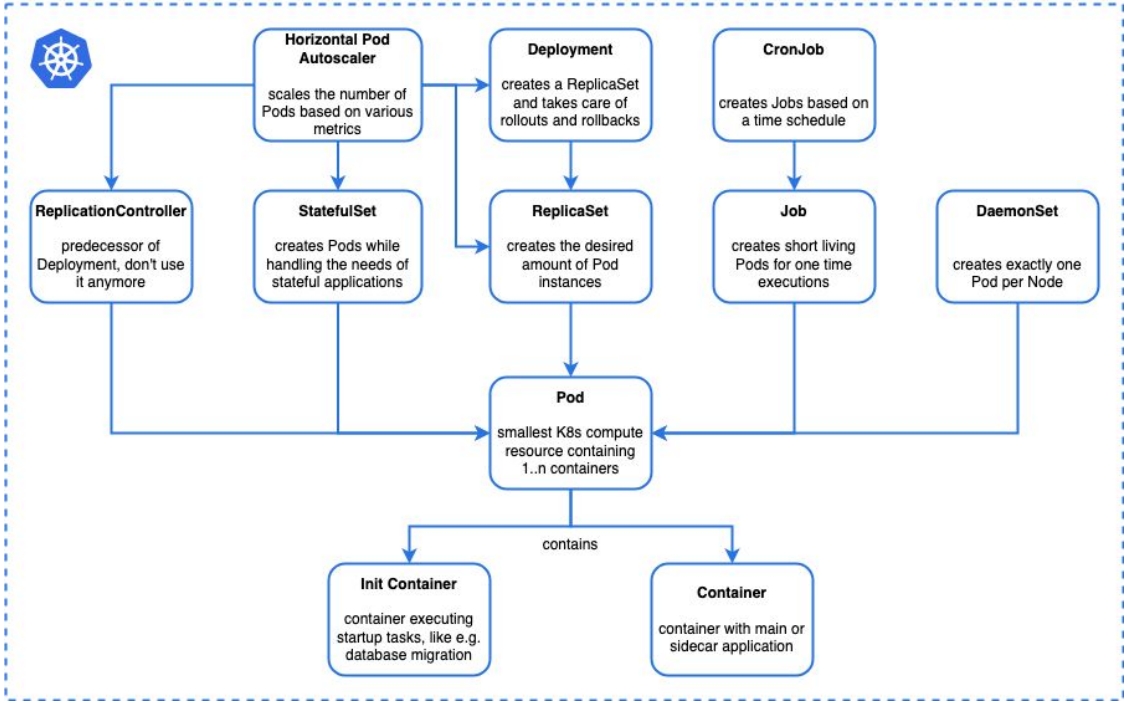



**Source:** Max Brenner - https://shipit.dev/posts/kubernetes-overview-diagrams.html

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Workload objects

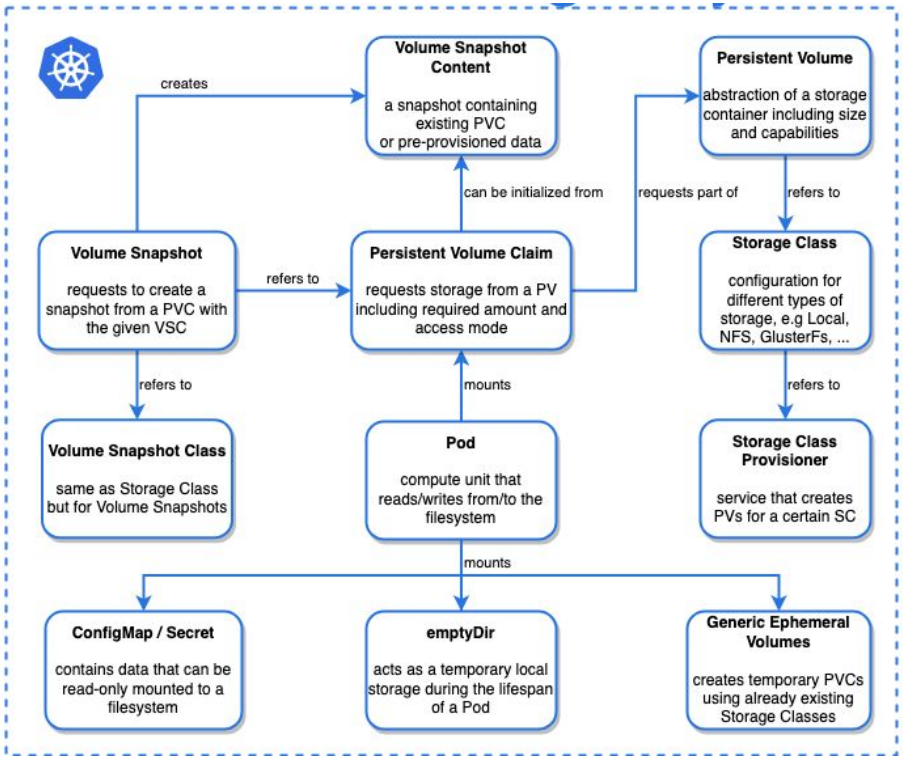



**Source:** Max Brenner - https://shipit.dev/posts/kubernetes-overview-diagrams.html

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Storage objects

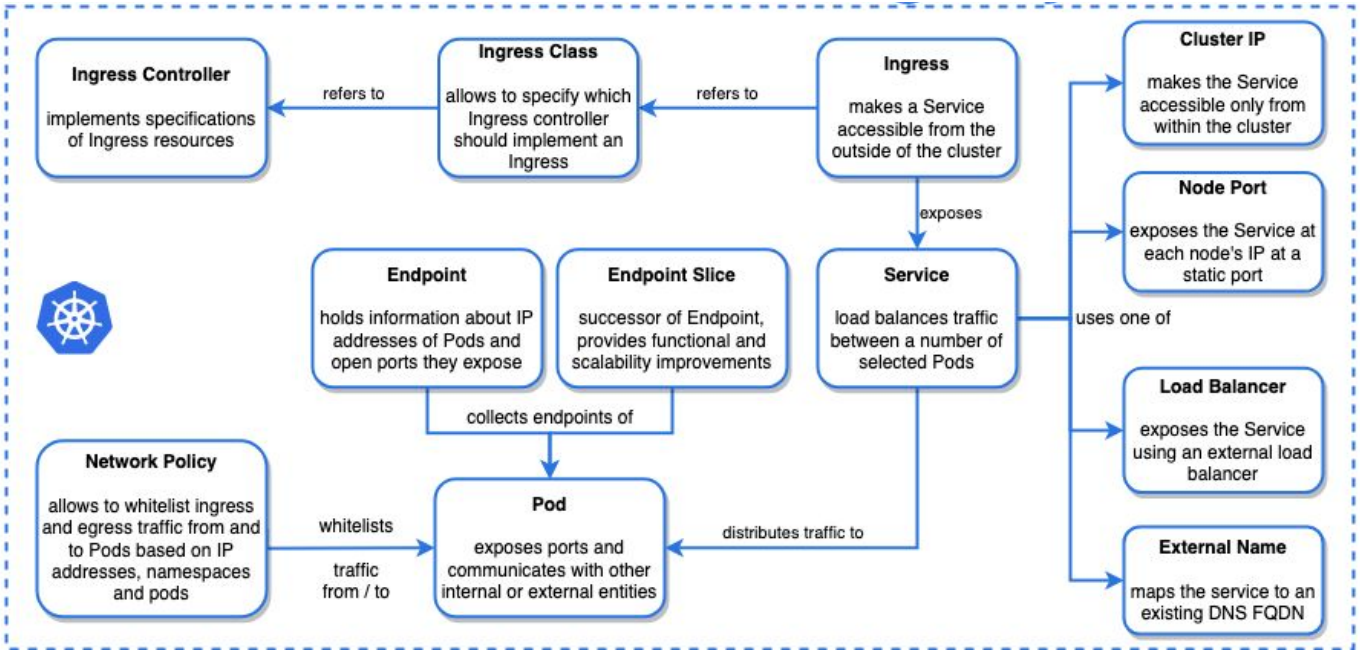



