



# Einführung in Kubernetes



# Wer sind wir? 🤝

Seit 2019 bietet DevNinjas Beratung in DevOps- und Cloud-Technologien. Unser Team unterstützt Unternehmen bei der effizienten Gestaltung und Skalierung ihrer IT-Infrastruktur durch moderne Lösungen wie Kubernetes.

- Unsere Mission: Beschleunigung der digitalen Transformation durch innovative DevOps-Lösungen 🚀.
- Unser Team: Erfahrene Experten in den Bereichen Cloud, Containerisierung und IT-Automatisierung 👤👤.
- Unsere Dienstleistungen: Beratung, Implementierung und Schulung in modernen DevOps-Technologien 📚.
- Unsere Vision: Führend in der Bereitstellung effizienter, skalierbarer und sicherer IT-Lösungen sein ✨.



# Was Sie erwartet

Diese Präsentation bietet einen kompakten Überblick über Kubernetes. Sie erfahren die Grundlagen, Architektur und Kernfunktionen der führenden Plattform zur Container-Orchestrierung. Zudem besprechen wir praxisnahe Anwendungen, DevOps-Integrationen und Sicherheitsaspekte.

- Grundlagen verstehen: Einführung und Kernkonzepte von Kubernetes .
- Architektur entdecken: Überblick über die interne Struktur und Komponenten von Kubernetes .
- Funktionalitäten erleben: Automatische Bereitstellung, Skalierung und Netzwerkfunktionen .
- Sicherheit und Skalierbarkeit: Erfahren, wie Kubernetes Anwendungen schützt und skaliert .
- Praxisbeispiele: Einblicke in erfolgreiche Kubernetes-Anwendungsfälle .
- Integration und Tools: Nutzung von CI/CD-Tools und Cloud-Services mit Kubernetes .

# Einführung in Kubernetes

Kubernetes ist eine leistungsstarke Open-Source-Plattform zur Automatisierung der Bereitstellung, Skalierung und Verwaltung von Container-Anwendungen.

Entwickelt von Google und jetzt von der CNCF unterstützt, bietet es eine robuste Lösung für die Verwaltung moderner Anwendungsarchitekturen in dynamischen Umgebungen.

- Unterstützt Multi-Cloud-Umgebungen für flexible Anwendungsarchitekturen Integriert nativ mit Docker-Containern für einfache Anwendungsportabilität
- Bietet APIs für umfassende Automatisierung und Anpassung.

# Kubernetes Architektur – Überblick

Kubernetes besteht aus Master- und Worker-Nodes für die zentrale Steuerung und Ausführung von Containern.

Der Master-Node beinhaltet den API-Server, etcd zur Speicherung des Clusterzustands, Scheduler zur Platzierung von Pods und den Controller-Manager zur Überwachung der Cluster-Statusänderungen. Die Worker-Nodes führen Container über Kubelet und Container-Runtime aus.

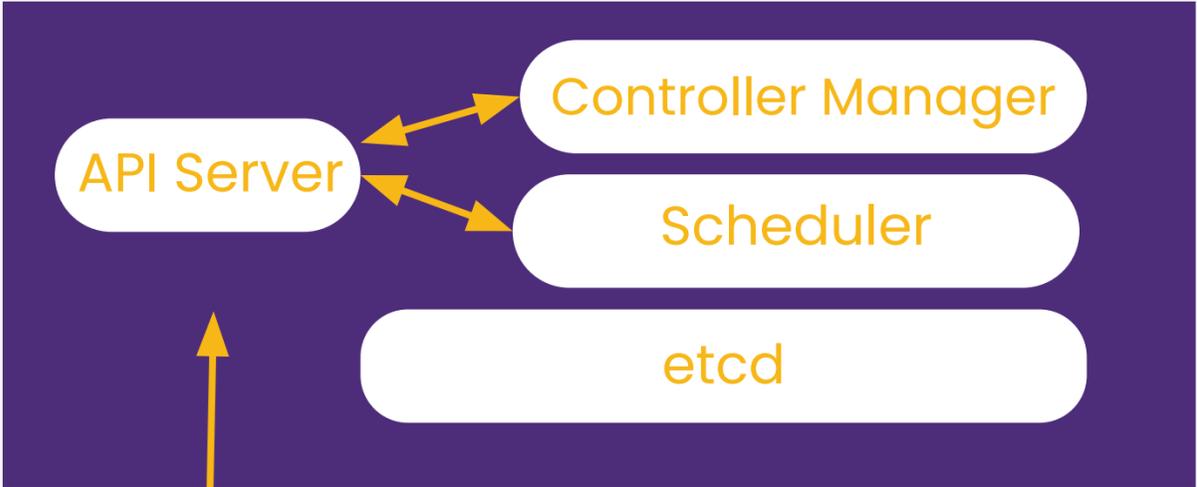
- Verwendet etcd zur Speicherung des Clusterzustands für hohe Skalierbarkeit.
- Scheduler platziert Pods basierend auf Ressourcenanforderungen und Policies.
- Controller-Manager überwacht die Cluster-Statusänderungen und steuert Ressourcen.

