



Erfolgreich in die digitale Zukunft mit Cloud Native

Die heute getätigten Investitionen in Cloud Computing bzw. Cloud Native Software legen den Grundstein für die digitale Transformation, die durch die Corona-Pandemie nochmals beschleunigt wurde.

Der Imperativ der Stunde ist das „Cloud Native“ genannte Design-Prinzip, welches den konsequenten Einsatz von modernen, dynamischen Public Cloud-Infrastrukturen sowie Container- bzw. Serverless Technologien und auf Standards setzende Microservices empfiehlt. Damit geht auch eine agile interne und somit abteilungsübergreifende als auch externe Zusammenarbeit mit IT-Dienstleistern einher, die im besten Fall auf Augenhöhe stattfindet bzw. auf einem partnerschaftlichen Level fußt.



Der Imperativ der Stunde ist das „Cloud Native“ genannte Design-Prinzip

Die Vorteile liegen dabei auf der Hand: Mehr Geschwindigkeit bzw. Agilität, höhere Qualität und vor allem Fokus auf das eigene Kerngeschäft sowie die Produkte/Services, die Kunden konsumieren. Ferner profitieren verzahnte Dev(Sec)Ops-Teams von robusten, anpassungsfähigen und innovativen Technologieplattformen, die professionell betreut werden wollen und tendenziell ein SaaS-Liefermodell anstreben.

Unternehmen sehen sich dabei vielen Herausforderungen und Risiken ausgesetzt, die nicht zuletzt aus dem Spannungsverhältnis von Agilität und Sicherheit hervorgehen. Dabei ist vor allem die End-2-End-Agilisierung des SW-Entwicklungs- und Maintenance- bzw. Governance-Prozesses entscheidend. Diese Entscheidungen können große Auswirkungen auf den mittelfristigen Geschäftserfolg und die Kundenzufriedenheit haben.

Damit vor lauter Agilität die Sicherheit und Qualität nicht vernachlässigt werden, gilt es Dev- und Ops-Teams zu synchronisieren, individuelle Update- und somit Release-Zyklen zu finden und dabei SW und Apps in den Mittelpunkt der digitalen Transformation zu stellen, ohne deren Betrieb zu vernachlässigen. Durch fehlendes Fachpersonal wird oftmals ein IT-Dienstleister involviert, der seinerseits selbst vor der Herausforderung steht, versierte Fachkräfte für den sicheren Betrieb im Dev(Sec)Ops-Modus zu finden und zu halten.

IT-Führungskräfte sollten kein kleinteiliges Management betreiben, sondern Spielraum zur Eigenständigkeit des Teams für die Entwicklung und vor allem die kontinuierliche Weiterentwicklung von Produkten/Services schaffen. Die Entwicklungsarbeit (Dev) kann dabei unter entsprechender Führung und Kontrolle ausgelagert werden, der Betrieb der IT-Infrastruktur bzw. -Plattform für das Deployment ist dagegen in den meisten Fällen in der Regel besser in den Händen eines professionellen Managed Cloud Service Providers und Dev(Sec)Ops-Experten aufgehoben. Anwenderunternehmen können sich so voll und ganz auf die Produkt-User-Experience und alle geschäftsnahen Pflichten konzentrieren.

DER MARKT IM WANDEL

Unter den Cloud-Nutzern und -Verantwortlichen in Unternehmen mit mehr als 50 Beschäftigten setzten im Jahr 2022 in Deutschland 27 Prozent Cloud-native Lösungen ein. Der Vergleichswert des Vorjahres lag bei 22 Prozent. Weitere 19 Prozent planen, Cloud-native Lösungen in den nächsten zwölf Monaten nutzen zu wollen, während nur 8 Prozent den Einsatz derzeit nicht in Betracht ziehen.

Unternehmen nahezu aller Größen und Branchen sind dazu gezwungen, Innovationen zu fördern und neue Geschäftspraktiken – ähnlich agil wie Start-ups – zu adaptieren. Dabei kristallisiert sich ein gemeinsamer Nenner heraus: Customer Experience Management. Kundenbedürfnisse und deren Zufriedenheit sind somit wichtiger denn je für den Geschäftserfolg. Der harte Wettbewerb durch innovative Neugründungen zusammen mit steigenden Kundenerwartungen veranlasst daher jedes Unternehmen, seine Angebots- und Geschäftspraxis kontinuierlich zu hinterfragen. Dazu kann der Wandel hin zu einem datenzentrierten Unternehmensmodell gehören, bei dem Entscheidungen so weit wie möglich durch harte Fakten bzw. Informationen gerechtfertigt werden. Diese Entscheidungsgrundlage bezieht sich auf das gesamte Portfolio und somit insbesondere auf das IT Service Lifecycle Management.

Der sich verändernde Markt wird bestimmt durch schnelllebige und individuelle Kundenanforderungen. Um diesen zu entsprechen, erfordert es einen cleveren Einsatz von modernsten Cloud Native-Technologien, die ebenfalls einen Kulturwandel bedeuten und damit ein entsprechendes Change-Management einfordern. Daher stehen Anbieter von Dev- bzw. Ops-Leistungen unter dem Druck, kooperative Ansätze zu entwickeln,

Erfolgreich in die digitale Zukunft mit Cloud Native

die die besten Kompetenzen aus beiden Welten verbinden, um damit die Möglichkeit auch für kurze Sprints in enger Zusammenarbeit mit Kunden zu liefern, ohne Verfügbarkeiten zu reduzieren. Hier etablieren sich „Shared Risk“-Modelle bis hin zum Site Reliability Engineering (SRE), bei dem sich Systemadministratoren zum Softwareingenieur weiterentwickeln müssen. Der steigende Automationsgrad wird hier ebenso zur Voraussetzung für den Erfolg.

Public Cloud Services und deren Standardisierung und vor allem die Möglichkeiten zur Skalierung unter Einbezug von Massendaten können entscheidende Mehrwerte in einem Großteil der Unternehmensbereiche liefern. Dank beinahe flächendeckender Internet-Konnektivität und -Mobilität sind inzwischen viele mögliche Use Cases zum Greifen nahe oder schon Realität geworden.

In diesem Kontext sind Communities wie die Cloud Native Computing Foundation (CNCF) als Rückgrat heutiger Innovationen erwähnenswert. Selbst die sehr verbreitete Container-Orchestrierung Kubernetes (K8s) von Google basiert auf Open Source und profitiert kontinuierlich von der schnellen Weiterentwicklung durch die Schwarm-Intelligenz. Zudem tragen immer professionellere Verwaltungsmöglichkeiten für Container dazu bei, die klassische Virtualisierung mit Hypervisoren zu verdrängen. In Anbetracht dessen ist nicht verwunderlich, dass Unternehmen mehr denn je quelloffene Lösungen nutzen und selbst andere Hyperscaler auf diesen De-facto-Standard setzen.

