

Kubernetes im Unternehmen

Einsatzmöglichkeiten und Herausforderungen im IT-Betrieb



claranet

crisp
RESEARCH

Inhalt/

Executive Summary	3
#01 Von agiler Entwicklung zu Microservices und Container Orchestration	4
#02 Why: Strategische Gründe und Chancen für den Einsatz von K8S	7
Vorteile für das Business	11
#03 What: Einsatzbereiche und Use Cases für Container und K8S	12
Cloud-native Applications	13
Application Modernization für bestehende Workloads	14
Digital Platform Design/Architecture	15
Portierung von SaaS-Produkten	15
#04 How: Deploymentvarianten, Tools und Betriebskonzepte für K8S	16
Deployment of Choice:	
Von Native K8S bis K8S-as-a-Service	17
Tooling: Monitoring, Logging & Security	17
Operations Best Practice: DevOps	19
Integration CI/CD Pipeline	20
K8S - Make or Buy	20
Evaluation Criteria für K8S - Service Provider	20
Empfehlung	22
Über Claranet	24
Über Crisp Research	25
Über die Autoren	26
Kontakt & Copyright	27



Executive Summary /

- **Microservices und Container als Kerntechnologien der Digitalisierung** - Neue digitale Apps, Plattformen oder Geschäftsmodelle haben eines gemeinsam - sie verlangen ein hohes Maß an Agilität in Entwicklung und Betrieb. Microservices-Architekturen und Container sind zum Kernbaustein für innovative und skalierbare digitale Workloads geworden.
- **Kubernetes ist der De-Facto-Standard für Container-Orchestrierung** - Mit einer breiten Entwicklergemeinde und der Unterstützung der Cloud Native Computing Foundation (CNCF) hat es Kubernetes geschafft, sich bei Herstellern, Anwendern und Entwicklern als Standard für die Orchestrierung und das Management von Containern zu etablieren.
- **Kubernetes goes Enterprise** - Immer mehr Unternehmen setzen Kubernetes nicht nur für digitale Workloads, sondern auch in produktiven und unternehmenskritischen Szenarien ein. Für 2019 planen zwei Drittel der deutschen Mittelstands- und Großunternehmen (57 Prozent) den Einsatz von Kubernetes.
- **Allzweckwaffe Kubernetes - breites Spektrum an Use Cases** - Dabei profitieren nicht nur neue Digital- und "Cloud-Native"-Workloads vom Container-Betrieb auf Basis von Kubernetes. Auch die Modernisierung von bestehenden Applikationen lässt sich durch den Einsatz von Containern und Kubernetes beschleunigen.
- **Komplexität und Innovationsgeschwindigkeit als Herausforderung** - Die hohe Innovationsgeschwindigkeit und Komplexität im Kubernetes-Universum stellt viele Verantwortliche vor erhebliche Probleme. Denn das CNCF-Ökosystem umfasst hunderte Technologien und Tools. Auch Kubernetes ist als Open Source-Framework hinreichend komplex und verändert sich stetig. Einen stabilen 24/7-Betrieb zu gewährleisten ist daher nicht einfach und viele Unternehmen setzen deshalb auf die Unterstützung durch erfahrene Managed Service Provider.
- **Kubernetes schafft Standards in Dev, Test und Produktion** - Um Produkte und neue Features schnell in den Markt zu bringen (Time-to-Market), ist eine Continuous Delivery Pipeline unabdingbar. Ein wichtiges Fundament bildet dabei Kubernetes, da es für einen einheitlichen Standard auf allen Umgebungen sorgt und Container entsprechend effizient orchestrieren kann.
- **Potentiale freisetzen und Investitionen optimieren** - Kubernetes bietet durch eine schnellere Feature-Geschwindigkeit kombiniert mit granularerer Infrastrukturnutzung eine gute Möglichkeit, Investitionen zu optimieren und Freiräume in den DevOps zu ermöglichen.

#01

Von agiler Entwicklung zu Microservices und Container Orchestration

#01

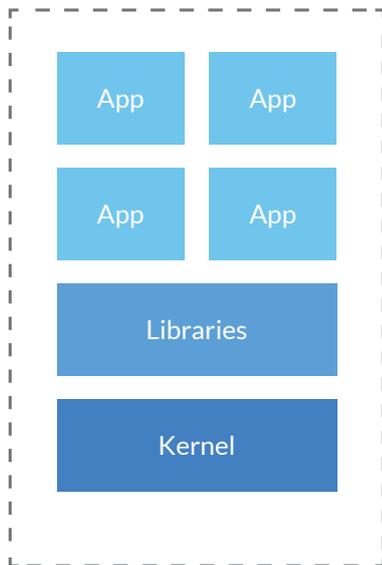
Software, traditionell für den Einsatz auf einem einzigen System gedacht, erlebt seit der Einführung von Cloud-Technologien eine Revolution. Skalierbarkeit, Hochverfügbarkeit, Multi-Tenancy, Multi-Cloud und Portabilität (Vermeidung eines Vendor Lock-in) sind die Top-Themen, mit denen sich CIOs, CTOs und IT-Architekten seit geraumer Zeit beschäftigen müssen, um die Erwartungen der Geschäftsseite zu erfüllen. Agile Entwicklungsprinzipien sind daher in den vergangenen Jahren auf dem Siegeszug - und das bei Software-Entwicklern und im IT-Betrieb.

