

CLOUD-NATIVE- ENTWICKLUNG FÜR SOFTWAREHERSTELLER



ERLEBEN, WAS VERBINDET.

In Kooperation mit



Inhaltsverzeichnis

1. Vorwort	03
2. Was bedeutet Cloud-Native?	04
3. So profitieren Sie als Software-Entwickler von Cloud-Native	06
4. Mehrwerte für Ihr Unternehmen und Ihre Kunden	07
5. So gelangen Sie in Etappen zu Cloud-Native	08
6. Meistern Sie die Herausforderungen von Cloud-Native – es lohnt sich	11
7. Cloud-Native und Azure	13
8. Im Partnernetzwerk gemeinsam zum Erfolg	16
9. Wie white duck Ihnen hilft	17

1. VORWORT



Dirk Heidemann

Telekom GK Partnermanagement

E-Mail:

reseller@telekom.de

LinkedIn:

www.linkedin.com/in/dirkheidemann

Sie wollen mehr über das Telekom Microsoft Cloud Solution Partner Programm CSP² wissen? Dann nehmen Sie am besten direkten Kontakt mit uns auf.

Liebe Leserinnen und Leser,

wenn man als Cloud Solution Partner nachhaltig erfolgreich sein will, dann führt kein Weg daran vorbei, eigene Services zu entwickeln. Das können eigene Dienstleistungsangebote sein oder selbst entwickelte Software-Anwendungen. Dabei gilt: Je mehr eigene Wertschöpfung das Angebot enthält und je einzigartiger es ist, desto höhere Margen können erzielt werden. Im Bereich Cloud-Anwendungsentwicklung liegen die Margen laut Studien von Forrester Research im Durchschnitt bei 45 bis 50 Prozent.

In diesem Whitepaper geben Ihnen die Experten unseres erfolgreichen Telekom Cloud Solution Partners **white duck** einen Überblick zu Cloud-Native Software und deren Entwicklung auf der Basis ihrer eigenen, langjährigen Praxiserfahrung. Sie geben nicht nur Empfehlungen, sondern zeigen auch Best Practices rund um die Konzeption und Entwicklung von Cloud-Native-Anwendungen auf und stellen den Nutzen von Azure als Plattform für Cloud-Native-Anwendungen vor. Jedes Kapitel wird aus der Business-Sicht – durch Markus Sümichen – und auf der technischen Ebene – durch Nico Meisenzahl – betrachtet. Dabei steht weniger die Theorie, sondern vielmehr die Projekterfahrung der beiden Experten im Mittelpunkt, wodurch das komplexe Thema deutlich greifbarer wird.

Viel Freude beim Lesen und wertvolle Erkenntnisse für Ihr Business wünscht Ihnen,

Ihr Dirk Heidemann

2. WAS BEDEUTET CLOUD-NATIVE?



Markus Sümmchen
CEO & Founder,
white duck GmbH

Twitter:
[@msuemmchen](https://twitter.com/msuemmchen)

LinkedIn:
[linkedin.com/in/msuemmchen](https://www.linkedin.com/in/msuemmchen)

Markus:

Cloud-Native ist der nächste logische Schritt nach den Entwicklungen der letzten 10 bis 15 Jahre im Bereich der Softwareentwicklung. Von SOA (service-orientierter Architektur) über Microservices, Virtualisierung, Containerisierung, dem Hosting in Rechenzentren bis hin zu Cloud Computing ist Cloud-Native die Essenz einer langen Entwicklungsgeschichte. Cloud-Native ermöglicht es nun, skalierbare, einfach zu wartende und mehrmandantenfähige Cloud-Lösungen nach definierten Architek-

tur- und Betriebsmustern in einem durchgängigen Prozess (DevOps) zu entwickeln und zu betreiben. Es beschreibt also ein technisches und methodisches Gesamtkonzept für den kompletten Lebenszyklus von modernen Cloud-Anwendungen. Für Sie als Softwarehersteller ändert sich damit Ihr Entwicklungsprozess, aber auch die Art und Weise, wie Sie Ihre Software bereitstellen. Denn Cloud-Native ermöglicht Ihnen die Erweiterung Ihres Geschäftsmodells um den Betrieb Ihrer Softwarelösungen für Ihre Endkunden. Sie sind nicht mehr nur Softwarelieferant, sondern auch Softwarebetreiber.



Nico Meisenzahl
Senior Cloud & DevOps
Consultant, Cloud &
Data Management MVP,
white duck GmbH

Twitter:
[@nmeisenzahl](https://twitter.com/nmeisenzahl)

LinkedIn:
[linkedin.com/in/nicomeisenzahl](https://www.linkedin.com/in/nicomeisenzahl)

Nico:

Der Begriff „Cloud-Native“ wurde von der Cloud Native Computing Foundation (CNCF) ins Leben gerufen. Unter Cloud-Native versteht man Anwendungen die basierend auf modernen, dynamischen und skalierbaren Public-, Private- oder Hybrid-Cloud-Lösungen entwickelt werden. Der Einsatz von Containern, Microservices, deklarativen APIs, Automatisierungen sowie unveränderlichen Infrastrukturen bilden den Kern dieser Art der Anwendungsentwicklung. Auf diese Weise lassen sich lose gekoppelte, belastbare und einfach zu verwaltende Anwendungen entwickeln und betreiben.

Cloud-Native setzt hierbei zudem einen Schwerpunkt auf eine agile Arbeitsweise, um neue Funktionen schnell und mit minimalem Arbeitsaufwand bereitstellen zu können. Die CNCF treibt die Verbreitung von Cloud-Native voran, indem sie ein Ökosystem von herstellernerneut-neutralen Open-Source-Projekten fördert und unterhält. Alle wichtigen Eckpunkte von Cloud-Native beschreibt die **Cloud Native Trail Map** der CNCF.

DIE CLOUD NATIVE TRAIL MAP

Hier klicken für Detailansicht

- CONTAINERISIERUNG

Container helfen die Anwendungen und ihre Abhängigkeiten deklarativ zu bündeln. Hierbei sollte Ihre Anwendung nach funktionalen Schwerpunkten in einzelne Microservices unterteilt werden.