# Was ist Kubernetes?

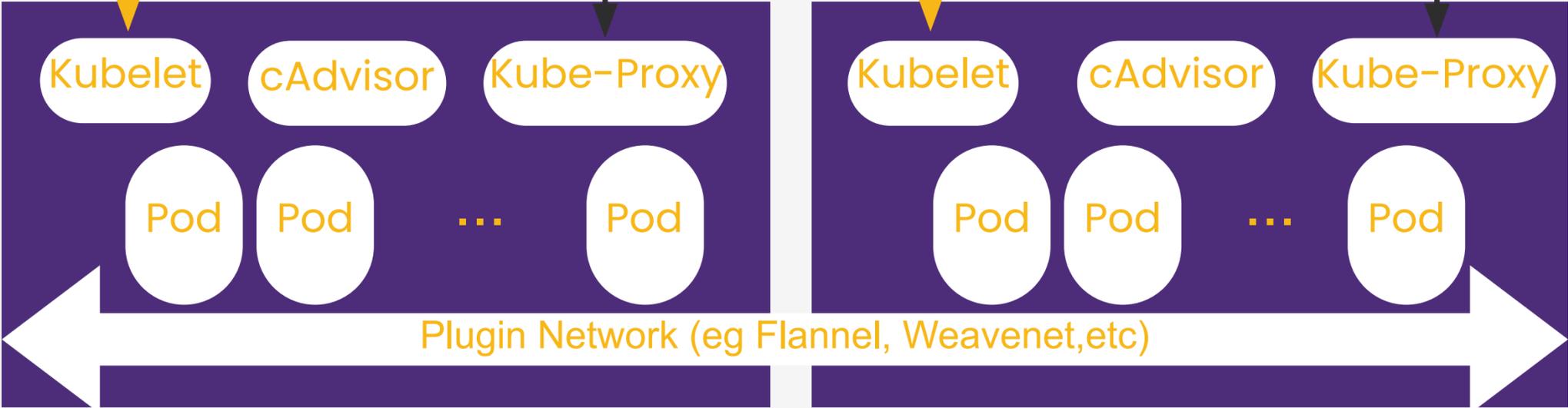
Developer



Control-Plan-Node



Users



Worker Node

Worker Node

# Kubernetes Vorteil vs. Nachteile

- ✓ Läuft in der Cloud
  - ✓ Kann auf öffentlichen und privaten Servern sowie in der Cloud ausgeführt werden
  - ✓ Wird von sehr vielen großen Unternehmen genutzt
  - ✓ Selbstheilung
  - ✓ Automatische replikation
  - ✓ Automatische Skalierung
  - ✓ Open Source
- ✗ Sehr schwierig einzurichten, wenn kein Cloud
  - ✗ Schwieriger zu erlernen als Docker Swarm
  - ✗ Keine Docker CLI



DevNinjas

## Automatische Bereitstellung und Rollbacks

Kubernetes automatisiert die Bereitstellung von Anwendungsänderungen und ermöglicht rollierende Updates für kontinuierliche Verfügbarkeit. Bei Bedarf können auch Rollbacks durchgeführt werden, um schnell auf Probleme oder Fehlkonfigurationen zu reagieren und die Verfügbarkeit der Anwendungen sicherzustellen.

- Verwendet Rolling-Update-Strategien für unterbrechungsfreie Anwendungsupdates.
- Unterstützt Deployment-Revisionen für schnelle Rückkehr zu vorherigen Konfigurationen.
- Integriert Health-Checks zur automatischen Überprüfung der
- Anwendungsverfügbarkeit.