Es wird das Ziel verfolgt, sich inkrementell an stetig ändernde Anforderungen anzupassen und schnell Innovationen in die Praxis umsetzen zu können. Dabei wird in der Softwareentwicklung in kleinen Iterationszyklen ein minimal lauffähiges Produkt (MVP) ausgeliefert. Um dieses Produkt auch in einem sehr kurzen Zeitfenster von klassischerweise zwei bis vier Wochen ausliefern zu können, benötigt man einen Architekturansatz, der dieser Anforderung Rechnung trägt und eine möglichst unabhängige Entwicklung von unterschiedlichen und vielleicht sogar weltweit verteilten Teams zulässt. Über die Jahre hinweg hat sich dazu der Ansatz von Microservices als ein ideales Architekturmuster herauskristallisiert.

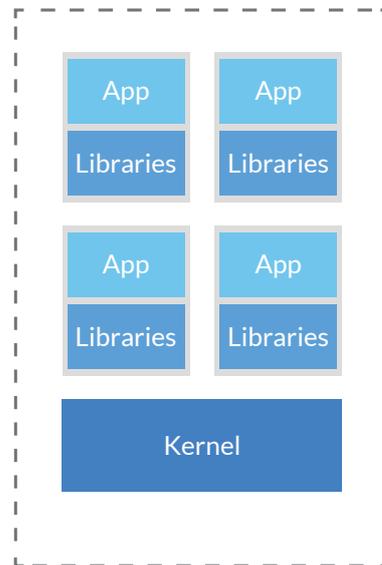
Die Idee ist dabei, komplexe Software in unabhängige und inhaltlich beschränkte Komponenten zu gliedern, welche dann untereinander mittels einer sprachunabhängigen Programmierschnittstelle (API) kommunizieren. Somit lassen sich einzelne Komponenten der Software unabhängiger zueinander entwickeln und auch individuell skalieren, welches eine fein granulare Anpassung an die Last ermöglicht. Doch je kleinteiliger die Applikation gestaltet wird, desto komplexer wird es, diese zu administrieren. Das Erfolgsrezept lautet an dieser Stelle Automation. Nur mittels eines hohen Automatisierungsgrads ist es möglich, moderne Microservices-Architekturen noch administrieren zu können.

Als Mittel zur Wahl für einen Betrieb von Microservices-Architekturen und -Applikationen haben sich über die letzten Jahre Container etabliert. Container haben den Vorteil, dass sie zum einen eine höhere Packungsdichte unterschiedlicher Applikationen auf einem Server ermöglichen und zum anderen auch gleich alle externen Abhängigkeiten mit beinhalten können. In dem Schichtenmodell in der folgenden Abbildung ist dies dargestellt. Der Container kann dann auf beliebigen Infrastrukturen ausgerollt werden. Doch mit einer steigenden Anzahl von Containern wächst auch der Aufwand und die Komplexität in der Orchestrierung auf die vorhandenen Ressourcen. Diese sollten effektiv genutzt werden und auch der Application-Lifecycle sollte kontrolliert werden können. Daher haben sich Lösungen am Markt etabliert, welche diese Aufgaben erfüllen. Zum De-Facto-Standard hat sich dabei das von Google als Open Source Software freigegebene Kubernetes Projekt etabliert.

#01

Traditionelle Anwendungen
auf einem Server

Schwergewichtig, schwer portierbar, an den Host und die Bibliotheken und das Betriebssystem gebunden

Anwendungen mit
Container Deployments

Klein, portierbar und individuelle Bibliotheken - auch in unterschiedlichen Versionen möglich, nutzt Betriebssystem Virtualisierung

Kubernetes (K8S) stellt eine Plattform bereit, um Container auf Cluster von physischen oder virtuellen Maschinen zu betreiben. Dabei besitzt Kubernetes die Eigenschaft, eine komplette Cluster-Umgebung bzw. die darin betriebenen Applikationen über einen sogenannten Pod, der automatisiert erstellt werden kann, wiederherzustellen. In Verbindung mit dem Prinzip der Immutable Infrastructures besitzt Kubernetes Self-Healing Funktionen, indem einzelne Komponenten des Deployments stetig überwacht und im Zweifel selbstständig zerstört und neu erstellt werden können. Damit sind Themen wie die Wartbarkeit, Sicherheit und Zuverlässigkeit von Anwendungen garantiert. Dies ist zum Standard bei der Implementierung einer vollständigen, zuverlässigen, Container-basierten Infrastruktur in Produktionsumgebungen avanciert. Kubernetes kann u.a. zum Betrieb von Hybrid-Cloud-Szenarien genutzt werden, da die betriebenen Applikationen innerhalb des Container-Clusters technologisch nicht an einen bestimmten Cloud-Anbieter oder eine spezielle Umgebung gebunden sind. Die Hauptaufgabe von Kubernetes ist somit die Verwaltung und Orchestrierung von Containern innerhalb eines Clusters, der üblicherweise aus mindestens einem Kubernetes Master und multiplen Worker Nodes besteht. Kubernetes hat somit die Fähigkeit, Spezifika der verschiedenen Infrastruktur-Umgebungen zu abstrahieren und für den Betrieb von containerisierten Anwendungen eine einheitliche Schnittstelle bereitzustellen.