- CONTINUOUS INTEGRATION / CONTINUOUS DELIVERY

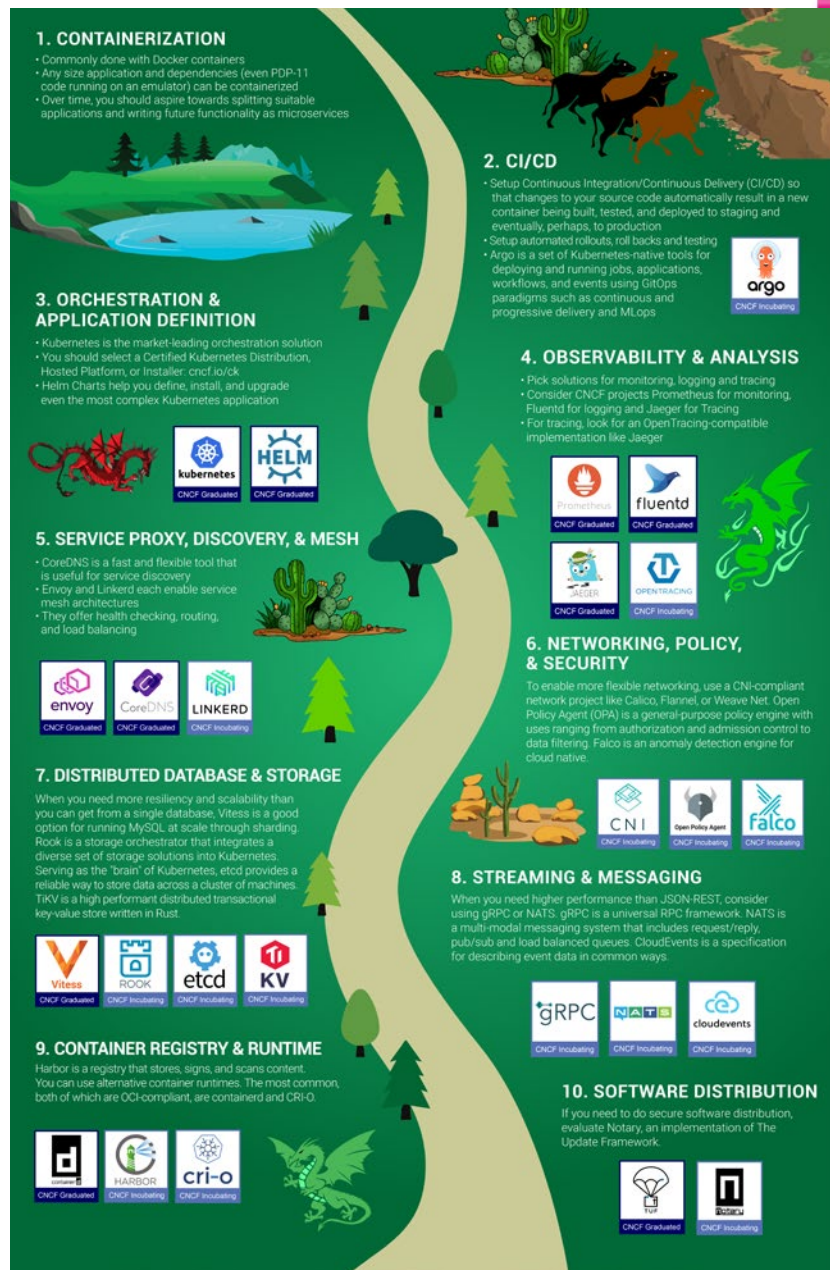
Automatisierung ist hierbei eine Kernkomponente und sollte auf Ihrer Cloud-Native-Reise von Anfang an betrachtet werden. Durch Automatisierung können neue Funktionen schnell und mit geringen Aufwänden bereitgestellt werden.

- ORCHESTRIERUNG

Eine Orchestrierung der Microservices ist essenziell, um einen fehlerfreien Betrieb der Anwendung sicherzustellen. Die Auswahl der Plattform sollte basierend auf den jeweiligen Voraussetzungen erfolgen. Kubernetes ist ein beliebter Orchestrator – jedoch nicht der Einzige.

- BEOBACHTBARKEIT & ANALYSE

Monitoring, Logging und Tracing sind die drei Säulen der Beobachtbarkeit. Sie helfen, Ihre Anwendungen erfolgreich zu betreiben und können Ihnen zudem tiefere Business-Metriken bereitstellen.



Neben diesen Kernthemen beschreibt die Cloud-Native Trail Map weitere wichtige Teilbereiche, die für verteilte Cloud-Native-Anwendungen essenziell sind.

3. SO PROFITIEREN SIE ALS SOFTWARE-ENTWICKLER VON CLOUD-NATIVE

Markus:

Mit den einzelnen Cloud-Native-Paradigmen wie DevOps, Microservices-Architektur, Containerisierung und Continuous-Integration / Continuous-Delivery (CI/CD) bekommen Sie als Softwarehersteller reife Methoden und Technologien an die Hand, um digitale, cloud-basierte Produkte zu entwickeln und zu vermarkten. Die Techniken sind inzwischen ausgereift, die Ökosysteme lebendig und mit Open Source erhalten sie eine nie da gewesene Dynamik.

Public-Cloud-Plattformen, wie z. B. Microsoft Azure, sind optimal auf die Cloud-Native-Entwicklung ausgerichtet und bieten alle Services, Frameworks und Tools, um erfolgreich Cloud-Native-Anwendungen zu entwickeln. Dank moderner Entwicklungsparadigmen, Tools und Frameworks erzielen Sie eine höhere Implementierungsgeschwindigkeit. Oftmals können außerdem bereits vorgefertigte Komponenten genutzt werden und die neuen Techniken passen perfekt zu neuen Anwendungsfällen wie beispielsweise Cloud, Software-as-a-Service (SaaS) oder Mobile.