Cloud Native-Technologien und Philosophien – Container und Kubernetes

Zusätzlich zu veränderten Betriebsmodellen bedingt der ständige Marktdruck eine Veränderung der Liefermodelle und die Adaption von Technologien. Die Möglichkeit, monolithische Architekturen bzw. Applikationen aufzubrechen in kleine, leichter zu orchestrierende Microservices, macht die Nutzung von Public Cloud Services respektive Cloud Native- und somit auch Container-Technologien zum Thema der Stunde.

In den letzten zehn Jahren haben sich neue Muster und Technologien für die Entwicklung, den Einsatz und den Betrieb moderner Anwendungen herausgebildet, die sich die in Cloud-Infrastrukturen verfügbaren Möglichkeiten zunutze machen. Dieser Cloud Native-Ansatz konzentriert sich auf die Erstellung von Anwendungen, die hochgradig modular, anpassungsfähig, fehlertolerant und besser in der Lage sind, den Endbenutzern einen Mehrwert zu bieten. Ferner werden Applikationen intelligenter in Bezug auf das Handling von

Erfolgreich in die digitale Zukunft mit Cloud Native

Events zur Infrastruktur-Steuerung bzw. Skalierung, Plattform-Migration und cross-funktionaler Datenaustausch als auch Kompatibilität sind wichtige Eigenschaften, die mittels Standards sowie Open Source realisiert werden können.

Ein Container ist eine Form der Virtualisierung, in der die Anwendung von der Laufzeitumgebung sowie dem Betriebssystem getrennt wird. Das schlanke Design dient dazu, mehrere Anwendungen isoliert voneinander und dem zugehörigen Betriebssystem betreiben zu können. Ein Container Image wird mittels einer Laufzeitumgebung wie bspw. Docker zu einem leichtgewichtigen, eigenständigen, ausführbaren und für sich abgegrenzten Softwarepaket, das als Vehikel für Code und zugehörige Abhängigkeiten gesehen werden kann. Images lassen sich einfach und auf reproduzierbare Weise in verschiedene Umgebungen deployen bzw. migrieren. Somit wird die Ressourcenauslastung, die Fähigkeit zur Skalierung wie auch die gesamte Agilität des IT-Betriebs durch Container-Technologien im Gegensatz zur herkömmlichen Virtualisierung verbessert. Dies gilt jedoch ebenso für vorhandene Bugs, die sich schnell multiplizieren und eine gewisse Angriffsfläche für Hacks bieten.

Insbesondere Kubernetes ist zum De-facto-Standard der Container-Orchestrierung und damit moderner Microservice-basierter IT-Landschaften geworden. Es bietet Softwarefunktionen, die eine einfachere Verwaltung von Multi-Container-Anwendungen ermöglichen, einschließlich automatischer Skalierung, Verwaltung von Containerausfällen und Routing des Netzwerkverkehrs.

Container Services und Open Source helfen Unternehmen dabei, den „Lock-in“ einzelner IT-Anbieter bzw. -Plattformen zu reduzieren, Migrationen zu vereinfachen und Kosten zu sparen. Unternehmen nutzen Cloud-native Technologien in Kombination mit agilen Methoden wie Dev(Sec)Ops und kontinuierlichen Integrations- bzw. Liefermodellen (CI/CD – siehe Kapitel 3) und gewinnen dabei an Geschwindigkeit und Flexibilität bei der Auslieferung und Wartung von IT-Services, die unabhängig voneinander betrieben werden.

Serverless Computing ist dabei ein weiterer Cloud Trend, bei dem die Applikation und somit das Geschäft stärker in den Vordergrund rückt, so dass die Bereitstellung, Skalierung und das Management von Server-Ressourcen wie CPU, RAM, Storage und Netzwerk weitestgehend automatisiert vonstattengehen kann. Dieses Ausführungsmodell wird auch als Function as a Service (FaaS) bezeichnet, bei dem es sich um Laufzeitumgebungen handelt, die der Infrastruktur-Provider in Abhängigkeit der Last dynamisch zuweist und skaliert.

Das Besondere an Cloud Native ist die Herausforderung des harmonischen Zusammenspiels vieler Parteien im Kontext von DevOps. Entwickler (Developers=Dev) und Operations (Operations=Ops) arbeiten eng zusammen. In einem kontinuierlichen Austausch auf Basis eines gemeinsamen Ziels und Nutzenversprechens werden Applikationen oder Microservices über automatisierte Prozesse bereitgestellt (Continuous Delivery (CD)). Die Entwickler können so viel schneller als bisher in produktive Umgebungen „deployen“. Aufgrund immer kleinteiliger Releases in kurzen Abständen ist zwar eine Problemlösung aufgrund der Änderungen in Form kleinerer „Batches“ einfacher, es braucht jedoch eine gewisse Transparenz bzw. eine Anwendungs-Telemetrie sowie Disziplin, um automatisch Daten über die geschäftliche und technische Leistung der Lösung zu sammeln – nur so gelingt eine datenzentrierte Unternehmensteuerung, die alte Bauchentscheidungen ablösen kann. Um die kontinuierliche Lieferpipeline und das Konzept der Freigabe auf Anforderung zu unterstützen, muss das System für eine risikoarme komponenten- oder service-basierte Lieferung inklusive schlanker Freigabe-Prozesse sowie Recovery-Szenarien ausgelegt sein. Hier greift das Prinzip „fix forward or rollback“.

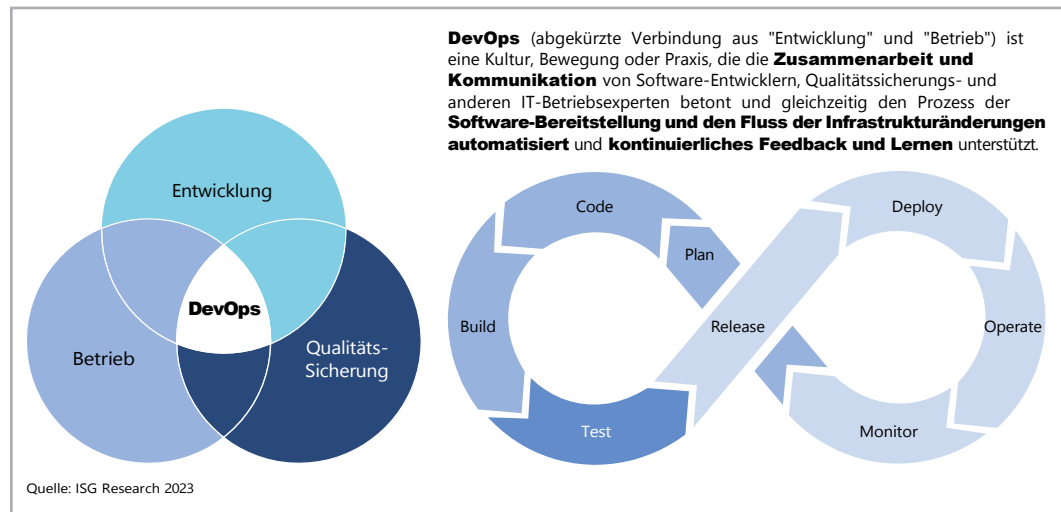


Figure 1: Was ist DevOps?