#02

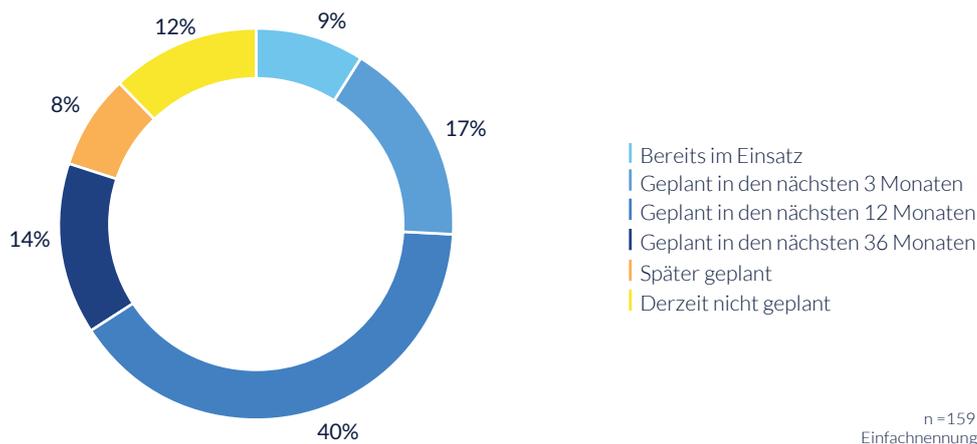
Why:
Strategische
Gründe und
Chancen für den
Einsatz von K8S

#02

Kubernetes ist gekommen, um zu bleiben und hat sich binnen kurzer Zeit zum De-Facto-Standard für die Nutzung von Containern im Unternehmen entwickelt. Da viele Unternehmen an der Entwicklung beteiligt sind, ist auch die Weiterentwicklung in Hinblick auf wichtige Funktionalitäten für Unternehmen bereits vollzogen oder in der Planung bzw. Umsetzung. Besonders relevant für einen produktiven Betrieb sind die Themen Sicherheit, Storage und Backup.

Die meisten Unternehmen in Deutschland planen ein Container-Management-Tool einzusetzen und entweder bestehende Anwendungen aus virtuellen Maschinen (VMs) für die Container-Technologie anzupassen und ggf. dahin zu migrieren oder neue Anwendungen direkt in Containern zu betreiben. Um den Einsatz im Unternehmen entsprechend zu planen und abzuwägen, sind die Stärken und Schwächen von Kubernetes im Folgenden noch einmal angeführt.

// Bis wann wollen Sie Kubernetes bzw. ein Container-Management-Tool im Unternehmen einsetzen?



Chancen beim Einsatz von Kubernetes

- Kubernetes verfügt über ein klares Governance-Modell, das von der Linux Foundation verwaltet wird. Google treibt die Produktfunktionen und die Roadmap aktiv voran, während der Rest des Ökosystems teilnehmen kann.
- Ein wachsendes und dynamisches Kubernetes-Ökosystem gibt den Unternehmen Vertrauen in die langfristige Stabilität und Innovationskraft. Google, Microsoft, Huawei, IBM, Intel und Red Hat sind einige der Unternehmen, die wichtige Beiträge zu dem Projekt leisten.
- Trotz des starken Anstiegs bei kommerziellen Distributionen vermeidet Kubernetes die Abhängigkeit und die Einbindung von Anbietern durch aktive Beteiligung der Gemeinschaft und Unterstützung von offenen Ökosystemen und Technologien.

#02

- Kubernetes ist weniger verbindlich als andere PaaS-Angebote. Das flexible Service Discovery- und Integrationsmodell erleichtert Entwicklern das Packen vorhandener Anwendungen für Kubernetes.
- Einfache und effizientere Erstellung von Container-Images im Vergleich zur Verwendung von VM-Images.
- Kontinuierliche Entwicklung, Integration und Bereitstellung (CI/CD): Kubernetes unterstützt das zuverlässige und häufige Erstellen und Bereitstellen von Application Stacks innerhalb von Containern sowie schnelle und einfache Rollbacks.
- Entkoppelung der Anwendungen von der Infrastruktur, über eine einheitliche Schnittstelle zu Compute, Network und Storage-Ressourcen in den unterschiedlichen Cloud-Kontexten.
- Informationen und Metriken für die Betriebssystemebene und den Zustand der Anwendung.
- Konsistenz der Anwendungsumgebung von der Entwicklung über Test bis hin zur Produktion.
- Erhöhung der Abstraktionsebene für den Betrieb von Anwendungen, wodurch Multi- und Hybrid-Cloud-Szenarien ideal umgesetzt werden können.
- Anwendungen werden in kleinere, unabhängige Teile aufgeteilt und können dynamisch bereitgestellt und verwaltet werden - nicht ein monolithischer Stapel, der auf einer großen Einzweckmaschine ausgeführt wird.
- Vorhersehbare Anwendungsleistung durch Isolation der Ressourcen.
- Hohe Effizienz und Packungsdichte führen zu einer effizienteren Auslastung und Nutzung von Ressourcen und dadurch indirekt auch zu Kosteneinsparungen bei der Infrastrukturnutzung.