Des Weiteren ist es für die Motivation des eigenen Entwicklungsteams enorm wichtig, moderne Entwicklungsparadigmen wie Cloud-Native zu nutzen, um Mitarbeiter und Kollegen auf einem aktuellen Wissensstand zu halten, aber auch, um neue Entwickler auf dem Markt zu gewinnen. Softwareentwicklung auf einem alten Technologie-Stack ist gerade bei der jüngeren Entwicklergeneration nicht mehr gefragt.

Nico:

Die Entwicklung von Cloud-Native-Anwendungen bringt Ihnen weitreichende Vorteile. Architekturen werden als ein System von miteinander verbundenen Microservices aufgebaut. Das erlaubt es Ihnen, einzelne Funktionen Ihrer Anwendung unabhängig voneinander zu betrachten. Dabei sind die einzelnen Microservices nicht nur voneinander entkoppelt, sondern auch von der Infrastruktur abstrahiert. Diese Entkopplung hilft Ihnen, auf notwendige Anpassungen schnell zu reagieren, um zum Beispiel eine neue Funktion auszuliefern und sie an einer bestimmten Kundengruppe zu testen, oder aber um einzelne Teilbereiche Ihrer Anwendung zu skalieren.

Diese und weitere Vorteile von Cloud-Native helfen Ihnen, Komplexität zu reduzieren und unterstützen Sie somit, schneller Mehrwerte zu generieren. Die Azure-Cloud sowie verschiedene Open-Source-Projekte bieten Ihnen hier die perfekte Grundlage. Mit Cloud-Native-Anwendungen machen Sie Ihr Unternehmen fit für die Zukunft. Sie verschaffen Ihnen einen Wettbewerbsvorteil, da sie sich mithilfe der skalierbaren und flexiblen Architektur besser auf die Differenzierung Ihres grundlegenden Geschäftsmodells konzentrieren können.

4. MEHRWERTE FÜR IHR UNTERNEHMEN UND IHRE ENDKUNDEN

Markus & Nico:

Dank seines neuen Technologie-Stacks und seiner neuen Methodik ist mit Cloud-Native eine neue Ära der Softwareentwicklung angebrochen, in der der Fokus noch stärker auf dem Roll-out und der Bereitstellung von Software über die Cloud und als SaaS-Angebot liegt. Auch in Zukunft werden immer mehr cloud-basierte Softwarelösungen und -produkte entstehen. Für Sie als Softwarehersteller besteht der Mehrwert darin, die neuen Distributionswege von Software effektiv zu nutzen und Ihre Lösung einfach skalierbar auf der gesamten Welt anbieten zu können. Ein weiterer positiver Effekt: Durch die Etablierung des Cloud-Native-Paradigmas in Ihrem Unternehmen werden Sie zu einem interessanten Arbeitgeber auf dem IT-Markt.

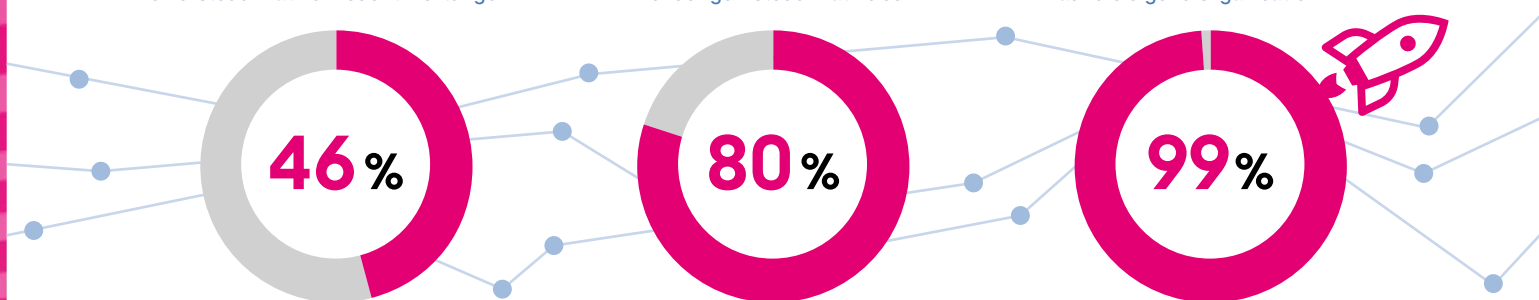
Aus Sicht Ihrer Endkunden positionieren Sie sich mit einer Cloud-Native-Agenda als ein Softwarelieferant, der auf der Höhe der Zeit agiert und auch in Zukunft ein verlässlicher Partner sein wird. Denn Ihre Kunden wollen Software einfach und unkompliziert konsumieren – wie sie es aus der Welt der Smartphone-Apps bald bereits seit 15 Jahren gewohnt sind.

Cloud-Native Adaptions-Level

Bereits 46% der Anwendungen sind Cloud-Native-Neuentwicklungen *

Bis 2023 sollen 80% der Anwendungen Cloud-Native sein **

DevOps hat einen positiven Einfluss auf die eigene Organisation ***



In Summe hilft Ihnen Cloud-Native, auf die Anforderungen des Marktes und Ihrer Kunden schnell und agil reagieren zu können. Mit der Umsetzung Ihrer eigenen Lösungen als Cloud-Native-Anwendungen steigern Sie Ihre Flexibilität, bei gleichzeitiger Reduzierung der Aufwände. So können Sie effizienter auf die Anforderungen Ihrer Endkunden eingehen und notwendige Funktionen mit einer geringen Time-to-Market bereitstellen. Durch den hohen Automatisierungsgrad von Cloud-Native und den Einsatz dezentraler Microservices reduzieren Sie außerdem die Komplexität und damit auch die Aufwände und Kosten in der Produktentwicklung.

Quellen: * IDG 2020 IDG Cloud Computing Study / ** IDC FutureScape: Worldwide Datacenter 2020 Predictions / *** Atlassian & CITE Research: 2020 DevOps Trends Survey

5. SO GELANGEN SIE IN ETAPPEN ZU CLOUD-NATIVE

Markus:

Aus der Sicht eines Softwareherstellers mit Bestandsprodukten gibt es im Wesentlichen zwei Szenarien, um in die Cloud-Native-Welt einzusteigen.