In der Konsequenz gewinnen vor allem Orchestrierungswerkzeuge wie Kubernetes (K8s) an Zugkraft, da verschiedene Dienste in einer Anwendung in separate Container verpackt und auf zahlreichen physischen oder virtuellen Maschinen eingesetzt und betrieben werden können. Zusätzlich wird die Migration von Plattform zu Plattform vereinfacht.

Kategorien von Anwendern im Kontext Cloud/Digital Transformation

Unternehmen über alle Größenklassen und Branchen stehen an unterschiedlichen Stellen hinsichtlich ihres Cloud-Reifegrads und müssen daher auch individuelle Ansätze zur Cloud-Adoption in Betracht ziehen. ISG klassifiziert Unternehmen im Kontext Cloud in vier Typen von Buying Centern oder auch Anwendertypen. Die Bandbreite ist groß und sicher nicht auf jedes Unternehmen zutreffend. Dennoch können sich die meisten Unternehmen mit einem der vier Typen bestehend aus Traditionalisten, Pragmatikern, Strategen bis hin zum Typ „Next Gen“ Innovatoren identifizieren.

Unternehmen sind selten gleich aufgestellt und haben stets individuelle Kenntnisstände von neuesten Technologien und Historien in Bezug auf die Erbringung von Leistungen gegenüber Kunden und dem Markt. Dies gilt auch für die Zusammenarbeit mit IT-Dienstleistern als Partner an ihrer Seite.

Anwendertypen bzw. Buying Center im Überblick

Traditionalisten	Pragmatiker	Strategen	Next Gen Innovatoren
<ul style="list-style-type: none"> haben die Relevanz der Cloud für Computing-Bedürfnisse nicht akzeptiert haben hauptsächlich IT-Umgebungen, die aus Mainframe- und Legacy-Anwendungen bestehen setzen auf Vorschriften und sind skeptisch gegenüber neuesten Trends sind inzwischen offen dafür, mehr über die Vorteile von Cloud Computing zu erfahren brauchen Unterstützung bei der Bewertung ihrer IT-Umgebung und der Formulierung einer Strategie sind risikoavers und konzentrieren sich auf Kosteneinsparungen durch ein FTE-basiertes Modell haben einen niedrigen Outsourcing- bzw. IT-Reifegrad (z.B. unzureichend qualifiziertes Personal oder Bandbreite) 	<ul style="list-style-type: none"> sind Outsourcer der zweiten oder dritten Generation suchen die Zusammenarbeit mit mehreren Dienstleistungsanbietern in einem Managed Services- und professionellen Modus bestehen auf die Einhaltung der Dienstleister Service Level Agreements (SLAs) oder Vereinbarungen auf Geschäftsebene bzw. fest definierte Termine verwalten kaum noch betriebliche Aspekte in Form eines Micro Managements, sondern überlassen es Dienstleistern, die ordnungsgemäße Überwachung und Messung der Produktivität sicherzustellen nutzen vorwiegend Festpreis- Vertragsmodelle mit sehr genau definiertem Leistungsumfang 	<ul style="list-style-type: none"> haben eine strategische und nachhaltige Sicht auf das gesamte IT-Ökosystem und Pläne, die derzeitige IT-Einrichtung in eine Cloud-Umgebung umzuwandeln erzwingen keine Anpassung der Legacy-Infrastruktur und Anwendungen an die Cloud sind bereit, Risiken einzugehen bzw. auch Workloads in unterschiedlichen Geschwindigkeiten und Modellen zu betreiben, um strategische Ziele zu erreichen haben als Ziel, schnellere, stärker integrierte und benutzerfreundliche Anwendungen durch Plattformen zu realisieren versuchen im Gegensatz zu Käufern von Managed Services, die nach Verbesserungen bei Prozessen und Systemen suchen, nicht die IT-Umgebung selbst zu verändern sondern in einer Partnerschaft mit Spezialisten 	<ul style="list-style-type: none"> sind Early Adopter der Cloud und verfolgen einen "Cloud First"-Ansatz legen den Schwerpunkt auf die Verwendung von "Born in the Cloud"-Anwendungen zur Nutzung der cloud-nativen Fähigkeiten, die zur Entwicklung von Anwendungen in Containern verwendet, als Mikrodienste bereitgestellt und auf einer elastischen Infrastruktur durch agile DevOps-Prozesse und kontinuierliche Bereitstellungs-Workflows verwaltet werden werden nicht durch die Anforderungen und Barrieren von Legacy-Operationen und Anwendungen belastet betrachten die IT als Change Agent und als Schlüsselkomponente von Umsatz- und Gewinnwachstum sind in hohem Maße kundenorientiert und versuchen, sich durch die Nutzung neuer Technologien einen Wettbewerbsvorteil zu verschaffen
<p>Steigende Akzeptanz für Cloud Native-Technologien und -Philosophien </p>			

Quelle: ISG Research 2023

Erfolgreich in die digitale Zukunft mit Cloud Native

HERAUSFORDERUNGEN UND LÖSUNGEN

Eine erfolgreiche Wettbewerbsstrategie war schon immer, neue Arbeitsweisen und Technologien des Marktes zu adaptieren. Diejenigen, die sich darauf einlassen und mitziehen, können sich dadurch entscheidende Wettbewerbsvorteile sichern. Doch die exponentiell laufenden Veränderungen erfordern eine schnelle Reaktionsfähigkeit auf Seiten aller Marktteilnehmer.



Exponentielle Veränderungen erfordern eine schnelle Reaktionsfähigkeit und agile Dev(Sec)Ops-Teams

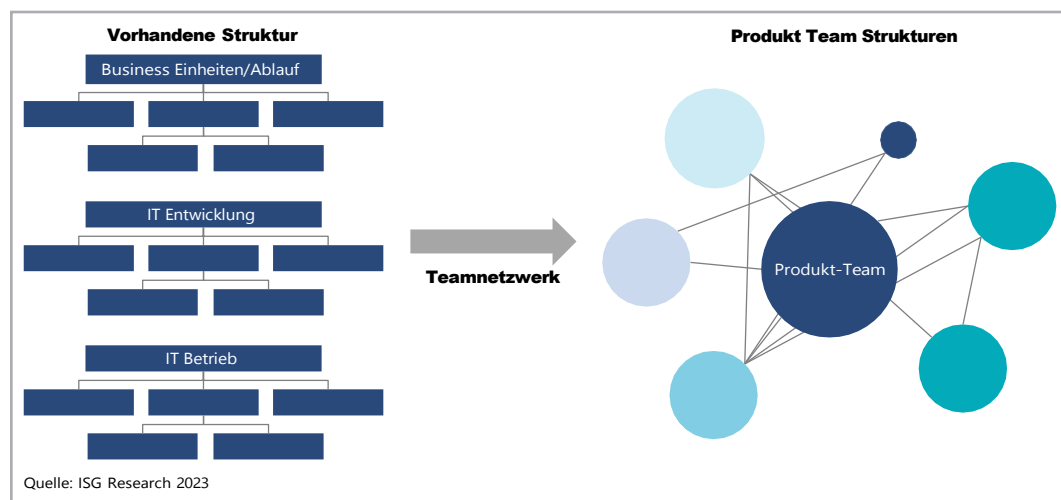


Figure 2: Eine Möglichkeit, die Umstellung auf Agilität zu unterstützen, ist die Ausrichtung von Ressourcen auf bestimmte IT-Produkte - das resultierende Team wird Produkt-Team (oder Pod) genannt