**Source:** Max Brenner - https://shipit.dev/posts/kubernetes-overview-diagrams.html

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Networking objects

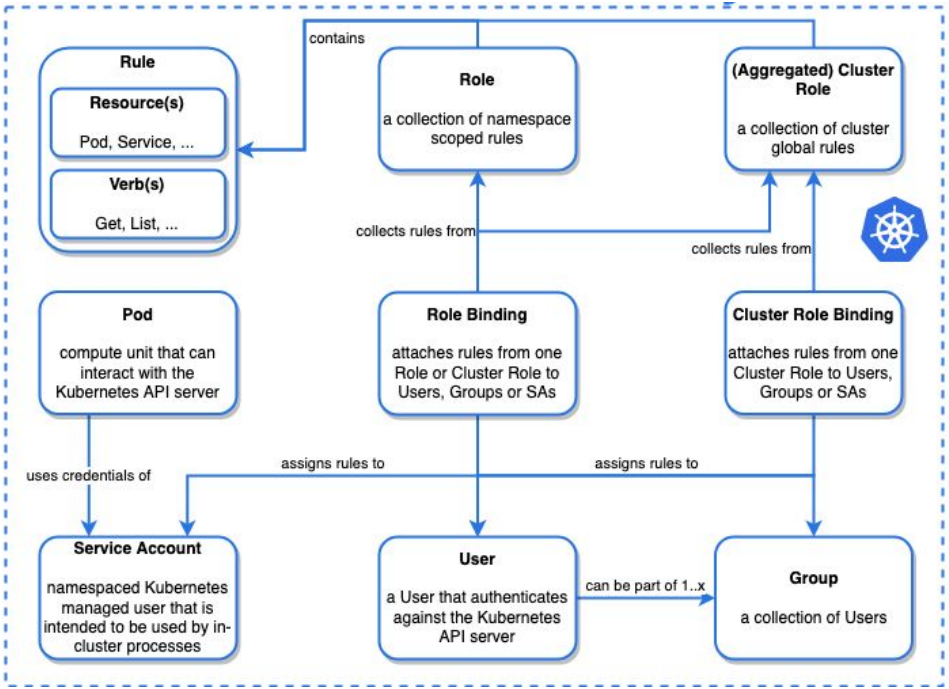



**Source:** Max Brenner - https://shipit.dev/posts/kubernetes-overview-diagrams.html

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## RBAC (Role-based access control) objects

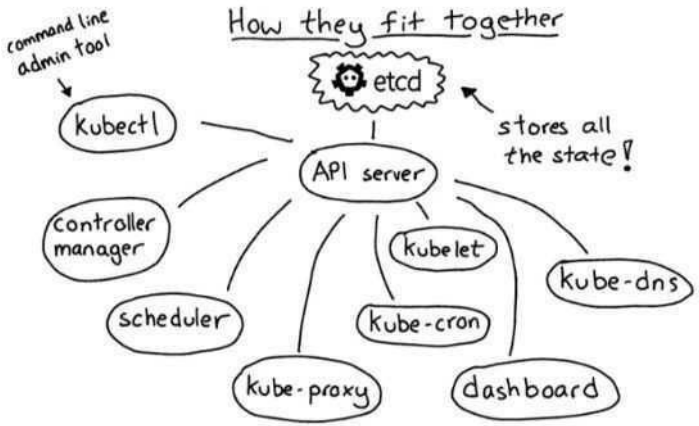