1. ABLÖSE EINES BESTANDSPRODUKTS DURCH EINE NEUENTWICKLUNG

Die Neuentwicklung mit dem Ziel der Ablöse eines Bestandsproduktes, das über viele Jahre gewachsen ist, ist sicherlich die größte Herausforderung für Softwarehersteller. Bei diesem Szenario muss ein bewährtes und oftmals sehr reifes Bestandsprodukt mithilfe neuer Techniken entwickelt oder besser noch nachgebaut werden. Bestandskunden erwarten mindestens die gleiche Funktionalität, bei einer noch moderneren und intuitiveren Bedienung, während der Migrationsaufwand gering sein muss. Diese Hauptanforderungen sind oftmals nur schwer zu realisieren und bedeuten auf dem Zeitstrahl nicht selten einen Entwicklungsaufwand von fünf Jahren oder mehr. Wenn das Bestandsprodukt und das dahinterstehende Business-Modell über diesen Zeitraum noch tragfähig sind und die nötigen Entwicklungskapazitäten – sowohl personell als auch wirtschaftlich – verfügbar sind, ist das die beste Art und Weise sein Business in die Zukunft zu bringen. In Summe ist das jedoch auch der deutlich aufwändigere Weg.

2. ENTWICKLUNG EINES NEUEN PRODUKTS FÜR EINE NEUE ZIELGRUPPE

Als einfacher umsetzbarer Ansatz steht dem die Entwicklung eines neuen Produktes gegenüber, das nicht mit dem Bestandsprodukt konkurriert und dennoch einen Mehrwert für die Bestandskundenbasis bietet, während es gleichzeitig neue Kundensegmente adressiert und somit neue Umsatzströme generiert. Im Optimalfall geht dieser Ansatz mit einer Transformations-Roadmap des eigenen Geschäftsmodells einher. Mithilfe dieses Ansatzes kann ein kleines Team auf der grünen Wiese beginnen. Es kann schnell kleine, neue Feature-Sets entwickeln und ausliefern, um damit erste Mehrwerte für die Kunden zu schaffen. Dafür eignen sich natürlich Cloud-Native-Anwendungen bestens.

Für beide Szenarios sollten Sie betrachten, inwiefern das Bestandsprodukt über einen Rehost-, Refactor- oder Rearchitect-Ansatz modernisiert und in die Cloud portiert werden kann. Mithilfe dieses evolutorisches Ansatzes können Sie Zeit gewinnen und schneller das Thema „Cloud-Lösung auf dem Markt“ adressieren. Allerdings sollten Sie sich dabei bewusst sein, dass dieser Ansatz immer mit Einschränkungen – auf der technischen Seite aufgrund alter Softwarearchitekturen und Techniken – oder auf der wirtschaftlichen Seite – bedingt durch höhere Kosten für Cloud-Ressourcen – einhergehen kann.

Nico:

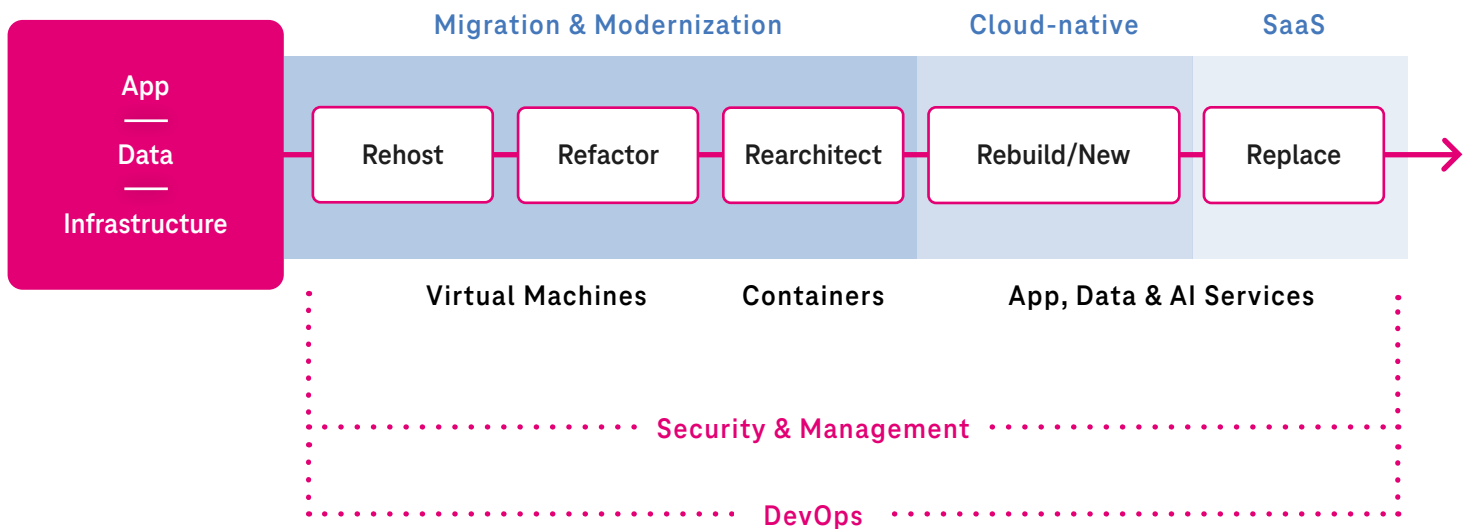
Sie können Ihre Cloud-Native-Reise in einem großen Schritt gehen, oder auch die Gesamtstrecke in einzelnen Etappen beschreiten. Dabei bieten sich die folgenden Strategien an, die Sie mithilfe der Azure-Cloud umsetzen können. Grob unterscheiden lassen sich die unterschiedlichen strategischen Ansätze wie folgt.

REHOST, REFACTOR UND REARCHITECT ...

... beschreiben die Modernisierung bzw. Migration einer Lösung für bzw. in die Cloud.

REBUILD UND REPLACE ...

... beschreiben die Transformation der Lösung zu einer Cloud-Native-Anwendung.