## Netzwerkfunktionalitäten

Kubernetes bietet verschiedene Netzwerkmodelle wie Cluster-Interne Kommunikation (ClusterIP) für interne Services, Externe Dienstverfügbarkeit (NodePort) für externe Zugriffe und Netzwerkisolierung durch Network Policies.

Diese Funktionen ermöglichen eine flexible und sichere Kommunikation zwischen den Anwendungen und Services in einem Kubernetes-Cluster.

- ClusterIP für sichere, interne Kommunikation zwischen Pods.  
NodePort zur Exposition von Diensten außerhalb des Clusters.
- Network Policies zur Definition von Sicherheitsrichtlinien für den Netzwerkzugriff.

## Ressourcenverwaltung ⚡

Kubernetes ermöglicht eine effiziente Ressourcenverwaltung durch die Definition von Ressourcenanforderungen und -limits für Pods.

Dies optimiert die Nutzung der Cluster-Ressourcen und verbessert die Leistungsfähigkeit der Anwendungen, indem Engpässe frühzeitig erkannt und behandelt werden.

- Definition von CPU- und Speicheranforderungen für Pods.
- Automatische Skalierung basierend auf Ressourcenverbrauch.
- Integration von Horizontal Pod Autoscaler für dynamische Anpassungen.

## Skalierbarkeit und Ausfallsicherheit

Kubernetes bietet automatische Skalierungsmöglichkeiten, wie den Horizontal Pod Autoscaler, der die Anzahl der Pods basierend auf der CPU-Auslastung oder anderen Metriken anpasst.

Durch die Selbstheilungsfunktionen und die Fähigkeit zur automatischen Wiederherstellung unterstützt Kubernetes eine hohe Verfügbarkeit und Ausfallsicherheit für Anwendungen.

- Automatische Skalierung zur Bewältigung von Spitzenlasten.
- Selbstheilung zur Wiederherstellung nach Ausfällen.
- Einsatz von Readiness und Liveness Probes für Anwendungsintegrität.

## Anwendungsbeispiele

Kubernetes wird für eine Vielzahl von Anwendungen genutzt, darunter skalierbare Webanwendungen, datenintensive Verarbeitungsaufgaben und komplexe Microservices-Architekturen.

Große Unternehmen wie Spotify und Airbnb setzen Kubernetes erfolgreich ein, um ihre Dienste zu skalieren und die Verfügbarkeit zu verbessern.

- Unterstützung für Continuous Deployment in Cloud-Umgebungen.
- Verwendung von Kubernetes für Big Data und KI-Anwendungen.
- Einsatz in DevOps-Umgebungen zur Beschleunigung der Softwarebereitstellung.

# CI/CD Integration

Kubernetes bietet nahtlose Integrationen in CI/CD-Pipelines, was automatisierte Bereitstellungen, rollierende Updates und Skalierung ermöglicht.

Diese Integration optimiert den Entwicklungsprozess und fördert eine agile und zuverlässige Softwarebereitstellung.

- Unterstützung für Jenkins, GitLab CI/CD und andere CI/CD-Tools.
- Automatisierung von Build-, Test- und Bereitstellungsprozessen.
- Integration von Container-Registern wie Docker Hub und AWS ECR.

## Kubernetes und Cloud-Anbieter

Kubernetes harmonisiert die Container-Orchestrierung über verschiedene Cloud-Plattformen wie AWS (EKS), Google Cloud (GKE) und Azure (AKS).

Es nutzt spezifische Cloud-Funktionen zur Verbesserung der Skalierbarkeit, Sicherheit und Leistung von Anwendungen.

- Unterstützung für hybride Cloud-Umgebungen und Multi-Cloud-Strategien.
- Integration von Cloud-spezifischen Diensten wie AWS RDS und Google Cloud BigQuery.
- Nutzung von Cloud-Provider-Services zur Optimierung der Infrastrukturkosten.

## Sicherheitsaspekte

Kubernetes implementiert fortschrittliche Sicherheitsmaßnahmen wie Role-Based Access Control (RBAC), Network Policies und Secrets Management.

Diese Funktionen schützen Anwendungen und Daten vor unautorisierten Zugriffen und verbessern die Gesamtsicherheit in Kubernetes-Clustern.