#02

Risiken und Einschränkungen beim Einsatz von Kubernetes

- Kubernetes ist bereits für viele Workloads als Orchestrierungs-Plattform sinnvoll, dennoch sind zustandsbehaftete Anwendungen (stateful vs. stateless), Big Data-Workloads und die Nutzung von transaktionsbasierten Datenbanken aktuell noch mit Vorsicht zu genießen.
- Auch bei der Einbindung einer einheitlichen Speichersicht für das gesamte Cluster besitzt Kubernetes noch Schwächen. Die Community arbeitet aber an der Behebung der Schwachstellen durch unterschiedliche Integrationsmöglichkeiten von Netzwerk-Dateisystemen bis hin zu Objektspeicher bei den großen Public Cloud Providern.
- Die fehlende Unterstützung für Windows ist eine weitere große Lücke im Kubernetes-Ökosystem. Es gibt keine Anbieter, die Integrationen mit Windows-Containern und Hyper-V-Containern anbieten, die in der Microsoft-Umgebung ausgeführt werden.
- Kubernetes ist noch ein relativ junges Projekt, und es gibt Bedenken hinsichtlich seiner Verwendung in der Produktion. Es gibt jedoch viele Beispiele von erfolgreichen Produktivsystemen. Für Benutzer, die mit Kubernetes noch nicht vertraut sind, besteht kein Zwang, es sofort in Produktionsumgebungen einzusetzen.
- Kubernetes unterstützt keine echte Mehrmandantenfähigkeit. Kunden müssen Workarounds finden, um Workloads bereitzustellen, die eine starke Isolation und unabhängige Governance-Modelle erfordern. Hier können erfahrene Beratungspartner und Managed Service Provider unterstützen.
- Die Fragmentierung von Plug-ins und Add-Ons ist eine kritische und nicht zu unterschätzende Herausforderung, ebenso wie es aktuell mit allen Open Source-Projekten der Fall ist. Viele Tools werden aktiv gewartet, während andere eingestellt werden. Ein nachhaltiges Lifecycle-Management und die Einführung von Reliability-Checks gilt es auch im Kubernetes-Umfeld zu etablieren.
- Die besonders hohe Innovationsgeschwindigkeit des gesamten Ökosystems erschwert standardisierte Ansätze für das Cluster und Workload Management. Daher setzen viele Unternehmen bei der Umsetzung der Kubernetes-Strategie auf zertifizierte Service Provider, die schon einige Erfahrungen während der vorangestellten Release-Zyklen sammeln konnten.

#02

Die Innovationskraft von Kubernetes zeigt sich zum einen in der rasanten Entwicklung von Kubernetes selbst, aber auch in der Adaption durch Anwender und Anbieter. Die Entwicklungsgemeinschaft ist in den letzten Jahren stark gestiegen und die Anzahl der Unternehmen, welche Kubernetes im produktiven Einsatz haben ebenso. Dennoch ist immer noch viel Unterstützung von Dienstleistern oder kommerziellen Angeboten notwendig, um den kompletten Lebenszyklus von Anwendungen zu administrieren.

Vorteile für das Business

Der Geschäftswert von Kubernetes entsteht nicht durch den Einsatz um seiner selbst willen. Stattdessen entsteht der Wert aus der Fähigkeit, die Bereitstellung neuer Funktionen zu beschleunigen, die Widerstandsfähigkeit einer Infrastruktur zu verbessern und einen Großteil der manuellen und aufwendigen Arbeit beim Aufbau, dem Betrieb und der Wartung von Produktionsumgebungen vom Menschen auf Software zu verlagern. Kubernetes macht die Nutzung und Wartung für Entwickler und Administratoren einfacher und effizienter, wodurch freie Zeiträume entstehen, welche wiederum in das Produkt fließen können. Eine höhere Feature-Geschwindigkeit bedeutet einen schnelleren Time-to-Market zu realisieren, während eine verbesserte Widerstandsfähigkeit weniger Incident-Management und Fehlersuche bedeutet. Zusammenfassend entstehen aus den folgenden Punkten Vorteile für den Geschäftswert:

- Höhere Agilität durch Continuous Integration / Deployment
- Mehr Stabilität und weniger Fehler durch Vereinheitlichung von Dev / Test / Prod-Umgebung, sowie "Self-Healing"-Funktionen in Kubernetes
- Effizienzsteigerung der Admin- und Entwicklungsteams
- Unterstützung von Hybrid- und Multi-Cloud-Betrieb
- Optimierung von Cloud- und Hardware-Investitionen durch bessere Auslastung durch höhere Packdichte
- Benutzerfreundlichkeit und Portabilität
- Skalierbarkeit und Modularität
- Replizierbarkeit: Infrastrukturkomponenten und Stacks können aufeinander abgestimmt und wiederverwendet werden, um bspw. Kostenvorteile zu generieren