„Rehost“ auch bekannt als „Lift & Shift“, ist ein gängiger Ansatz für die Migration von Anwendungen in die Cloud. Sie verlagern Ihre Anwendung mit minimalen Anpassungen mithilfe von Infrastructure-as-a-Service (IaaS) in die Cloud.

IaaS

Bei „Refactor“ geht es darum, Anwendungen so anzupassen, dass diese mithilfe von Platform-as-a-Service-Diensten (PaaS) zusammenarbeiten können. Damit reduzieren Sie Komplexität und Kosten und erhöhen die Geschwindigkeit bei der Auslieferung neuer Funktionen.

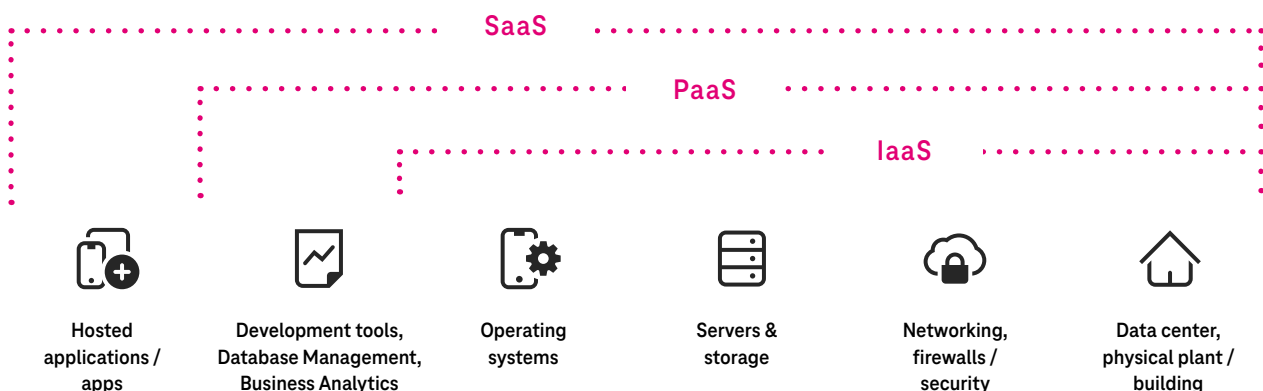
PaaS

„Rearchitect“ ermöglicht es Ihnen, Ihre Anwendung in mehrere funktionale Komponenten zu unterteilen, die individuell angepasst und weiterentwickelt werden können. Hierbei können Architekturentscheidungen mithilfe von PaaS-Diensten neu betrachtet und definiert werden. Sie erhalten dadurch eine bessere Flexibilität, eine erhöhte Skalierbarkeit sowie eine bessere Kontrolle über Ihre Anwendung und deren Architektur.

Eine „Rebuild“-Migrationsstrategie bedeutet im Gegensatz zum Rearchitect-Ansatz, bei dem Sie nur Teile der Anwendung modifizieren, dass Sie Ihre Anwendung auf einer PaaS-Infrastruktur unter Berücksichtigung von Cloud-Native-Methoden komplett neu entwickeln. So bieten sich Ihnen vollumfängliche Möglichkeiten und Sie können die Cloud-Dienste bestmöglich nutzen.

Bei einer „Replace“-Migration wird eine bestehende Anwendung vollständig durch Software-as-a-Service-Dienste (SaaS) ersetzt. Sofern die Ihnen zur Verfügung stehenden SaaS-Anwendungen alle erforderlichen Funktionalitäten bieten, können Sie mithilfe dieser Strategie künftig Kosten bei der Weiterentwicklung und dem Betrieb sparen.

SaaS



6. MEISTERN SIE DIE HERAUSFORDERUNGEN VON CLOUD-NATIVE – ES LOHNT SICH

Markus:

Der Schritt zu Cloud-Native-Produkten und einem Cloud-Native-Geschäftsmodell bringt neben einem Technologiewechsel auch einen Kulturwechsel im Unternehmen mit sich, der von „oben“ gelebt werden muss. Dieser Tatsache sollten Sie sich als Unternehmer, CEO oder Entscheider bewusst sein. So ist es als CEO eines Softwareherstellers heutzutage wichtig, nahe an der Technologie zu sein. Das bedeutet nicht, dass Sie als CEO oder Entscheider, alles bis ins letzte Detail verstehen oder gar selbst Software entwickeln können müssen. Es ist allerdings von großem Vorteil, sich mit den eigenen Cloud-Solution- und Software-Architekten unterhalten zu können. Nur so können Sie einschätzen, um was es geht und welcher der richtige Weg für Ihr Unternehmen und Ihr Produkt ist. Sollte das nicht der Fall sein, sollten Sie sich beratende Unterstützung hinzuziehen – beispielsweise aus dem eigenen Entwicklungsteam.

Die Erfahrung zeigt, dass sich die Softwareentwicklungs-Teams meist selbst bereits mit den Themen der Cloud-Native-Entwicklung beschäftigen und oftmals eine recht konkrete Meinung oder eigene Ideen im Kontext Ihres Produktes haben. In der Regel hängen Software-Entwickler mit Herzblut an „ihrem“ Produkt und haben konstruktive Vorschläge für dessen Weiterentwicklung oder Neuentwicklung, die Sie sich unbedingt anhören sollten. Die generelle Vorgabe, in Richtung Cloud-Native zu marschieren, muss aber aus dem Management kommen und von diesem komplett unterstützt und gefördert werden. Denn hinter der Cloud-Native-Entwicklung steckt nicht nur der Einsatz neuer Techniken, sondern auch ein Paradigmenwechsel, der das gesamte Unternehmen betrifft.