- Einsatz von RBAC zur feingranularen Zugriffssteuerung auf Kubernetes-Ressourcen.
- Verwendung von Network Policies zur Segmentierung von Netzwerkzugriffen. V
- erhaltung und Sicherung von Geheimnissen wie API-Schlüsseln und Passwörtern.

## Monitoring und Logging

Kubernetes bietet umfassendes Monitoring mit Tools wie Prometheus für Metriken und Grafana für Visualisierungen.

Der ELK Stack (Elasticsearch, Logstash, Kibana) ermöglicht zentrales Logging und detaillierte Protokollanalysen, die für die Fehlerbehebung und Leistungsoptimierung unerlässlich sind.

- Nutzung von Prometheus für das Monitoring von Anwendungs- und Systemmetriken.
- Einsatz von Grafana zur Erstellung benutzerdefinierter Dashboards und Visualisierungen.
- Verwendung von ELK Stack für zentrales Logging und Protokollanalyse in Kubernetes-Clustern.

## Integration von Storage-Lösungen

Kubernetes bietet integrierte Lösungen zur Speicherverwaltung, einschließlich Persistenter Volumes (PVs) und Persistenter Volumenansprüche (PVCs).

Diese ermöglichen die Speicherung von Daten über Container-Lebenszyklen hinweg und unterstützen die Datenpersistenz und -verwaltung in Kubernetes-Clustern.

- Verwendung von PVs für die dynamische Bereitstellung und Verwaltung von Speicherressourcen.
- Integration von PVCs für die flexible Zuweisung und Nutzung von Speicher in Pods.
- Unterstützung für Cloud-basierte Speicherlösungen wie AWS EBS und Google Persistent Disk.

## Multi-Cluster-Management

Kubernetes erleichtert das Management von mehreren Clustern mit Tools wie Rancher und OpenShift.

Diese bieten eine zentrale Verwaltung und Überwachung, um die Konsistenz über verschiedene Cloud-Infrastrukturen hinweg sicherzustellen und die Effizienz der Kubernetes-Bereitstellungen zu verbessern.

- Verwaltung mehrerer Kubernetes-Cluster über eine gemeinsame Benutzeroberfläche.
- Einsatz von Rancher zur Automatisierung von Cluster-Provisioning und Konfiguration.
- Nutzung von OpenShift für Sicherheitsmanagement und Compliance in Multi-Cluster-Umgebungen.

## Warum Kubernetes-Derivate? 💡

Kubernetes-Derivate bieten maßgeschneiderte Lösungen für spezifische Anwendungsfälle und Branchenanforderungen. Diese Abwandlungen von Kubernetes optimieren die Plattform durch zusätzliche Funktionen, die über die Grundfunktionen hinausgehen.

Unternehmen profitieren von vereinfachten Installationsprozessen, speziellen Sicherheitsmerkmalen und erweiterten Automatisierungsoptionen, die es ermöglichen, spezifische Anforderungen effizient zu erfüllen und die IT-Infrastruktur besser zu gestalten.

- Spezifische Anpassungen: Maßgeschneiderte Lösungen für verschiedene Branchen und Anwendungsfälle
- Erweiterte Funktionen: Zusätzliche Tools und Integrationen, die über das Basis-Kubernetes hinausgehen
- Verbesserte Effizienz: Einfachere Installation, spezielle Sicherheitsmerkmale und erweiterte Automatisierungsoptionen.

## Red Hat OpenShift

Red Hat OpenShift ist vollständig unterstützt und produktionsreif. Es bietet eingebaute CI/CD-Tools für eine effiziente Softwarebereitstellung und ist auf Red Hat Enterprise Linux basiert. Die Plattform integriert Entwickler- und Operations-Tools nahtlos, um Teams eine umfassende Lösung für die Container-Orchestrierung zu bieten.