Für die Umsetzung von Schnelligkeit und Anpassungsfähigkeit ist es notwendig, eng verwobene Teams zu etablieren, die im sog. Dev(Sec)Ops Mode agil arbeiten. Dies geht weit über die herkömmliche Philosophie der notwendigen Verzahnung von IT und Business hinaus und bedingt eine permanente Zusammenarbeit und neue Rollen.

Die IT muss verschiedene Stakeholder, Kunden und Partner möglichst in Echtzeit sicher und strukturiert vernetzen, um ein digital-transformiertes Produkt- bzw. Service Lifecycle Management zu ermöglichen. Die Cloud Native- und Digital-Experten sind jedoch Mangelware, weshalb Anwenderunternehmen meist auf professionelle Unterstützung zurückgreifen müssen und dabei auch neue Vertragsmodelle wie bspw. das Shared-Responsibility-Modell für sich entdecken.

Konsequenz der neuen Taktgeber Public Cloud bzw. Cloud Native

Unternehmen müssen sich organisatorisch umstellen und Methoden wie beispielsweise SCRUM und somit kleinteilige Sprint-Arbeitszyklen aneignen, um Kundenanforderungen gerecht zu werden und neue IT-Liefer- und Betriebsmodelle für sich auszuschöpfen. Als absehbare Konsequenz werden Unternehmen immer mehr Gewicht auf Cloud-Experten und -Entwickler bzw. Software-Architekten legen müssen, um IT bzw. Cloud als essenziellen Produktionsfaktor im Griff zu haben. Es gilt mehr denn je, zum Software- und Plattform-Experten zu werden und auf Standards bzw. SaaS sowie auf die dadurch entstehenden Möglichkeiten digitaler Plattformen und Ökosysteme zu setzen.

Pipeline & Container Management

Moderne Applikationen bzw. Microservices werden tendenziell in Containern oder gar Serverless betrieben. Unternehmen brauchen daher eine entsprechende Orchestration Engine bzw. Lösungen zur Verwaltung von Containerarchitekturen (für Anwendungen, Daten, Sicherheitsfunktionen, Infrastruktur) für die Entwicklung und den Betrieb von Applikationen. Dies stellt die Produkt- bzw. Serviceverfügbarkeit sowie die Plattformflexibilität sicher und erzielt letztlich Kosteneinsparungen. Die Architektur eines Betriebs- und Liefermodells ist dabei eine große Herausforderung.

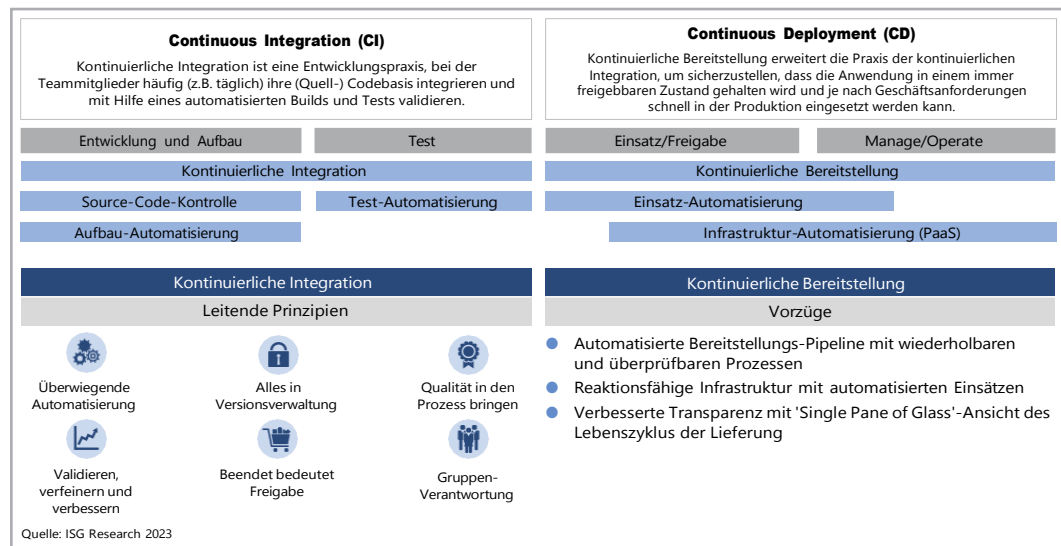


Figure 3: Überblick CI/CD

Erfolgreich in die digitale Zukunft mit Cloud Native

Eine Alternative ist die Übernahme des Architektur-Designs und Betriebs durch spezialisierte IT-Dienstleister. Zusätzlich zu einem erforderlichen Strategie-Support werden spezielle Dienstleistungen für containerbasierte Anwendungsoperationen wie Monitoring, Protokollierung und Patching, aber auch prädiktive Analysen für Leistungsgarantien und Sicherheitsverbesserungen angeboten.

Aus Anwendersicht sind folgende Kriterien essenziell für einen performanten und sicheren Container-Betrieb von bestehenden oder zu entwickelnden Applikationen:

- Kompetenzen für das Netzwerkleistungsmanagement (software-basiert), um Flaschenhälse zu eliminieren bzw. zu skalieren und einen hohen Datendurchsatz zu garantieren.
- Support für Hybrid Clouds, d. h. für Kombinationen aus privat gehosteten Containerservices und Public-Cloud-Angeboten von AWS, Microsoft Azure, Google Cloud Platform (GCP), IBM Cloud o.a.
- Bereitstellung von Monitoring, Protokollierung, Patching und prädiktiven Analysen für Leistungsgarantien und Sicherheitsverbesserungen über den gesamten Container-Lebenszyklus hinweg
- Etablierte oder sich entwickelnde Partnerschaften mit mindestens einer Container-Orchestrierung und Cluster Management Tools/Services wie Kubernetes, Docker Swarm, Rancher, Nomad oder OpenShift
- Automatisierte und je nach Applikationsprofil ausgewählte Architektur-Templates für Infrastruktur- und Plattform-Services, um darauf aufbauend eine kontinuierliche Microservice-Integration und Lieferung zu ermöglichen. Man spricht von Continuous Integration & Delivery (CI/CD).