**Source:** Max Brenner - https://shipit.dev/posts/kubernetes-overview-diagrams.html

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Architecture

## Main components (another nice visual)

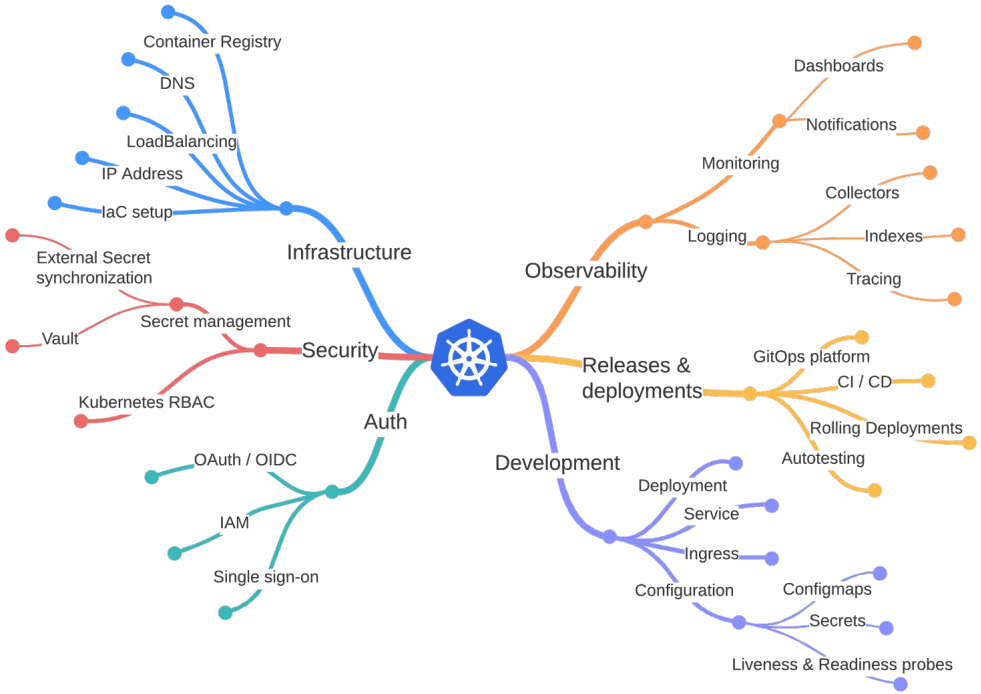



**Source:** Julia Evans

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Building Blocks

## Summary




**Source:** Pionative - https://www.pionative.com/post/6-important-things-you-need-to-run-kubernetes-in-production

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (4)