#03

What:
Einsatzbereiche
und Use Cases
für Container
und K8S

#03

Container und Kubernetes versprechen für Unternehmen ein großes Innovations- und Automationspotenzial. Jedoch stellt sich beim Blick auf das eigene Rechenzentrum oder auf die eigene IT-Landschaft sehr schnell die Frage, ob die Technologien überhaupt für die eigenen Anwendungen geeignet sind. Besonders in Hinblick auf bereits entwickelte Anwendungen ist diese Fragestellung mehr als berechtigt. Somit stellt sich die Frage, in welchen Anwendungsszenarien sich Kubernetes und Container sinnvoll einsetzen lassen.

Cloud-native Applications

Neue Software sollte heutzutage in der Regel mittels einer Microservice-Architektur entwickelt werden. Die einzelnen Bestandteile der Anwendungsarchitektur werden dabei in einzelne logische Einheiten aufgeteilt, damit diese dann möglichst unabhängig voneinander agieren, sowie entwickelt, verwaltet und betrieben werden können. Damit erhält man die Fähigkeit, die einzelnen Anwendungskomponenten individuell aktualisieren und skalieren zu können. Die einzelnen Microservices werden in Containern ausgerollt und diese dann mit Kubernetes orchestriert. Damit eignet sich Kubernetes für alle modernen und neuen Anwendungen sehr gut. Weiterhin lassen sich die Erstellung der Container, das Testing der Container, die Bereitstellung und der Roll-out direkt in einen Continuous Delivery Prozess integrieren und somit als Teil der Entwicklung professionalisieren - und automatisieren!

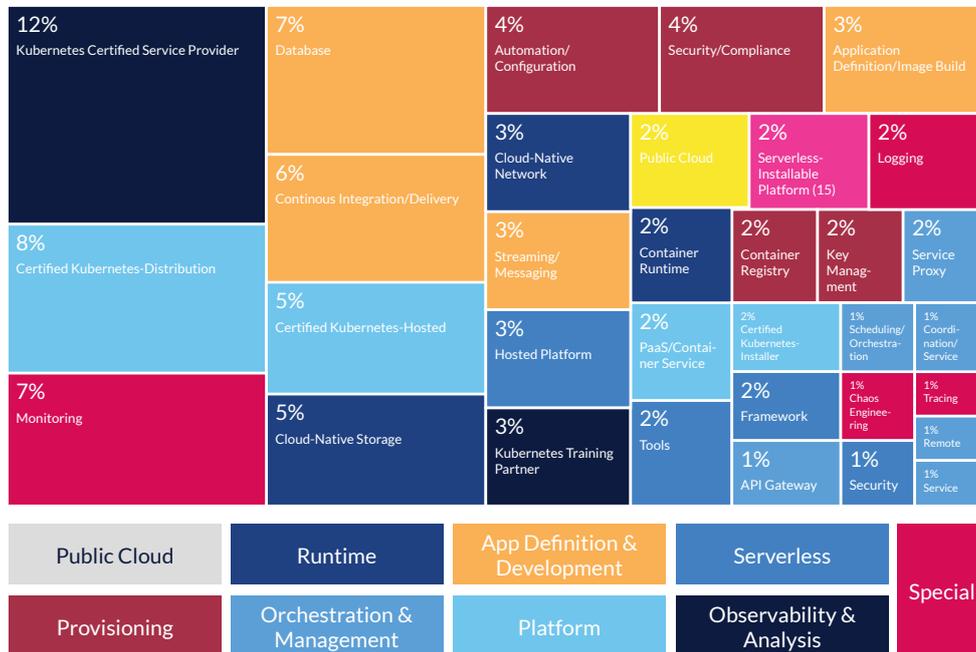
Cloud-native Technologien werden verwendet, um Anwendungen als Microservice-Architektur durch agile DevOps-Prozesse zu entwickeln, diese in Containern zu verpacken und auf einer elastischen Infrastruktur durch kontinuierliche Delivery-Workflows zu verwalten.

Da die Anzahl der Tools und Technologien in diesem Umfeld immer unübersichtlicher wurde und auch einige Entwickler alleine einzelne Projekte betreut haben, hat die Linux Foundation die Cloud Native Computing Foundation (CNCF) ins Leben gerufen.

Diese baut nachhaltige Ökosysteme auf und fördert eine Gemeinschaft um eine Konstellation von hochwertigen Projekten, die Container als Teil einer Microservice-Architektur orchestrieren. Ebenso wie die Technologien selbst, wächst auch die CNCF sehr schnell. Mittlerweile hat die CNCF über 375 Mitglieder (Stand: März 2019) und wächst weiter. Ebenso verhält es sich mit dem Ökosystem. Dieses hat sich über die Partner, aber auch mittels eigener CNCF-Zertifizierungen stetig weiterentwickelt und bedient damit alle Cloud-native relevanten Bereiche, die zur Erstellung und dem Betrieb einer Anwendung notwendig sind.