- **Vollständig unterstützt und produktionsreif:** Red Hat OpenShift wird aktiv unterstützt und ist für den Einsatz in Produktionsumgebungen geeignet. Es bietet Sicherheit und Zuverlässigkeit für geschäftskritische Anwendungen
- **Eingebaute CI/CD-Tools:** Mit integrierten Continuous Integration und Continuous Deployment (CI/CD)-Tools automatisiert OpenShift den Software-Bereitstellungsprozess
- **Basierend auf Red Hat Enterprise Linux:** Als Grundlage verwendet OpenShift die bewährte Red Hat Enterprise Linux Plattform, die Sicherheit und Stabilität gewährleistet
- **Integrierte Entwickler- und Operations-Tools:** OpenShift bietet eine nahtlose Integration von Entwickler- und Operations-Tools, um die Effizienz bei der Verwaltung und Skalierung von Container-Anwendungen zu steigern





Rancher bietet umfassendes Mehrcluster-Management für eine effiziente Verwaltung mehrerer Kubernetes-Distributionen. Es erleichtert das Benutzermanagement und integriert Monitoring- und Logging-Lösungen, um eine nahtlose Überwachung und Verwaltung zu gewährleisten.

- Mehrcluster-Management: Rancher ermöglicht die zentrale Verwaltung mehrerer Kubernetes-Cluster für eine effiziente Ressourcennutzung
- Unterstützung für mehrere Kubernetes-Distributionen: Es unterstützt verschiedene Kubernetes-Distributionen und bietet Flexibilität bei der Wahl der Infrastruktur
- Einfaches Benutzermanagement: Rancher vereinfacht das Benutzermanagement durch übersichtliche Zugriffsrechte und Benutzerverwaltung
- Integrierte Monitoring- und Logging-Lösungen: Durch integrierte Monitoring- und Logging-Lösungen bietet Rancher umfassende Einblicke in die Cluster-Performance und Ereignisprotokolle



## Google Kubernetes Engine (GKE)

Google Kubernetes Engine (GKE) ist ein vollständig verwalteter Kubernetes-Dienst von Google Cloud. Er bietet automatische Upgrades und Skalierungsmöglichkeiten sowie eine nahtlose Integration mit anderen Google Cloud-Diensten. GKE zeichnet sich durch starke Sicherheitsfunktionen aus, die eine sichere Bereitstellung und Verwaltung von Container-Workloads gewährleisten.

- Vollständig verwalteter Kubernetes-Dienst von Google Cloud: GKE bietet eine verwaltete Kubernetes-Plattform in der Google Cloud-Umgebung
- Automatische Upgrades und Skalierung: Automatisierte Upgrades der Kubernetes-Versionen und Skalierung der Ressourcen je nach Bedarf
- Integration mit anderen Google Cloud-Diensten: Nahtlose Integration mit anderen Diensten der Google Cloud, um zusätzliche Funktionen und Services zu nutzen
- Starke Sicherheitsfunktionen: GKE bietet umfassende Sicherheitsmaßnahmen wie IAM-Rollen, Netzwerkrichtlinien und verschlüsselte Datenübertragung



DevNinjas

## Amazon EKS (Elastic Kubernetes Service)

Amazon EKS (Elastic Kubernetes Service) ist ein verwalteter Kubernetes-Dienst von AWS, der eine nahtlose Integration mit AWS-Services bietet. Die Plattform gewährleistet hohe Verfügbarkeit und Skalierbarkeit für Kubernetes-Cluster und unterstützt sowohl EC2-Instanzen als auch AWS Fargate für eine flexible Ressourcenverwaltung.

- **Verwalteter Kubernetes-Dienst von AWS:** Amazon EKS bietet eine verwaltete Kubernetes-Plattform in der AWS-Umgebung
- **Integration mit AWS-Services:** Nahtlose Integration mit anderen AWS-Services, um zusätzliche Funktionen und Services zu nutzen
- **Hohe Verfügbarkeit und Skalierbarkeit:** EKS gewährleistet hohe Verfügbarkeit und Skalierbarkeit für Kubernetes-Cluster, um die Leistungsfähigkeit zu maximieren
- **Unterstützung für EC2 und Fargate:** Unterstützt sowohl EC2-Instanzen als auch AWS Fargate für eine flexible Ressourcenverwaltung und Betriebskostenoptimierung

## Azure Kubernetes Service (AKS)

Azure Kubernetes Service (AKS) ist ein verwalteter Kubernetes-Dienst von Microsoft Azure, der eine nahtlose Integration mit Azure-Tools bietet. Die Plattform bietet automatische Upgrades und Patching für Kubernetes-Cluster sowie eine einfache Netzwerk- und Sicherheitskonfiguration für eine effiziente Verwaltung.