Obwohl Dev(Sec)Ops in den letzten Jahren zusammen mit einer verstärkten Adaption an Attraktivität gewonnen hat, kann es ohne professionelles Tooling und robuste Automatisierung nicht bestehen. Der Marktschub ist das Ergebnis der wachsenden Nachfrage nach geschäftlicher Agilität. Bis zu einem gewissen Ausmaß kann die agile Bereitstellung Innovationen und digitale Geschäfte beschleunigen, doch die manuelle Übernahme des Anwendungscodes durch Entwicklung, Testen, Produktionskonfiguration, Codebereitstellung und Datenaustausch wird normalerweise zum Flaschenhals. Für Kunden und Dienstleistungsanbieter ist die Integrationsfähigkeit entscheidend, für die Anbieter DevOps-Plattformen entwickelt haben, um eine Vielzahl an Tools zu verknüpfen und Deployments zu automatisieren.

DevOps zur Testautomatisierung: Obwohl sein Hauptvorteil in der automatisierten Code-Einsetzung in die Produktion besteht, hat sich die Testautomatisierung, einschließlich Sicherheits- und Konformitätstests, zu einer der wichtigsten Triebkräfte für die Einführung von DevOps entwickelt. Der Zweck der Testautomatisierung mit standardisierten Container Images besteht darin, die Funktionalitäten sicherzustellen, die in Images eingefügt werden, und nicht die zugrundeliegende Infrastruktur – diese ist austauschbar.

Größeres Sicherheitsbedürfnis: Unternehmen und Anbieter stellen fest, dass die Implementierung von Sicherheit nicht im Nachhinein bedacht werden kann. Dev(Sec)Ops ersetzt DevOps zunehmend auf breiter Front. In den ersten Phasen der Dev(Sec)Ops-Implementierung werden daher Sicherheitsprinzipien als Standardfunktion integriert.

Heterogene Systeme sind eine Herausforderung, aber kein Hindernis: Ältere Systeme sind immer noch die größte Hürde für die Agilität von Unternehmen, da sie eine agile Geschäftsprozessentwicklung behindern und in der Regel ein wenig dynamisches und sehr stabiles IT-System und Prozessumfeld brauchen. Unternehmen suchen daher nach Wegen, um die Blockade ihrer älteren Systeme zu überwinden. Dies geschieht häufig indem sie Berater nutzen, die ihnen Mittel und Wege hin zur nötigen Enterprise Agility im Rahmen ihrer digitalen Transformation aufzeigen und die bestehende Landschaft einzuordnen wissen. Viele Dienstleister bieten auch dedizierte Agile-Dienste für die Modernisierung älterer Anwendungen an.

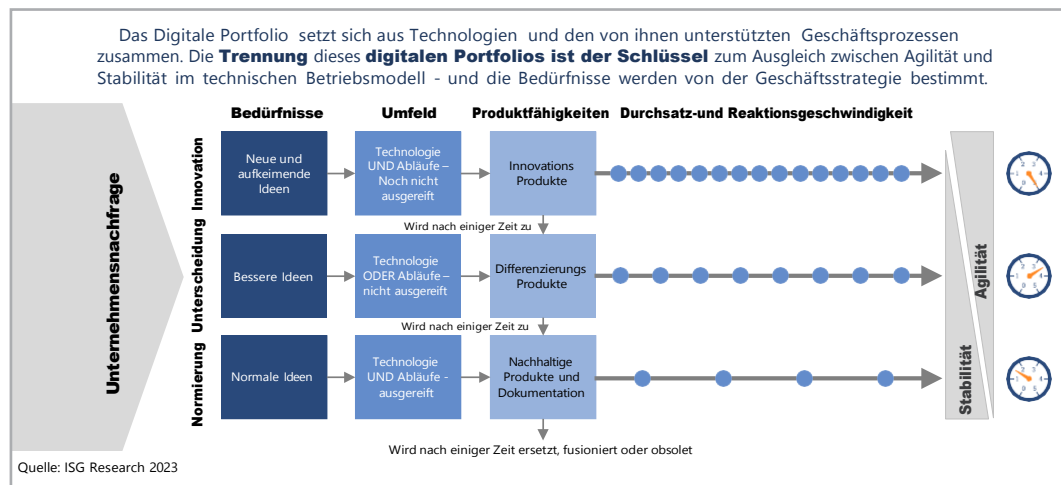


Figure 4 : Unternehmensbedürfnisse fließend anpassen

Dennoch: Die tiefgreifende Kenntnis der wichtigsten Altsysteme und deren Anforderungen hinsichtlich Stabilität und Agilität wird weiterhin ein entscheidender Aspekt für viele Branchen sein. Etablierte Unternehmen können eben nicht einfach den Schalter von alt auf neu umlegen, sondern müssen verschiedene Geschäftspraktiken und Geschwindigkeiten miteinander verbinden.

Neue Anforderungen an den modernen Applikationsbetrieb im Kontext kontinuierlicher Integration und Entwicklung (Continuous Integration & Delivery – CI/CD) überfordern viele Anwender.

Die helfende Hand professioneller Service Provider kann hier notwendig sein, um bei der Strategie und dem Infrastruktur- bzw. Plattform-Betrieb oder der Bereitstellung der Deployment Pipelines zu unterstützen. Dies muss stets im Kontext der Notwendigkeit des Aufbaus eines kontinuierlichen IT-Lifecycle Managements betrachtet werden, um den Weg zu datengetriebenen Anwendungen und intelligenten, hoch automatisierten Services zu ebnet.

Alte vs. neue Welt der Applikationen in heterogenen Systemlandschaften

Inzwischen gibt es auch viele Möglichkeiten, Altsysteme in neuen Umgebungen zu betreiben oder Teile des Codes neu zu entwickeln. Die Einrichtung eines einzelnen Kubernetes-Clusters auf einer homogenen Infrastruktur ist mit den derzeitigen Lösungen auf dem Markt relativ einfach. Die Realität sieht jedoch so aus, dass Unternehmen unterschiedliche Infrastrukturen mit unterschiedlichen Server-, Speicher- und Netzwerkanbietern haben. Bei diesen Voraussetzungen hilft Kubernetes, die Bereitstellung der Infrastruktur zu automatisieren und so einzurichten, zu konfigurieren und aufzurüsten, dass alle Systeme konsistent funktionieren.