// CNCF Landscape - Projekte/Provider nach Kategorien

// Quelle: CNCF 2019

**Application Modernization für bestehende Workloads**

Nicht nur Neuentwicklungen sind für den Einsatz für Container-Technologien geeignet. Viele PHP, Java oder .Net basierten Anwendungen lassen sich ebenso von virtuellen Maschinen (VM) auf Container umziehen und dann mit Kubernetes verwalten. Dazu wird die existierende Anwendung mit Hilfe von Tools analysiert und dann ein Container Image für die VM erstellt. Auf diese Weise können klassische LAMP-Stacks (Linux, Apache, MySQL, PHP) oder auch Windows- und ASP.NET- und IIS-basierte Anwendungen portiert werden. Je nachdem wie viele Anwendungen auf diese Weise portiert werden sollen, lohnt sich hier auch ein automatischer Prozess mit integrierten Tests. Die VMs sollten dazu vorher geprüft werden, z.B. auf Passwörter, Konfigurationsdateien oder ähnlich sensible Daten, und dann nach dem Portieren auch noch einmal validiert werden, damit der Betrieb reibungsfrei weiterlaufen kann. Bereits in diesem "Lift & Shift"-Ansatz ist es in der Regel notwendig, einzelne Komponenten der Applikation auf ihre Container-Readiness und Portierbarkeit zu überprüfen. Meistens können die Lösungen zwar relativ gut migriert werden, in der Praxis müssen fast immer kleinere Anpassungen an den Anwendungen stattfinden, um ein Container-Szenario wirklich abbilden zu können. Daher macht es für viele Alt- bzw. Legacy-Anwendungen, die noch unternehmenskritische Prozesse abbilden, häufig Sinn, diese in Teilen oder komplett zu modernisieren. Sprich, Teile der Anwendungsarchitektur und Logik neu zu programmieren und zu strukturieren. Auf diese Weise halten Container und Kubernetes sukzessive

#03

Einzug in der Corporate- und Business-IT. Eine solche "softe" Transformation verringert zudem das Migrationsrisiko und ermöglicht einen sukzessiven Aufbau von Erfahrungen und Skills. Alternativ können IT-Dienstleister bei der Anwendungsmodernisierung und -Transformation unterstützen, um schneller die potenziellen Mehrwerte oder Kosteneinsparungen einzufahren.

Digital Platform Design/Architecture

Die Digitalisierungswelle hat mittlerweile alle Unternehmen erfasst und der Aufbau digitaler Plattformen ist für Unternehmen ein elementarer Schritt hin zu neuen digitalen Diensten und Produkten - und somit zu "Digital Revenue". Für eine digitale Plattform eignet sich der Einsatz von Kubernetes als eine Komponente der Plattform-Architektur sehr gut, da Kubernetes für den Betrieb von Hybrid- und Multi-Cloud-Szenarien bestens geeignet ist. Die Nutzer der Plattform haben dann die Möglichkeit, die Entwicklungen der Anwendungen nach denselben Entwicklungsmethoden vorzunehmen, egal auf welcher Infrastruktur oder Cloud gerade entwickelt, getestet oder produktiv gegangen wird. Damit bekommen alle im Unternehmen ein einheitliches Vorgehensmodell für die Entwicklung und eine einheitliche Abstraktionsebene (Container).

Portierung von SaaS-Produkten

Einige Unternehmen betreiben seit Jahren für die eigenen Kunden SaaS-Produkte und Dienste, welche in großer Anzahl und Vielfalt auf virtuellen Maschinen betrieben werden. Als Beispiel für einen SaaS-Provider kann ein Anbieter für Wordpress-Installationen dienen. Der Anbieter stellt Kunden individuelle Wordpress-Setups und -Hostings zur Verfügung und das mit verschiedenen Leistungsangeboten. Da die Spannbreite oftmals von privaten Nutzern bis hin zu großen Unternehmenskunden reicht, verteilt sich auch die Last auf den einzelnen virtuellen Maschinen dann entsprechend. Es wird Kunden geben, welche viele Dienste des Angebots nutzen und ebenso auch viel Ressourcen beanspruchen. Andere Kunden werden kaum Ressourcen nutzen und auch wenig Dienste beanspruchen. Damit entsteht für den Anbieter eine schwer zu kalkulierende Ressourcenausnutzung, wodurch die Auslastung der Maschinen im Mittel sehr gering ist. Auch in diesem Szenario eignet sich der Einsatz von Containern, da die einzelnen VMs zu Containern portiert werden können und dann effizienter durch Kubernetes verwaltet werden können. Auch Lastspitzen können dadurch effizienter adressiert werden.