- **Verwalteter Kubernetes-Dienst von Microsoft Azure:** Azure Kubernetes Service bietet eine verwaltete Kubernetes-Plattform in der Microsoft Azure-Umgebung
- **Nahtlose Integration mit Azure-Tools:** AKS bietet eine nahtlose Integration mit anderen Azure-Tools, um zusätzliche Funktionen und Services zu nutzen
- **Automatische Upgrades und Patching:** Automatisierte Upgrades und Patching für Kubernetes-Cluster, um die Plattform stets auf dem neuesten Stand zu halten
- **Einfache Netzwerk- und Sicherheitskonfiguration:** AKS erleichtert die Netzwerk- und Sicherheitskonfiguration für eine effiziente Verwaltung und sichere Bereitstellung von Anwendungen

## VMware Tanzu

VMware Tanzu ermöglicht die Bereitstellung und Verwaltung von Kubernetes-Clustern auf VMware-Infrastruktur. Als integrierte Plattform für Anwendungslieferung bietet Tanzu Unterstützung für Multi-Cloud-Umgebungen und integrierte Sicherheits- sowie Governance-Funktionen für eine umfassende Kontrolle.

- Bereitstellung und Verwaltung von Kubernetes-Clustern auf VMware-Infrastruktur: VMware Tanzu ermöglicht die einfache Bereitstellung und Verwaltung von Kubernetes-Clustern auf VMware-Umgebungen
- Integrierte Plattform für Anwendungslieferung: Tanzu bietet eine integrierte Plattform für die schnelle und zuverlässige Bereitstellung von Anwendungen
- Multi-Cloud-Unterstützung: Unterstützt Multi-Cloud-Umgebungen für Flexibilität bei der Wahl der Cloud-Plattform
- Sicherheits- und Governance-Funktionen: Bietet umfassende Sicherheits- und Governance-Funktionen für eine sichere und regelkonforme Anwendungsbereitstellung

## Eigene Cloud mit Kubernetes

Kubernetes bietet die Möglichkeit, ein eigenes Cloud-System zu betreiben, indem es als Betriebssystem für die Cloud genutzt wird. Dies ermöglicht Unternehmen, ihre Infrastruktur flexibel und kosteneffizient zu skalieren, oft ohne Lizenzkosten. Es reduziert die Abhängigkeit von proprietärer Software und eignet sich hervorragend für On-Premises-Implementierungen.

- Kubernetes als Betriebssystem für eigene Cloud möglich: Nutzung von Kubernetes, um eine flexible und skalierbare Cloud-Infrastruktur zu schaffen
- Keine oder geringe Lizenzkosten: Oft keine zusätzlichen Lizenzkosten für die Verwendung von Kubernetes im eigenen Cloud-System
- Keine Abhängigkeit von proprietärer Software: Reduziert die Abhängigkeit von proprietären Lösungen und ermöglicht eine größere Flexibilität und Kontrolle über die Infrastruktur
- Auch gut für On-Premises-Implementierungen: Geeignet für lokale, On-Premises-Implementierungen, um die Cloud direkt im eigenen Rechenzentrum zu betreiben

## Zusammenfassung

Kubernetes-Derivate bieten spezialisierte Funktionen, die verschiedenen Bedürfnissen gerecht werden. Die Wahl des Derivats hängt von spezifischen Anwendungsfällen und individuellen Anforderungen ab. Die nahtlose Integration in bestehende Ökosysteme spielt dabei eine entscheidende Rolle für den Erfolg der Implementierung.

- Spezialisierte Funktionen für verschiedene Bedürfnisse: Kubernetes-Derivate bieten spezifische Funktionen, die auf unterschiedliche Anforderungen zugeschnitten sind.
- Anwendungsfälle und individuelle Anforderungen: Die Auswahl des richtigen Derivats ist abhängig von den konkreten Anwendungsszenarien und den spezifischen Anforderungen des Unternehmens.
- Integration in bestehende Ökosysteme: Die nahtlose Integration in bestehende IT-Infrastrukturen und Ökosysteme ist ein entscheidender Faktor für den Erfolg der Implementierung von Kubernetes-Derivaten.

# Java App in K8s!





# Scan mich!





# Danke für Ihre Aufmerksamkeit Fragen?