## Which component is a key value store for all cluster data?

a. Nodes

b. Kubelet

c. API server

d. Kubectl

e. etcd

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes Commands

## Main categories of commands

*Exam tip*

- Cluster Management
- Daemonsets
- Deployments
- Events
- Logs
- Manifest Files
- Namespaces
- Nodes
- Pods
- Replication Controllers
- ReplicaSets
- Secrets
- Services
- Service Accounts
- StatefulSet
- Common Options

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Cluster Management***

Display endpoint information about the master and services in the cluster
```
kubectl cluster-info
```

Display the Kubernetes version running on the client and server
```
kubectl version
```

Get the configuration of the cluster
```
kubectl config view
```

List the API resources that are available
```
kubectl api-resources
```

List the API versions that are available
```
kubectl api-versions
```

List everything
```
kubectl get all --all-namespaces
```

*A Kubernetes cluster is a set of nodes that run containerized applications.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Daemonsets "ds"***

List one or more daemonsets
```
kubectl get daemonset
```

Edit and update the definition of one or more daemonset
```
kubectl edit daemonset <daemonset_name>
```

Delete a daemonset
```
kubectl delete daemonset <daemonset_name>
```

Create a new daemonset
```
kubectl create daemonset <daemonset_name>
```

Manage the rollout of a daemonset
```
kubectl rollout daemonset
```

Display the detailed state of daemonsets within a namespace
```
kubectl describe ds <daemonset_name> -n <namespace_name>
```

*A DaemonSet is a feature used to ensure that some or all of your pods are scheduled and running on every single available node.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Deployments "deploy"***

List one or more deployments
```
kubectl get deployment
```

Display the detailed state of one or more deployments
```
kubectl describe deployment <deployment_name>
```

Edit and update the definition of one or more deployment on the server
```
kubectl edit deployment <deployment_name>
```

Create one a new deployment
```
kubectl create deployment <deployment_name>
```

Delete deployments
```
kubectl delete deployment <deployment_name>
```

See the rollout status of a deployment
```
kubectl rollout status deployment <deployment_name>
```

*A Kubernetes deployment is a resource object in Kubernetes that provides declarative updates to applications.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Events "ev"***

List recent events for all resources in the system

```
kubectl get events
```

List Warnings only

```
kubectl get events --field-selector type=Warning
```

List events but exclude Pod events

```
kubectl get events --field-selector involvedObject.kind!=Pod
```

Pull events for a single node with a specific name

```
kubectl get events --field-selector involvedObject.kind=Node, involvedObject.name=<node_name>
```

Filter out normal events from a list of events

```
kubectl get events --field-selector type!=Normal
```

*An event in Kubernetes is an object in the framework that is automatically generated in response to changes with other resources—like nodes, pods, or containers. State changes lie at the center of this.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Logs***

Print the logs for a pod

```
kubectl logs <pod_name>
```

Print the logs for the last hour for a pod

```
kubectl logs --since=1h <pod_name>
```

Get the most recent 20 lines of logs

```
kubectl logs --tail=20 <pod_name>
```

Get logs from a service and optionally select which container

```
kubectl logs -f <service_name> [-c <$container>]
```

Print the logs for a pod and follow new logs

```
kubectl logs -f <pod_name>
```

Print the logs for a container in a pod

```
kubectl logs -c <container_name> <pod_name>
```

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Namespaces "ns"***

Create namespace <name>

```
kubectl create namespace <namespace_name>
```

List one or more namespaces

```
kubectl get namespace <namespace_name>
```

Display the detailed state of one or more namespace

```
kubectl describe namespace <namespace_name>
```

Delete a namespace

```
kubectl delete namespace <namespace_name>
```

Edit and update the definition of a namespace

```
kubectl edit namespace <namespace_name>
```

Display Resource (CPU/Memory/Storage) usage for a namespace

```
kubectl top namespace <namespace_name>
```

*Namespaces are Kubernetes objects which partition a single Kubernetes cluster into multiple virtual clusters.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Nodes "no"***

Update the taints on one or more nodes

```
kubectl taint node <node_name>
```

List one or more nodes

```
kubectl get node
```

Delete a node or multiple nodes

```
kubectl delete node <node_name>
```

Display Resource usage (CPU/Memory/Storage) for nodes

```
kubectl top node
```

Resource allocation per node

```
kubectl describe nodes | grep Allocated -A 5
```

Pods running on a node

```
kubectl get pods -o wide | grep <node_name>
```

*A Kubernetes Node is a logical collection of IT resources that supports one or more containers.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands - ***Pods "po"***

List one or more pods
```
kubectl get pod
```

Delete a pod
```
kubectl delete pod <pod_name>
```

Display the detailed state of a pods
```
kubectl describe pod <pod_name>
```

Create a pod
```
kubectl create pod <pod_name>
```

Execute a command against a container in a pod
```
kubectl exec <pod_name> -c <container_name> <command>
```