#04

How:
Deployment-
varianten, Tools
und Betriebs-
konzepte für K8S

#04

Deployment of Choice: Von Native K8S bis K8S-as-a-Service

Wo sollte man Kubernetes betreiben und wo kann es überhaupt betrieben werden? Da Kubernetes immer schneller Einzug in die Unternehmen hält, gehen damit auch unterschiedliche Anforderungen an ein Deployment einher. Die Bereitstellung von Anwendungen und damit auch die Einsatzbereiche von Kubernetes sind daher nicht beschränkt auf die Public Cloud. Die Entwicklergemeinschaft und die Hersteller haben viel Energie darauf verwendet, Kubernetes in unterschiedlichen Ausprägungen und Kombinationsmöglichkeiten in die eigene IT-Landschaft zu integrieren. Viele Anwendungen sollen beispielsweise produktiv im eigenen Rechenzentrum eingesetzt werden. Die am weitesten verbreiteten Kubernetes Deployment-Modelle sind:

- Managed Kubernetes Service bzw. "Container as a Service" Angebot der globalen Cloud Provider (Produkt / Dienst der Cloud Provider)
- Managed Kubernetes Service als "hosted Service" eines unabhängigen Managed Service Providers (entweder auf IaaS-Basis der globalen Cloud Provider oder im Rechenzentrum des Hosters / Managed Service Providers)
- im On-Premises und Co-Location Umfeld
- als hybrides Deployment
- als Multi Cloud Deployment

Damit hat der Kunde die Wahl, die Verantwortung und die Aufgaben fein granular nach den eigenen Bedürfnissen zu schneiden. Viele Unternehmen nutzen einen hybriden Ansatz und modernisieren gleichzeitig mit der Einführung von Kubernetes auch die On-Premises Infrastrukturen, um auch im eigenen Hause dieselben dynamischen Bereitstellungsoptionen für Entwickler anbieten zu können.

Tooling: Monitoring, Logging & Security

Ist die Wahl für eine oder mehrere Deploymentvarianten erst einmal getroffen worden, gilt es Kubernetes in einem sinnvollen Setup zu administrieren. Dazu sind im Markt viele unterschiedliche Tools und Lösungen vorhanden. Die Landkarte der CNCF (<https://landscape.cncf.io/>) bietet hier eine breite Auswahl an Tools und Lösungen. Meistens wird das beim deutschen Unternehmen Soundcloud in Berlin entwickelte Prometheus (welches übrigens das zweite akzeptierte CNCF Projekt war) für Monitoring-Aufgaben genutzt. Aufgrund der Open Source-Charakteristik kann ein Unternehmen jedoch frei entscheiden, welche Tool-Stacks zum Einsatz kommen sollen oder durch

seinen Managed Service Provider zur Verfügung gestellt werden. Einige sind hier exemplarisch angeführt:

- **EFK Stack** (Fluentd, Elasticsearch, Kibana zur Analyse und Visualisierung von Logdaten und Monitoring)
- **Kubewatch** (Monitoring mit Anschluss an Slack)
- **Jaeger** (Tracing System für Transaktionen)
- **Prometheus** (Gängigste Monitoring Lösung mit Time Series Datenbank)
- **Kubernetes-Dashboard** (Kubernetes eigene GUI zur Verwaltung und Administration von Ressourcen wie Nodes, Namespaces, Volumes und etliches andere mehr)
- **Kube-state-metrics** (überwacht den Kubernetes-API-Server und generiert Metriken über den Status zahlreicher Kubernetes-Objekte, einschließlich Cron-Jobs, Konfigurationskarten, Pods und Knoten)
- **cAdvisor** (Container Performance Analyse)
- **Weave Scope** (generisches Monitoring- und Visualisierungs-Tool für Docker und Kubernetes)
- **fabric8** (fokussiert weniger auf die Schwerpunkte der Verwaltung von K8s Clustern, sondern adressiert die Belange einer CD-Umgebung)

Wie bereits erwähnt, sorgt das große Interesse und die breite Entwicklergemeinde dafür, dass auch Anforderungen aus der Unternehmens-IT schnell in das Projekt einfließen. Das Thema Security ist davon nicht ausgenommen. Die Sicherheit spielt eine zentrale Rolle, von der Konnektivität über Zugriffskontrolle bis hin zu Sicherheitsrichtlinien. Bei der Nutzung von Kubernetes sind folgende Punkte in Bezug auf die Sicherheit zu beachten:

- Definieren Sie Rollen für die Nutzung des Clusters?
- Nutzen Sie rollenbasierte Zugriffskontrollen (RBAC)?
- Nutzen Sie Frameworks für das Identifizierungsmanagement?
- Erstellen Sie sichere und performante Container Images (idealerweise mit Zertifizierung/Signierung)?