Die Herausforderung beginnt in vielen Fällen jedoch weitaus früher: bei der Erfassung der Applikationslandschaft mit allen Abhängigkeiten zum Geschäft. Das Portfolio muss sauber analysiert und bewertet werden. Die Strategien sind mannigfaltig und reichen von „remove“ bzw. „retain“, über „Lift & Shift“ bis hin zum „Replacement“ durch alternative SaaS- bzw. Cloud Native-Lösungen. Diesbezüglich hat sich im Markt eine Cloud First-Regel etabliert, um langwierige und kostspielige Applikations-Transformationen zu reduzieren oder gar zu unterbinden und neue Applikationen bzw. Services stets auf Basis von Cloud Native-Technologien zu entwickeln.

App Development Best Practices

Die Zwölf-Faktoren-Vorgehensweise zur Anwendungserstellung und -modernisierung ist eine Reihe von etablierten Best Practices (oder Architektur-Pattern) für die moderne Anwendungsentwicklung und konzentriert sich auf die Optimierung der Geschwindigkeit, Sicherheit und Skalierbarkeit als Treiber Cloud-nativer Architekturkonzepte.

Die entwickelte Anwendung sollte daher folgende Muster zur Optimierung der Cloud-Nutzung unterstützen:

- Architektur der Microservices
- Containerisierte Services
- Verteilte Verwaltung und Orchestrierung

App Dev Best Practices	
<p>1. Codebase Eine im Versionsmanagementsystem verwaltete Codebase, viele Deployments</p> <p>2. Abhängigkeiten Abhängigkeiten explizit deklarieren und isolieren</p> <p>3. Konfiguration Die Konfiguration in Umgebungsvariablen ablegen</p> <p>4. Unterstützende Dienste Unterstützende Dienste als angehängte Ressourcen behandeln</p> <p>5. Build, release, run Build- und Run-Phase strikt trennen</p> <p>6. Prozesse Die App als einen oder mehrere Prozesse ausführen</p>	<p>7. Bindung an Ports Dienste durch das Binden von Ports exportieren</p> <p>8. Nebenläufigkeit Mit dem Prozess-Modell skalieren</p> <p>9. Einweggebrauch Robuster mit schnellem Start und problemlosem Stopp</p> <p>10. Dev-Prod-Vergleichbarkeit Entwicklung, Staging und Produktion so ähnlich wie möglich halten</p> <p>11. Logs Logs als Strom von Ereignissen behandeln</p> <p>12. Admin-Prozesse Admin/Management-Aufgaben als einmalige Vorgänge behandeln</p>
<small>Quelle: https://12factor.net</small>	

Figure 5: App Development – 12 erfolgsentscheidende Faktoren

Migration und Rollout sind nicht alles

Die Optimierung von Rollouts, Skalierbarkeit, Migration und der Ressourcen-Einsatz sind große Herausforderungen, neben dem Infrastruktur- und Plattform-Management. Die Entwicklung, Bereitstellung und der Betrieb von Cloud-nativen Anwendungen erfordert mehr als nur Container-Orchestrierung. Beispielsweise müssen IT-Betriebsteams immer noch Firewalls, Load Balancer, DNS-Dienste und möglicherweise Datenbanken einrichten, um nur einige zu nennen. Sie müssen nach wie vor Infrastrukturoperationen wie die Wartung physischer Hosts, das Hinzufügen/Entfernen/Ersetzen von Festplatten bzw. physischer Hosts verwalten. Sie müssen immer noch Kapazitätsplanung betreiben und die Auslastung, Zuweisung, Rechenleistung, Speicherung und Vernetzung überwachen. Kubernetes hilft bei all diesen Aufgaben nur bedingt weiter. Es ist deshalb nicht überraschend, dass die technische Infrastruktur- und Plattform-Verwaltung immer häufiger in die Hände von professionellen Dienstleistern gelegt wird. Als Konsequenz haben Unternehmen mehr Zeit, sich auf die Entwicklung und die funktionelle Wartung von Applikationen zu konzentrieren.

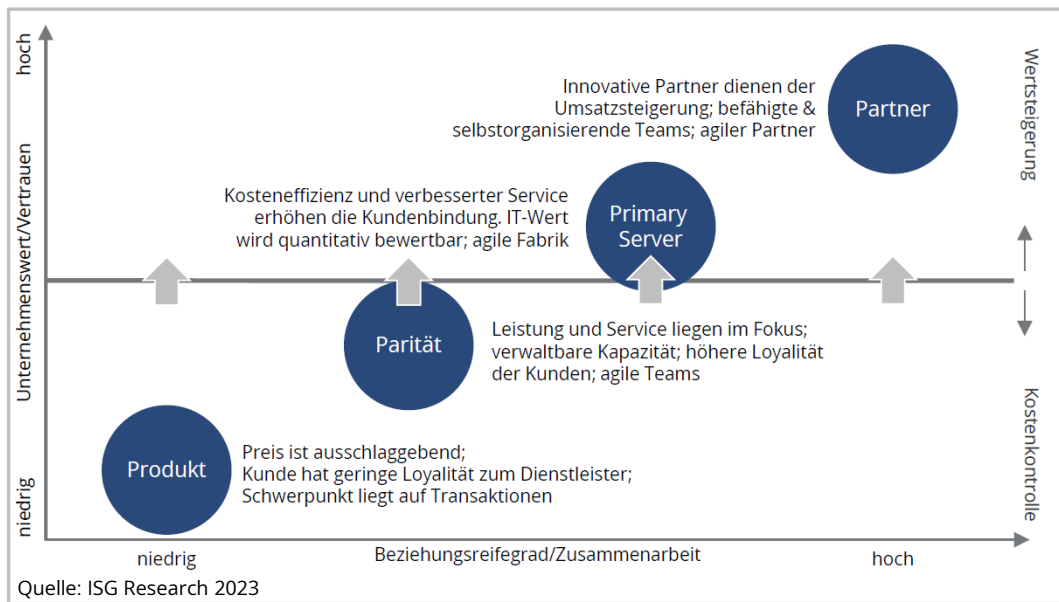


Figure 6: Was definiert die Cloud Native-Partnerschaft?

Shared Responsibility – IT-Anwender und -Anbieter im partnerschaftlichen Duett

Die Suche nach einem geeigneten IT-Dienstleister ist aus Anwendersicht nicht einfach. Kaum ein Unternehmen startet auf der grünen Wiese und kann die Altlasten komplett beiseiteschieben. Zudem gilt es in puncto Cloud Native einen agilen und wirklich fortschrittlichen Partner zu suchen, der sich bspw. mit dem Cloud Native Computing Foundation- (CNCF-) Ökosystem beschäftigt, um aus der Community neue Impulse aufzusaugen und sich mit Zertifizierungen wie der für Kubernetes rühmt – dies und viele weitere Punkte sind gute Indizien für die Performance. Damit geht allerdings eine veränderte Form der Zusammenarbeit einher, die auf neue Formen der Projekt-Vertragsmeilensteine setzt und beide Parteien (Kunde und IT-Dienstleister) als gleichwertige Partner betrachtet.