Get interactive shell on a a single-container pod
```
kubectl exec -it <pod_name> /bin/sh
```

*Pods are the smallest, most basic deployable objects in Kubernetes.*

**Source:** Blue Matador - https://www.bluematador.com/learn/kubectl-cheatsheet

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Kubernetes fundamentals

## The importance of kubectl commands

*Exam tip*

**Explore other details:**

- General syntax → https://kubernetes.io/docs/reference/kubectl/#syntax
- Operations → https://kubernetes.io/docs/reference/kubectl/#operations
- Resource types → https://kubernetes.io/docs/reference/kubectl/#resource-types
- Output options → https://kubernetes.io/docs/reference/kubectl/#output-options

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (5)

## Which of these is not an actual type of Kubernetes resource?

a. StatefulSet

b. ReplicaSet

c. Nodes

d. Pods

e. Developments

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Q&A

2 of 3



**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Break

*See you in 10 minutes*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

3

# *Overview of external resources and Linux Foundation exam process*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources

## There is a huge amount of resources out there – key is to choose the relevant ones

- Fundamentals cloud / cloud-native
- Technical K8s
- Training / exam prep
- Public repositories

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources (1)

## General cloud computing and cloud native topics

1. Types of cloud computing, Red Hat https://www.redhat.com/en/topics/cloud-computing/public-cloud-vs-private-cloud-and-hybrid-cloud
2. A Practical Introduction to Container Terminology, Red Hat Developer https://developers.redhat.com/blog/2018/02/22/container-terminology-practical-introduction
3. A Brief History Of Cloud Native Applications, Redapt https://www.redapt.com/blog/cloud-native-applications-a-brief-history
4. CNCF Landscape and detailed guide, CNCF https://landscape.cncf.io + https://landscape.cncf.io/guide
5. Cloud Native Glossary, CNCF https://glossary.cncf.io
6. Getting started with your CNCF membership, CNCF https://www.cncf.io/blog/2022/01/25/getting-started-with-your-cncf-membership
7. The History of Kubernetes on a Timeline, Rising Stack https://blog.risingstack.com/the-history-of-kubernetes/
8. Docker Tutorial for Beginners, Nana Janashia https://www.youtube.com/watch?app=desktop&v=3c-iBn73dDE
9. CNCFMinutes, Saiyam Pathak https://www.youtube.com/playlist?list=PL5uLNcv9SibB658blGUEv18IhcMGL0dxC

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources (2)

## Technical Kubernetes topics

1. The Illustrated Children's Guide to Kubernetes, Deis Inc. https://www.youtube.com/watch?v=4ht22ReBjno
2. Introduction to Kubernetes architecture, Red Hat https://www.redhat.com/en/topics/containers/kubernetes-architecture
3. How To Create Kubernetes Jobs and Cronjobs, DevOps Cube https://devopscube.com/create-kubernetes-jobs-cron-jobs
4. Kubectl Commands, Tutorials Point https://www.tutorialspoint.com/kubernetes/kubernetes_kubectl_commands.htm
5. Kubernetes tasks, Kubernetes.io https://kubernetes.io/docs/tasks
6. Deployments vs StatefulSets vs DaemonSets, Stakater https://medium.com/stakater/k8s-deployments-vs-statefulsets-vs-daemonsets-60582f0c62d4
7. Kubernetes Full Beginners Tutorial, Bogdan Stashchuk https://www.youtube.com/watch?v=d6WC5n9G_sM&feature=youtu.be
8. Fundamentals of Kubernetes Networking, Nived Velayudhan https://www.linkedin.com/pulse/fundamentals-kubernetes-networking-nived-velayudhan
9. Kubernetes: The Documentary, Honeypot https://www.youtube.com/watch?v=BE77h7dmoQU
10. The Kubernetes Learning Slides, Kubernauts https://docs.google.com/presentation/d/13EQKZSQDounPC1I6EC4PmqaRmdCrpT3qswQJz9KRCyE
11. Kubernetes in Action (book) https://livebook.manning.com/book/kubernetes-in-action/about-this-book

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources (3)