Die Auswirkungen von DevOps auf Unternehmen *

Qualitativ hochwertigere Leistungen

Schnellere Time to Market / Lieferzeit

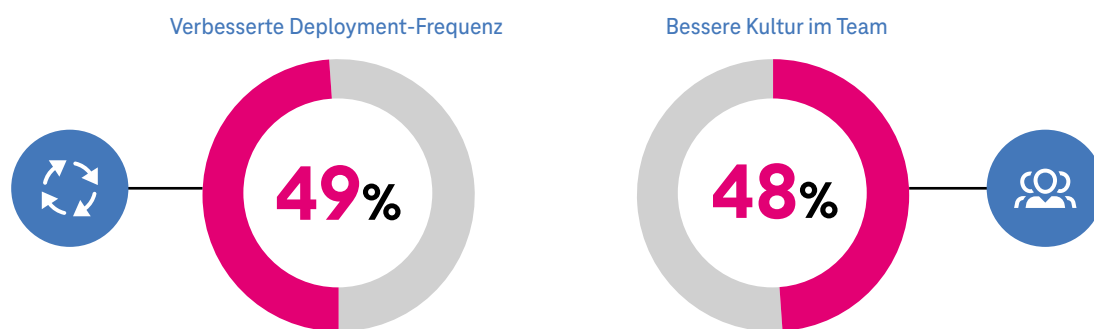


Quelle: * Atlassian & CITE Research: 2020 DevOps Trends Survey

Nico:

Hinter Cloud-Native steht nicht nur die Technologie, sondern auch die Implementierung verschiedener Methodiken sowie die Umstellung der täglichen Arbeitsweise. Der wahrscheinlich wichtigste Baustein ist DevOps. Dabei handelt es sich um eine Denk- und Vorgehensweise, die Development und Operations in einem Team vereint, um so Mehrwerte zu generieren. Häufig wird DevOps aber fehlverstanden und mit der simplen Einführung einer Toolchain gleichgesetzt. Dabei geht es hier jedoch viel mehr um einen Paradigmenwechsel, ohne den sich Cloud-Native-Anwendungen nicht erfolgreich umsetzen lassen. Wie bereits dargelegt wurde, ist auch die Automatisierung ein weiterer wichtiger Bestandteil. Es muss sichergestellt werden, dass alle wiederkehrenden Aufgaben automatisiert sind, um eine gleichbleibende Qualität und eine schnelle Auslieferung neuer Funktionen zu gewährleisten.

Die Auswirkungen von DevOps auf Unternehmen *



Eine weitere wichtige Komponente von Cloud-Native ist die Containerisierung. Container helfen, Anwendungen und ihre Anforderungen in Paketen zusammenzufassen und von der zugrundeliegenden Infrastruktur der Anwendung zu abstrahieren. Auf diese Weise machen Sie Ihre Anwendung flexibel und skalierbar und legen die Grundlage dafür, sie mit gleichbleibender Qualität zu betreiben. Dafür steht Ihnen eine Reihe verschiedener Plattformen zur Verfügung. Das wohl bekannteste Beispiel dürfte die Open-Source-Plattform Kubernetes sein. Entscheiden Sie sich nicht aufgrund seiner Bekanntheit für Kubernetes, sondern prüfen Sie im Vorfeld genau die Anforderungen Ihres Anwendungsfalls und entscheiden Sie sich dann aktiv auf dieser Grundlage für eine passende Plattform.

7. CLOUD-NATIVE UND AZURE

Markus:

Für die Entwicklung von cloud-nativen Anwendungen bietet Microsoft einen sehr durchgängigen und ausgereiften Baukasten. Mit der Azure-Cloud steht, unserer Ansicht nach und aus dem Blickwinkel eines Softwareherstellers, mitunter die umfassendste Public-Cloud auf dem Markt zur Verfügung. Sämtliche Cloud-Native-Technologien wie Container-Orchestrator, Datenbank-Dienste und Identity Provider (Azure AD und Azure AD B2C) stehen ebenfalls in sehr ausgereifter Form zur Verfügung. Es werden alle gängigen Open Source-Tools und Frameworks für die Entwicklung unterstützt und es gibt eine tiefe Integration in Application-Lifecycle-Management-Plattformen (ALM), wie Azure DevOps und GitHub für die Abbildung von CI/CD-Prozessen.

„Neben der Azure-Cloud als Betriebsplattform sind auch Integrationen in die Microsoft 365-Produktivitätsplattform und deren Marktplätze einfach möglich, wodurch die Zielgruppe für Cloud-Native-Anwendungen noch einmal erweitert wird.“

Auch die Entwicklungsumgebungen Visual Studio und Visual Studio Code weisen eine tiefe Integration auf. Ein sehr lebendiges Ökosystem und Communitys, welche eine Vielzahl an Code-Beispielen z. B. auf GitHub bereitstellen oder auch vordefinierte Lösungen über den Azure Marketplace publizieren, runden das Angebot ab. Gerade bei der Entwicklung von B2B-Lösungen ist Microsoft mit seinen Technologieplattformen ein sehr starker Partner. Neben der Azure-Cloud als Betriebsplattform sind auch Integrationen in die Microsoft 365-Produktivitätsplattform und deren Marktplätze einfach möglich, wodurch die Zielgruppe für Cloud-Native-Anwendungen noch einmal erweitert wird.

Als für die Cloud-Native-Entwicklung sehr wichtigen Azure-Service ist definitiv Azure Active Directory (AAD) hervorzuheben. Hier werden alle notwendigen Identitäten zentral abgelegt und verwaltet, um die Authentifizierung und Autorisierung durchgängiger Business-Szenarien und -Prozesse mit verschiedenen Anwendungen integriert abbilden zu können. Beispielsweise wird so nur eine Identität für die Authentifizierung und Autorisierung von Microsoft 365, Dynamics CRM und Ihrem eigenen branchenspezifischen Cloud-Produkt benötigt. Das stellt einen der zentralen Mehrwerte der Nutzung von Azure und dessen Ökosystem für die Entwicklung von Cloud-Native-Anwendungen dar.

Nico:

Für die Microservices Ihrer Cloud-Native-Anwendungen stellt Azure die folgenden Dienste bereit:

- AZURE KUBERNETES SERVICE

Ein hochverfügbarer, sicherer und vollständig verwalteter Kubernetes-Dienst.

- AZURE CONTAINER INSTANCES

Ein Dienst der das Betreiben von Containern serverless für Batch und event-basierte Workloads ermöglicht.