Neben dem gegenseitigen Vertrauen auf dem Weg zu einer solchen Partnerschaft, gilt es klare Rollen und Aufgaben zu definieren. Dabei ist es zum einen wichtig, das Zusammenspiel von IaaS-PaaS-SaaS aus Provider-Händen und On-Premise-IT-Management in Bezug auf Datenverantwortung, IAM sowie die Kontrolle auf Plattform- und Applikationsebene eindeutig zu regeln. Ferner gilt es einen gesunden Mix aus Fixed Price-Leistungen und Komponenten wie On-Demand-Bausteinen & Consumption-based-Dienstleistungen aufzusetzen. Dies erfordert Erfahrung und Vertrauen – auf beiden Seiten.

CASE STUDY: MOBILITÄT DER NÄCHSTEN GENERATION AUF DER GOOGLE CLOUD



Digitalisierung und Public Cloud-Nutzung am Kundenbeispiel

Welche Herausforderungen und Effekte sich durch die Einführung von Cloud Native-Technologien ergeben, zeigt die Case Study eines großstädtischen Verkehrsbetriebs aus dem Public Sector. In einem umfassenden Digitalisierungsprojekt sollte dem Endkunden ein zukunftssicherer Zugang zu urbaner Mobilität ermöglicht werden. Der Ansatz: Eine Anwendung bündelt eine Vielzahl von Angeboten und stellt einen direkten und unkomplizierten Zugang zu den unterschiedlichen Verkehrsmitteln sicher. Wechselt der Kunde das Transportmittel, wie beispielsweise Bahn, Bus, Fahrrad oder Auto, muss er sich nicht mit unterschiedlichen Apps oder Anbietern auseinandersetzen.

Erfolgreich in die digitale Zukunft mit Cloud Native

Die zentrale Herausforderung des Projektes bestand darin, eine robuste und skalierbare Plattform zu schaffen, die unterschiedlichsten Applikationen und Services einen gemeinsamen betrieblichen Rahmen bietet. Außerdem sollte die zu Grunde liegende Infrastruktur offen sein für veränderte Anforderungen, die sich aus dem agilen Software-Entwicklungsprozess ergeben, in dem unterschiedliche Teams aus Mobilitäts-, Software- und Infrastruktur-Experten zusammenarbeiten.

Vom Re-Platforming zum Re-Factoring: Das Projekt verlief sehr dynamisch, sodass sich wesentliche Erkenntnisse erst während der Projektlaufzeit ergaben. So war das ursprüngliche Ziel für den Projektstart das Re-Platforming der Java-basierten monolithischen Applikation auf Basis von Container-Technologien. Die Software selbst sollte nur durch ein entsprechendes Tooling ergänzt, aber nicht selbst restrukturiert oder angepasst werden.

Der Ansatz, die Code Base der Applikation auszuklammern, erwies sich jedoch als unzureichend. Der Monolith verhinderte durch enge Koppelung und mangelnde Automatisierung, dass unabhängige Teams parallel verschiedene Komponenten der Applikation weiterentwickeln konnten. Das Potential der Cloud- und Container-basierten Plattform ließ sich auf diese Weise nicht „ausspielen“. Aus einem bloßen „Re-Platforming“-Projekt wurde ein umfassendes „Re-Factoring“, also ein Re-Design der Architektur und des Codes zur Erstellung Cloud-nativer Anwendungen.

Hierzu mussten die vorhandenen unterschiedlichen Deployments vereinheitlicht und der bislang klassische monolithische Betrieb durch eine Microservice-Architektur abgelöst werden. Hinzu kam die Bereitstellung von zusätzlichen Services wie Loadbalancing, Firewalling, Monitoring und Logging sowie das Management der Anwendungen. Denn die klassischen Managed Services hatten nichts von ihrer Bedeutung für einen reibungslosen und stabilen Betrieb verloren, sondern mussten auch in der neuen containerisierten Umgebung gewährleistet und entsprechend angepasst werden.

Flexible Zusammenarbeit auf Basis von „Shared Responsibility“: Um beiden Zielen, der Stabilität im Betrieb und der Agilität in der Entwicklung, gerecht zu werden, führte Claranet ein Shared-Responsibility-Modell auf Basis granularer Service-Level und spezifischer Automatisierungslösungen ein. Der Kunde kann sehr flexibel und schnell zwischen fully-managed, co-managed und self-managed Ansätzen wählen. Während beim ersten Ansatz die SLA auf die Applikation gegeben werden, beziehen sich beim co-managed Ansatz die SLA auf den Kubernetes/Container Level. Technisch realisiert wird dies durch eine von Claranet implementierte und bereitgestellte Deployment- und Microservice-Orchestrierungs-Lösung auf Basis von Gitlab, die das Fundament für das Shared-Responsibility-Modell darstellt. Der Kunde selbst hat die Kontrolle über die Orchestrierung und Konfiguration der Microservices. Claranet als Managed Service Provider kümmert sich um Reviews, Approvals und natürlich das finale Deployment in die Produktionsumgebung.

Cloud Native Tooling als Fundament: Um die Anforderungen an die Performance optimal abzubilden, werden die Microservices auf der skalierbaren Google Cloud Platform betrieben. Über Scale-Up- und Scale-Out-Mechanismen kann die Umgebung jederzeit dem aktuellen Bedarf angepasst werden. Das Container-Management erfolgt über die Kubernetes-Orchestrierungsoberfläche unter Zuhilfenahme der Google Kubernetes Engine (GKE). Die Microservice-basierte Landschaft kann der Kunde über definierte Schnittstellen via Deployment Pipelines in weiten Teilen selbst steuern. Von außen werden die Microservices über ein API Gateway adressiert. Der Microservices Stack wurde in drei Umgebungen ausgeliefert (Development, Staging und Produktion).

Zur Konkretisierung noch einige technische Details: Sämtliche Anwendungen, deren Abhängigkeiten und konkrete Infrastrukturausprägungen werden im Rahmen eines Deployments als Helm Charts codiert, als Clustering-/HA-Konzept entworfen und für die einzelnen Anwendungen umgesetzt.

Darüber hinaus sorgte Claranet für einen stabilen und performanten Betrieb, der auch für agile Umgebungen essenziell ist. So wurden unterschiedlich skalierende Node Pools erstellt, die vom Laufzeitverhalten ähnliche Applikationen gruppieren und auf den gleichen Ressourcen betreiben. Im Rahmen des Capacity Managements wurden darüber hinaus umfassende Maßnahmen ergriffen, um die Plattform gegen jegliche Art von Workload-Überlast abzusichern. Dazu zählen Namespace-Isolation durch Quotas oder der Einsatz von Horizontal Pod Autoscaler. Hinzu kamen umfassende Security Services wie gehärtete Cluster Nodes, Anwendung von Admission Controller, Konfiguration von Network Policies zur Microsegmentation, Container Image Vulnerability Scanning und natürlich der umfassende Einsatz von konsistenten Berechtigungskonzepten.