## Available training options

1. Cloud Native Fundamentals, Udacity https://www.udacity.com/course/cloud-native-fundamentals--ud064
2. Introduction to GitOps, LF https://training.linuxfoundation.org/training/introduction-to-gitops-lfs169
3. K8s & Cloud Native Essentials, LF https://training.linuxfoundation.org/training/kubernetes-and-cloud-native-essentials-lfs250 $
4. Fundamentals of Open Source, LF https://training.linuxfoundation.org/training/fundamentals-of-open-source-it-and-cloud-computing-lfs200
5. Introduction to Cloud Infrastructure Technologies, edX https://www.edx.org/course/introduction-to-cloud-infrastructure-technologies
6. Introduction to Kubernetes, edX https://www.edx.org/course/introduction-to-kubernetes
7. Kube by Example, Red Hat https://kubebyexample.com
8. Free Kubernetes Training, Kasten.io https://learning.kasten.io
9. KubeAcademy, VMWare https://kube.academy
10. Learn Kubernetes, Civo https://www.civo.com/academy
11. CNCF Kubernetes and Cloud Native Associate Certification Course, Andrew Brown https://www.youtube.com/watch?v=ApIluksKvzI
12. CNCF Kubernetes and Cloud-Native Associate (KCNA), Exampro https://www.exampro.co/kcna $
13. Intro to Containers, K8s & OpenShift, IBM https://ibmdeveloper.skillsnetwork.site/courses/course-v1:IBMDeveloperSkillsNetwork+CC0201EN+v1
14. School of Kubernetes, Infracloud https://www.infracloud.io/kubernetes-school

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources (4)

## Public repositories

1. Official CNCF KCNA repository https://github.com/cncf/curriculum/tree/master/kcna
2. KCNA Curriculum https://github.com/cncf/curriculum/blob/master/KCNA_Curriculum.pdf
3. 100 days of DevOps https://github.com/100daysofdevops/Kubernetes-and-Cloud-Native-Associate/blob/main/README.md
4. Mohamed Abukar https://github.com/moabukar/Kubernetes-and-Cloud-Native-Associate-KCNA
5. Brad McCoy https://github.com/bradmccoydev/mentoring/blob/main/programs/kcna.md
6. Walid Shaari https://github.com/walidshaari/Kubernetes-and-Cloud-Native-Associate
    a. https://community.cncf.io/events/details/cncf-cloud-native-dojo-presents-kcna-prep-kubernetes-fundamentals-part-1
    b. https://community.cncf.io/events/details/cncf-cloud-native-dojo-presents-kcna-prep-kubernetes-fundamentals-part-2

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Curated resources (5)

## Exam preparation resources

1. Slack Communities
   a. Official Kubernetes workspace (kubernetes.slack.com), ***#kcna-exam-prep*** channel
   b. Cloud Native Mentoring workspace (cloudnative-mentoring.slack.com), ***#kcna*** channel
2. Important Instructions: KCNA (Linux Foundation) https://docs.linuxfoundation.org/tc-docs/certification/important-instructions-kcna
3. How to Pass your KCNA Exam (Brad McCoy) https://blog.bradmccoy.io/how-to-pass-your-kcna-exam-cf98cfa7d70f
4. Deep Dive KCNA Exam (CloudNativeFM) https://www.youtube.com/watch?v=wPbsvF_SGmc
5. What do you think about KCNA exam (Rob Muhlestein) https://www.youtube.com/watch?v=klDIH1y2-Wc
6. KCNA Exam — A quick guide (Marino Wijay)
   https://medium.com/@marino.wijay/the-kcna-exam-a-quick-guide-to-kicking-off-your-k8s-and-cloud-native-journey-56a3a5be345
7. KCNA quiz (100 days of DevOps)
   https://www.101daysofdevops.com/courses/kubernetes-and-cloud-native-essentials-lfs250/quizzes/kubernetes-and-cloud-native-essentials-exam
8. Mock Exam Questions (Mohamed Abukar) https://github.com/moabukar/Kubernetes-and-Cloud-Native-Associate-KCNA/tree/main/mock-exam-questions
9. KCNA Exam Study Guide (Ravikiran Srinivasulu) https://ravikirans.com/kcna-kubernetes-certification-exam-study-guide
10. KCNA Exam Notes (Pauline Narvas) https://www.youtube.com/watch?v=WbfvRuEyAJ0
11. The KCNA Book: Kubernetes and Cloud Native Associate https://www.amazon.com/KCNA-Book-Kubernetes-Native-Associate/dp/B09SNKVG5M

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# 3

# *Overview of external resources and Linux Foundation exam process*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Exam process

## Step 1 of 4: Purchase the online exam

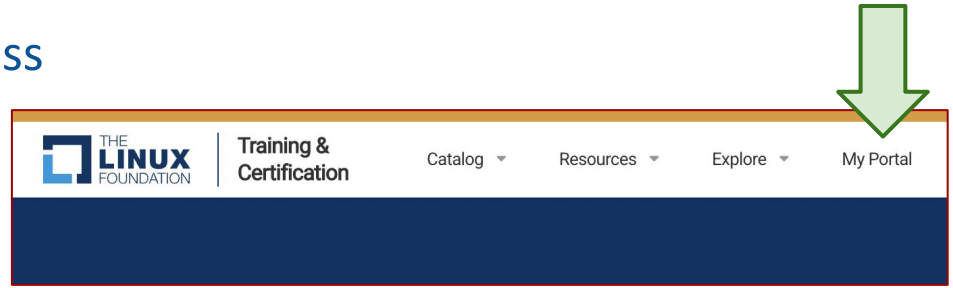
1. Go to https://training.linuxfoundation.org
2. Explore the catalog section, search for KCNA resources
   a. Exam URL: https://training.linuxfoundation.org/certification/kubernetes-cloud-native-associate
   b. Exam cost: $250
3. Click on “enroll today”
4. Sign up (if it is your first time at Linux Foundation’ training portal)
5. Complete the payment process