- AZURE FUNCTIONS

Beschleunigen und Vereinfachen der Anwendungsentwicklung mit serverless und event-basierten Funktionen.

- AZURE RED HAT OPENSIFT

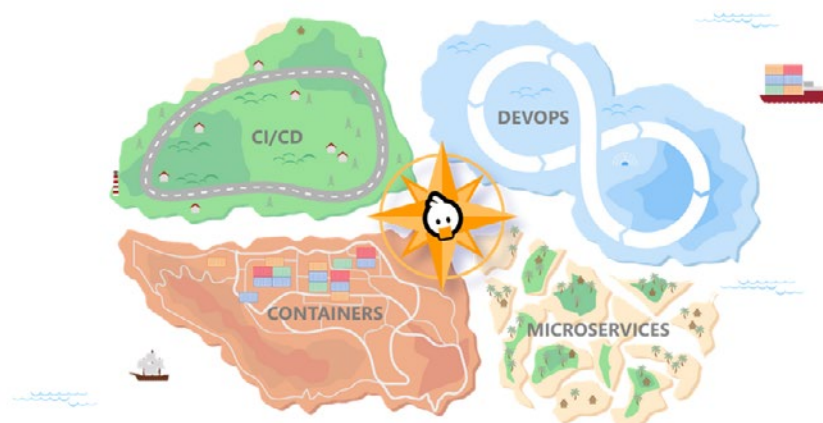
Eine vollständig verwaltete OpenShift-Umgebung, die gemeinsam von Red Hat und Microsoft betrieben wird.

Azure bietet mit der **Azure Container Registry** einen verwalteten Dienst zum Organisieren und Erstellen eigener Container Images.

Continuous Integration / Continuous Delivery können über **Azure DevOps Pipelines** oder GitHub Actions umgesetzt werden.

Identitäts- und Zugriffsverwaltung kann über ein vollständig verwaltetes **Azure Active Directory** (AAD und AAD B2C) implementiert werden.

Azure Storage ermöglicht die Ablage und das Management von Dateien über CLI und REST-API.



Zudem bietet die Azure Cloud eine Vielzahl an verwalteten Datenbanken, zu denen die folgenden zählen:

• **AZURE COSMOSDB**

Ein vollständig verwalteter NoSQL-Datenbankdienst für die moderne App-Entwicklung. Sie ist schnell, flexibel nutzbar und skalierbar

• **AZURE SQL DATABASE**

Eine verwaltete und intelligente SQL-Datenbank, die mit Ihren Anforderungen skaliert

• **AZURE DATABASES FÜR MARIADB, POSTGRESQL & MYSQL**

Verwaltete Open Source-Datenbanken für jeden Use-Case

• **AZURE CACHE FOR REDIS**

Eine vollständig verwaltete Version des beliebten Open-Source Redis Servers

Mit **Azure Monitor** erhalten Sie außerdem eine verwaltete Beobachtungsumgebung, mit der Sie Ihre Logs und Metriken organisieren. Mithilfe von Application Insights erkennen und diagnostizieren Sie Probleme über Anwendungen und Abhängigkeiten hinweg. Verteilte Microservices integrieren Sie mithilfe von Event Grid, Service Bus und API Management ineinander.

Microsoft stellt auch hilfreiche Developer-Tools und Open-Source-Projekte für die Cloud-Native-Community bereit. Hier finden Sie einige ausgewählte Beispiele:

• **DAPR**

Eine ereignisgesteuerte, portable Laufzeitumgebung für den Aufbau von Microservices in der Cloud und auf Edge-Geräten

• **KEDA**

Eine ereignisgesteuerte Autoskalierung für Kubernetes-Workloads

• **GITHUB CODESPACES**

Eine Online-Entwicklungsumgebung mit der Sie vollständig in der Cloud entwickeln können

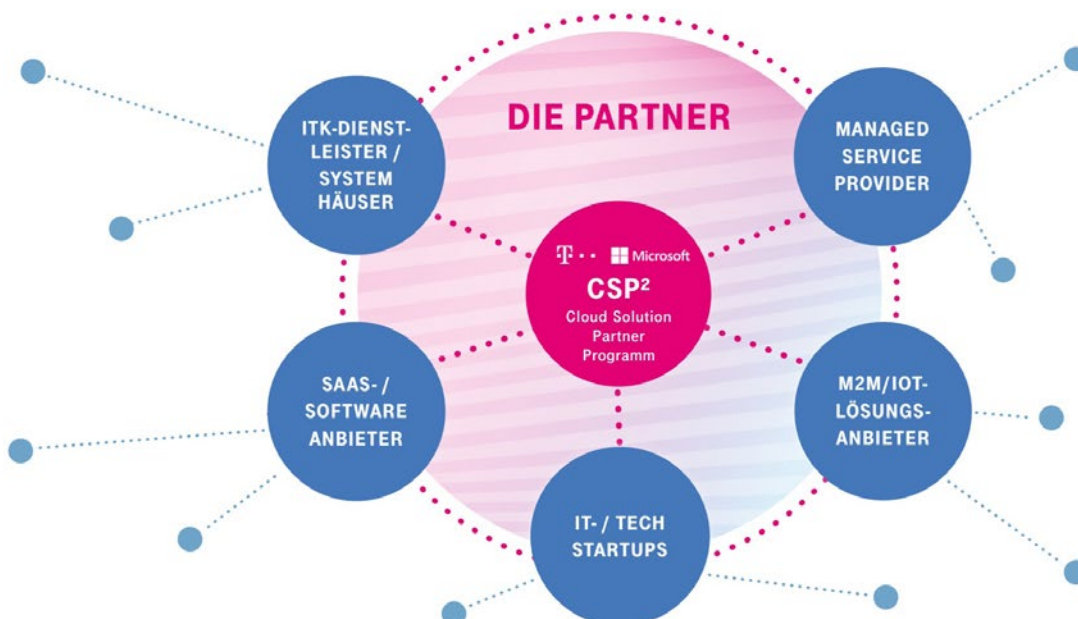
8. IM PARTNERNETZWERK GEMEINSAM ZUM ERFOLG

Markus und Nico:

Um ganzheitliche Cloud-Native-Anwendungen entwickeln zu können und das volle Potential aus modernen Cloud-Plattformen schöpfen zu können, ist das Zusammenarbeiten in Partnernetzwerken ein wichtiger Baustein für den Erfolg.