Die Zusammenarbeit mit dem Kunden und anderen Dienstleistern basierte auf einem agilen Ansatz mit entsprechenden Tools zum Fortschrittstracking und kurzen Release-Zyklen, um frühestmögliches Feedback in der Erstellung der Plattform und Services geben zu können.

Durch den Einsatz von Cloud Native-Applikationen auf Basis der Google Cloud wurde eine hochmoderne Mobilitätsplattform realisiert, die große Freiheitsgrade beim Management der Microservices-Landschaft ermöglicht, klassische Managed Services umfasst und auch Antworten auf zukünftige Herausforderungen bereithält.

CHECKLISTE FÜR DIE CLOUD NATIVE TRANSFORMATION

Wie die Case Study zeigt, bieten Cloud Native-Technologien zahlreiche Vorteile und die Chance, die Geschwindigkeit und Customer Experience signifikant zu verbessern. Dennoch ist ihr Einsatz im Unternehmen eine komplexe Aufgabe, bei der unterschiedliche Faktoren und Rahmenbedingungen zu berücksichtigen sind. Im Folgenden einige Hinweise:



Checkliste zur Umsetzung von Cloud Native-Projekten

- **Business Case:** Man sollte im Vorfeld des Projektes prüfen, bei welchen Szenarien und Applikationen sich Cloud Native-Technologien mit Erfolg einsetzen lassen. In vielen Unternehmen weist die IT-Landschaft zahlreiche bestehende Anwendungen auf. Nicht immer ist die Einführung von Cloud Native-Technologien technisch oder wirtschaftlich sinnvoll.
- **Richtiges Migrationsmodell:** Die Migration kann, abhängig von den konkreten Anforderungen, unterschiedliche Ausprägungen haben. Während es sich bei Greenfield-Projekten empfiehlt, direkt mit einer Microservice-Architektur zu starten, kann für Legacy-Projekte die Anpassung der Plattform durch Re-Factoring oder der Umbau der Applikation via Re-Factoring der geeignete Ansatz sein.

Erfolgreich in die digitale Zukunft mit Cloud Native

- **Erfolgsfaktor DevSecOps:** Sorgen Sie dafür, dass die Anforderungen von Development und Operations als auch Security in gleichem Maße berücksichtigt und von Beginn an in das App- bzw. Service-orientierte Lifecycle Management integriert sind. So gewährleisten Sie, dass sich Entwicklung und Auslieferung der Software beschleunigt, ohne Verfügbarkeiten und Sicherheit zu reduzieren.
- **Sichere Bereitstellung und Pflege:** Containerisierte Systemlandschaften sollten stets auf Informationssicherheits-Standards wie Identity Access Management (IAM), Verschlüsselung und Monitoring setzen.
- **Release Management:** Es gilt die Freigabe-Prozesse für Applikationen im Hinblick auf deren Container-Sicherheit zu überdenken, um dem Business das Erreichen schnellerer Release-Zyklen sowie ein beschleunigtes, aber dennoch sicheres Go-to-Market zu ermöglichen. Freigaben sollten auf Basis nachweisbar genutzter Sec-Mindestanforderungen erfolgen (agiles Pentesting/Bug Bounty).
- **Agiles Service Management:** DevOps ist der Schlüssel für die agile Zusammenarbeit zwischen Ops und Dev. Doch wie sieht diese Zusammenarbeit konkret aus, wenn man nicht auf die Vorteile von ITIL wie Nachvollziehbarkeit, Transparenz und Reporting bei jeglichen Geschäftsvorfällen verzichten möchte? Wird bei der Wahl eines Dienstleisters auf agiles Service Management geachtet?
- **Aufbrechen von Silos:** Neue Services sollte man nicht in Silos betrachten und managen: Beachten Sie auch herkömmliche Übergabe- und Kooperationspunkte, um diese in einem Microservice-orientierten Dienstleistungsszenario und einem Container Workflow zu harmonisieren.
- **Software is Key:** Man sollte konsequent auf Software Defined-Technologien und Infrastrukturen bzw. Plattformen setzen, die entweder eine Public Cloud sind oder deren Prinzipien entsprechen.
- **Konsistenz der Umgebung:** Die Sicherstellung der Anwendungskonsistenz und die Optimierung der zugehörigen Umgebung ist entscheidend, um den größtmöglichen Nutzen aus der Technologie zu ziehen.
- **Effizientes Management:** Container-Orchestrierungs- und Management-Tools wie Kubernetes bieten eine perfekte Möglichkeit, Container effizient zu managen. Da diese jedoch oft komplex sind und sich dynamisch verändern, kann die Einbeziehung von spezialisierten Dienstleistern hilfreich sein.

PARTNER SPONSOR

Claranet

Claranet unterstützt Unternehmen mit innovativen Cloud-Hosting-, Cyber-Security-, SAP- und Netzwerk-Services bei ihrer Digitalisierung. Der Managed Service Provider ist darauf spezialisiert, unternehmenskritische Umgebungen auf flexiblen Cloud-Infrastrukturen zu hosten und unter höchsten Sicherheits-, Performance- und Verfügbarkeitsanforderungen agil zu betreiben. Mit über 3.000 Beschäftigten realisiert Claranet große Hosting-Lösungen in 40 Rechenzentren sowie auf Public-Cloud-Umgebungen wie zum Beispiel von AWS, Google Cloud und Azure.

Weitere Informationen gibt es unter www.claranet.com/de

claranet[®]

ABOUT ISG

ISG (Information Services Group) (Nasdaq: III) ist ein führendes, globales Marktforschungs- und Beratungsunternehmen im Informationstechnologie-Segment. Als zuverlässiger Geschäftspartner für über 700 Kunden, darunter 75 der 100 weltweit größten Unternehmen, unterstützt ISG Unternehmen, öffentliche Organisationen sowie Service- und Technologie-Anbieter dabei, Operational Excellence und schnelleres Wachstum zu erzielen. Der Fokus des Unternehmens liegt auf Services im Kontext der digitalen Transformation, inklusive Automatisierung, Cloud und Daten-Analytik, des Weiteren auf Sourcing-Beratung, Managed Governance und Risk Services, Services für den Netzwerkbetrieb, Design von Technologie-Strategie und -Betrieb, Change Management sowie Marktforschung und Analysen in den Bereichen neuer Technologien. 2006 gegründet, beschäftigt ISG mit Sitz in Stamford, Connecticut, über 1.300 Experten und ist in mehr als 20 Ländern tätig. Das globale Team von ISG ist bekannt für sein innovatives Denken, seine geschätzte Stimme im Markt, tiefgehende Branchen- und Technologie- Expertise sowie weltweit führende Marktforschungs- und Analyse-Ressourcen, die auf den umfangreichsten Marktdaten der Branche basieren.

www.isg-one.com

Let's connect NOW...



Erfolgreich in die digitale Zukunft mit Cloud Native