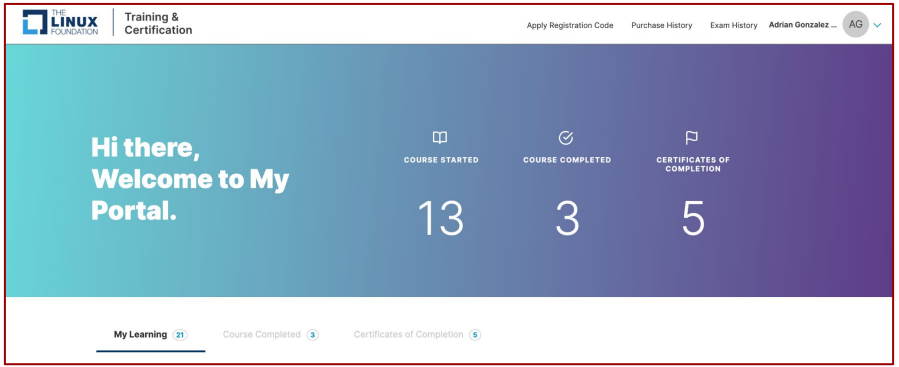
**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

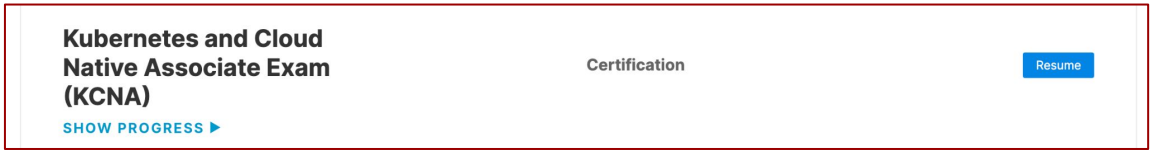
# Exam process

## Step 2 of 4: Follow the online exam process

1. Go to https://training.linuxfoundation.org
2. Select “My Portal”
3. Sign in
4. Find the KCNA exam from your “My Learning” dashboard
5. Click on “Resume”



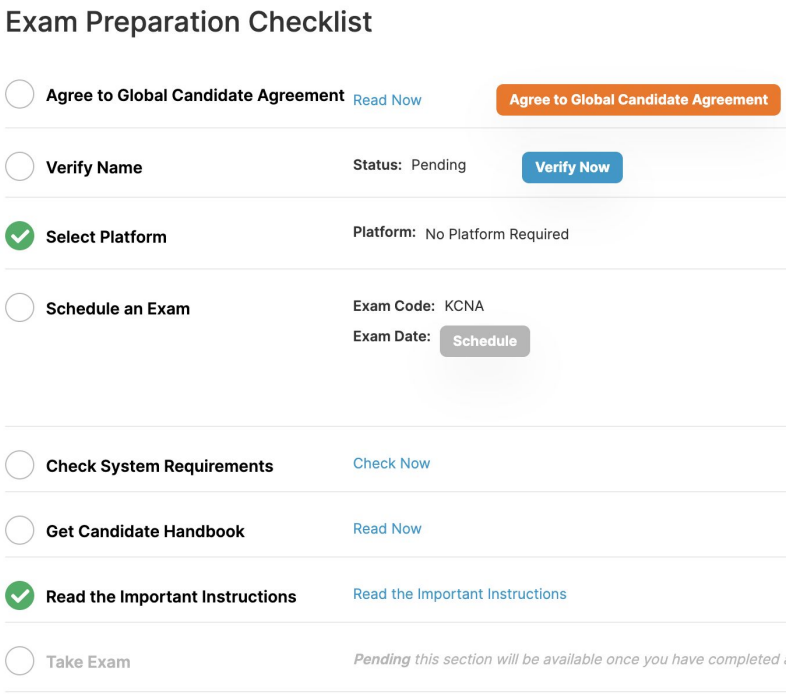

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Exam process

## Step 3 of 4: Book the online exam date/time

6. Explore the checklist
7. Read the handbook

   https://docs.linuxfoundation.org/tc-docs/certification/lf-handbook2
8. Schedule your exam *(this option is only available once you complete preliminary steps, ~30 minutes after)*