Dies hat uns die Praxis aus diversen Cloud-Projekten gezeigt. Die technologische Bandbreite der Cloud-Plattformen ist inzwischen so groß, dass ein Unternehmen, Dienstleister oder Partner diese kaum noch alleine in der notwendigen Know-How-Tiefe beherrschen kann. Neben den softwareentwicklungsorientierten Cloud-Native-Disziplinen wie Microservices-Architekturen, Containerisierung, DevOps und CI/CD ist für die Umsetzung moderner Cloud-Anwendungen oftmals auch Big Data-, KI-, IoT-, Security-, Betriebs- und auch Prozess-Spezialwissen erforderlich.

Daraus ergibt sich ein Denken und Arbeiten in Partnernetzwerken als logische Herangehensweise, um erfolgreich und ganzheitlich Cloud-Native-Projekte umzusetzen. Hier bietet z. B. die Telekom mit Ihrem Cloud Solution Partnerkanal eine Plattform, um Partner zusammenzubringen, mit dem Ziel gemeinsam in Ihrem Marktsegment, Ihrer Branche und für Ihre Endkunden Cloud-Lösungen zu entwickeln und zu betreiben.



9. WIE WHITE DUCK IHNEN HILFT

Markus und Nico:

Wir unterstützen Sie als Softwarehersteller dabei, Ihre Cloud-Native-Reise schneller und effektiver zu beschreiten. Nutzen Sie mit uns als Partner die Vorteile von Cloud-Native und reagieren Sie flexibel auf die sich stetig ändernden Anforderungen Ihrer Kunden – für ein resilientes und zukunftsfähiges Unternehmen.

Wir sind ein auf Software- und Cloud-Native-Engineering spezialisiertes Technologiehaus. Mithilfe unserer Cloud-Native-Journey unterstützen wir Sie in den verschiedenen Situationen und Stadien, die Sie auf Ihrer Reise durchlaufen werden. Unter unseren Kunden sind sämtlich Branchen vertreten – vom kleinen Start-up mit einer innovativen, cloud-basierten Geschäftsidee, über Softwarehersteller mit komplexen Softwareprodukten, bis hin zu Großunternehmen mit eigener Digitalisierungsagenda, die wir bei der Transformation ihrer Geschäftsprozesse in die Cloud unterstützen. Dafür steht Ihnen unser hochspezialisiertes Team von Cloud-Solution-Architects, Software-Engineers, DevOps-Engineers und Software-Testern bereit. Bei Bedarf erhalten Sie bei uns alles durchgängig und aus einer Hand: Wir konzeptionieren, entwickeln und betreiben Ihre Cloud-Lösungen. Gerne stehen wir Ihnen auch mit Beratung und Coaching in den Themenbereichen „Cloud“ und „Cloud-Native“ zur Seite.

Beim Rebuild oder Replace Ihrer schon bestehenden Lösung profitieren Sie von unserem Know-How. Gemeinsam mit Ihnen und Ihrem Entwicklungsteam analysieren wir zunächst Ihre bestehenden Anwendungen und Anforderungen. Im Anschluss definieren wir eine passgenaue Cloud-Native-Architektur für Ihren Use-Case. Dafür greifen wir auf die PaaS- und SaaS-Dienste der Azure-Cloud als Baukasten zurück. Während der Entwicklung und der Betriebsvorbereitung stehen Ihrem Team unsere Software- und DevOps-Engineers beratend und unterstützend zur Verfügung.

Auch bei der Entwicklung eines neuen Produktes und / oder Business-Cases stehen wir Ihnen gerne zur Seite. Dabei beginnen wir mit einem Minimum Viable Product (MVP). Im ersten Schritt definieren und entwickeln wir gemeinsam mit Ihnen die minimal funktionsfähige Version Ihres Produktes, mit der Sie bei geringem Aufwand den Kunden-, Markt- oder Funktionsbedarf decken und das Sie dann im Anschluss weiter ausbauen können.

Solution
Assessments
(Bewerten)



Proof of
Concept
(Praxiskonzept)



Implementation
(Umsetzen)



Unser Angebot für Softwarehersteller:

Sie wollen Ihr Unternehmen mithilfe von Cloud-Native für die Zukunft der Software-entwicklung vorbereiten? Melden Sie sich gerne bei uns! Wir unterstützen Sie persönlich bei Ihren individuellen Herausforderungen.

CLOUD NATIVE JOURNEY

whiteduck.de/cloud-native-journey

AZURE WORKSHOP

whiteduck.de/workshops

AZURE FÜR ISVS

whiteduck.de/microsoft-azure-kaufen

Stichwortverzeichnis

A

Azure	03
Azure Active Directory (AAD)	13
Azure API Management	15
Azure Cache for Redis	15
Azure Container Instance	14
Azure Container Registry	14
Azure CosmosDB	15
Azure DevOps Pipelines	14
Azure Event Grid	15
Azure Functions	14
Azure Marketplace	13
Azure Monitor	15
Azure Red Hat OpenShift	14
Azure Service Bus	15
Azure SQL Database	15
Azure Storage	14

C

Cloud-Native	03
Containerisierung	04
Continuous Delivery	05
Continuous Integration	05

D

Dapr	15
------	----

G

GitHub Actions	14
GitHub Codespace	15

I

Infrastructure as a Service (IaaS)	10
------------------------------------	----

K

Keda	15
Kubernetes	05
Kubernetes Service	14

M

Minimum Viable Product (MVP)	17
------------------------------	----

O

Orchestrierung	05
----------------	----

P

Platform as a Service (PaaS)	10
------------------------------	----

R

Rearchitect	09
Rebuild	09
Refactor	09
Rehost	09
Replace	09

S

Software as a Service (SaaS)	06
------------------------------	----

V

Visual Studio / Visual Studio Code	13
------------------------------------	----