**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Exam process

## Step 4 of 4: Get ready before the exam

1. Prepare a valid ID
2. Follow the security procedures from PSI exams. i.e:
    a. Prepare the exam environment - You must take your exam in a private room, free from distractions.
    b. To check your system compatibility https://syscheck.bridge.psiexams.com (try to use a Chromium browser)
    c. Get ready to start in time - You can launch the exam within 30 minutes of your appointment time.
    d. Verify technical details
        i. Your location must support Voice over Internet Protocol (VoIP) in order to complete the exam.
        ii. Corporate firewalls and VPNs may restrict user access. It is recommended that you use a personal device.
3. Create a Credly account with the same email you are using for Linux Foundation portal

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Quick question (6)

## Which of these are not actual requirements for the KCNA exam?

a. A Linux Foundation account

b. A laptop

c. Kubernetes knowledge

d. A CNCF membership

e. A Github profile

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# What's next?

## My recommendations for you to pass the KCNA exam

*Exam tip*

1. Plan from 5 to 15 hours of self-study, depending on your current level of knowledge.
2. Make sure you understand each of the KCNA exam sections, and try to focus on those that are new for you. Don't forget to explore the CNCF landscape and the detailed guide.
3. Review the list of curated resources, especially Github repositories from Kubernetes community experts… they are gold!
4. Join the CNCF and Kubernetes open communities, it will help you be aware of what's going on in terms of new tools, experts, available resources, etc.

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# What's next?

## Make sure you follow these super MVPs

- Andrew Brown https://www.linkedin.com/in/andrew-wc-brown
- Katie Gamanji https://www.linkedin.com/in/katie-gamanji
- Walid Shaari https://www.linkedin.com/in/walidshaari
- Brad McCoy https://www.linkedin.com/in/bradmccoy3
- Mohamed Abukar https://www.linkedin.com/in/mohamed-abukar
- Kunal Kushwaha https://www.linkedin.com/in/kunal-kushwaha

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Q&A

3 of 3



**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

# Good luck… and share your exam experience!

## #KCNAExamPrep

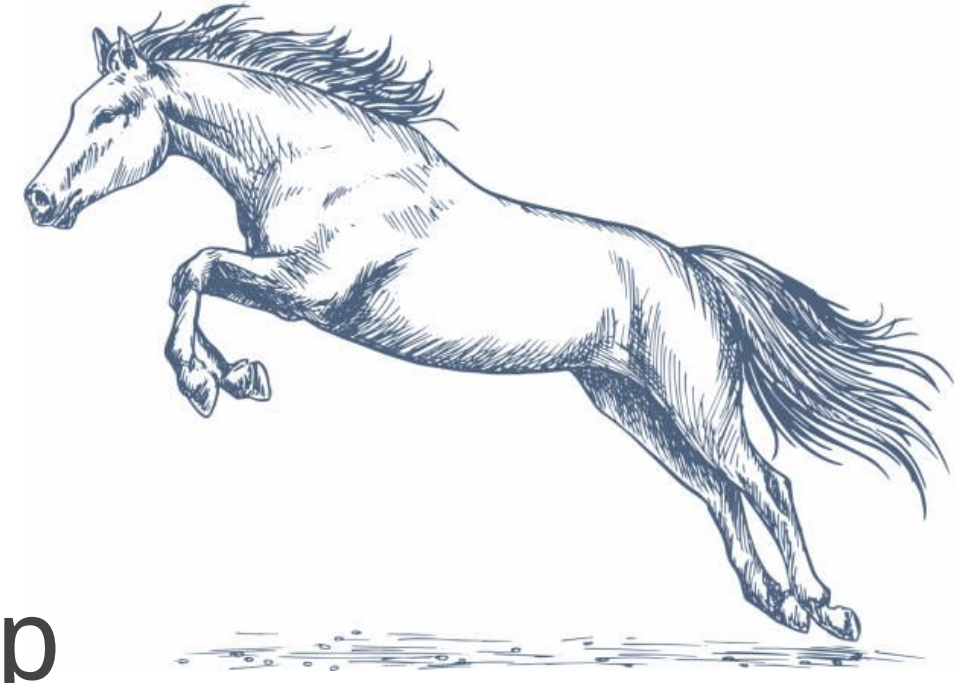
**Ping me and/or share feedback via**
www.linkedin.com/in/adriangs86
https://twitter.com/adriangs86
or https://github.com/AdrianGonzalezSanchez

+ post your exam result using the **#KCNAExamPrep**,
and tagging @ **"O'Reilly Media"** and **"Adrian Gonzalez Sanchez"**

*Good luck!*

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved

O'REILLY®



*Live Online Training*

# KCNA Exam Prep

**April 4, 2022**

Adrian Gonzalez Sanchez

**Copyright © 2022, Adrian Gonzalez Sanchez**
Reproduction is forbidden unless authorized. All rights reserved