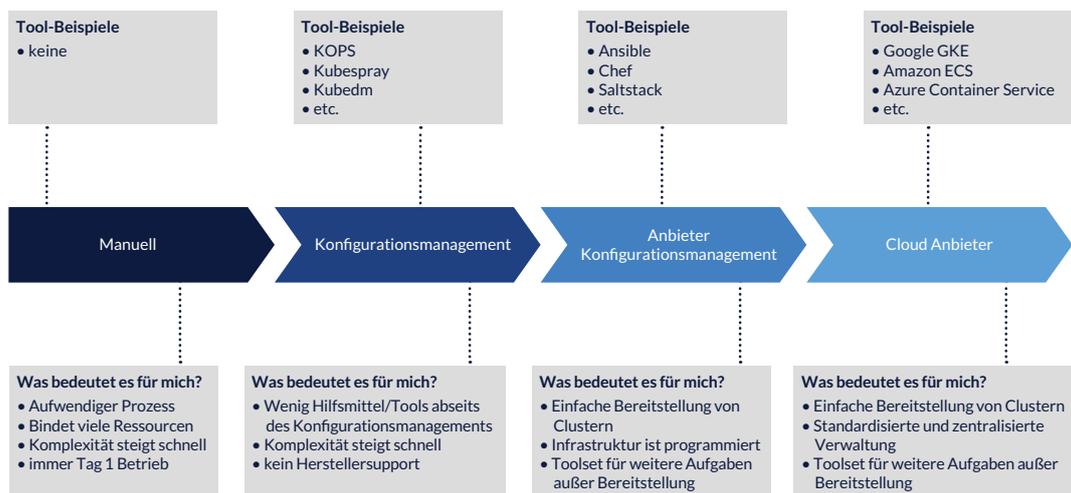
#04

- Überlegen Sie sich, wie Richtlinien auf Anwendungsebene umgesetzt werden sollen (Netzwerkrichtlinien- und Sicherheits-Richtlinien oder Service Meshes wie Istio oder Conduit)
- Soll eine Lösung zur Durchsetzung von Richtlinien eingeführt werden?

Darüber hinaus gibt es noch einige Tools, welche helfen, Sicherheitstests für Kubernetes-Deployments oder auch die Container auf Schwachstellen hin zu untersuchen. Auch hier können erfahrene Cloud- und Managed Service Provider meist weiterhelfen - vorausgesetzt, diese haben bereits Erfahrungen mit produktiven und unternehmenskritischen Kubernetes-Umgebungen gemacht.

Operations Best Practice: DevOps

Der Betrieb von Anwendungen hat sich über die Jahre hinweg weiterentwickelt ebenso wie die Programmiermodelle. Wo einst monolithische Blöcke auf einzelnen Servern administriert werden mussten, haben sich nun viele einzelne Microservices auf unterschiedlichen Infrastrukturen ausgebreitet. Die pure Anzahl der einzelnen Programme bedingt bereits einen hohen Grad an Standardisierung und Automatisierung. Dennoch gibt es von der manuellen Verwaltung bis hin zur Nutzung eines Cloud Anbieters einige Möglichkeiten, um Kubernetes zu verwalten.



Dabei kann man die einzelnen Möglichkeiten entweder direkt selbst durch das eigene Betriebsteam erbringen oder mit Hilfe eines Partners, der einen zusätzlich noch den 24/7 Support und andere Dienste mitliefert.

#04

Integration CI/CD Pipeline

Die Integration von Kubernetes und Containern in die CI/CD Prozesse bringt einen enormen Vorteil: Works on every machine! Mit Hilfe der Technologien ist es nun nicht mehr relevant, ob ein Container auf dem Rechner des Entwicklers getestet und betrieben wird, auf dem Entwicklungs-Cluster oder der Produktivumgebung. Auf allen Umgebungen sind die Voraussetzungen identisch und damit dann auch die Lauffähigkeit des Systems bzw. der Anwendung. Diese einheitliche Vorgehensweise spart auch Zeit bei der Konfiguration und hilft bei einem vollautomatisierten Entwicklungs- und Testbetrieb.

K8S - Make or Buy

Die Frage, ob man Kubernetes in Eigenregie oder doch als Dienst oder mit Hilfe eines Dienstleisters in die IT-Strategie integrieren sollte, hängt von vielen Faktoren ab. Zunächst sollte man sich die Frage stellen, wofür Kubernetes genutzt werden soll und in welchem Umfang? Daran schließt sich direkt die Fragestellung an, wo die Cluster betrieben werden und wie viele es werden können? Die personellen Ressourcen (Verfügbarkeit, Skills) runden die Fragerunde auf oberster Ebene ab. Bereits nach diesen wenigen Fragen kann meistens bereits die Entscheidung getroffen werden. Aufgrund der Komplexität tendieren viele Unternehmen gerade in der Anfangsphase dazu, Lösungen von Anbietern zu kaufen bzw. lizensieren und mit Hilfe von Dienstleistern bzw. Managed Service Providern zu starten. Da derzeit am Markt nur wenige Cloud-Architekten, DevOps und Site Reliability Engineers (SRE) mit ausgewiesener Kubernetes-Erfahrung verfügbar sind, empfiehlt es sich in der Anfangsphase, mit externen und erfahrenen Partnern die Cloud-Native- und Kubernetes-Reise zu beginnen.

Evaluation Criteria für K8S - Service Provider

Dies führt auch gleich zur Evaluation von Service Providern. Kubernetes legt eine enorme Geschwindigkeit bei der Entwicklung und dem Einzug in die Unternehmen vor. Daher ist es für Service Provider ebenfalls nicht so einfach, auch hier mit Skills und Ressourcen entsprechend mithalten zu können.

Mittlerweile gibt es auch Zertifizierungsprogramme beispielsweise von der CNCF, welche einen ersten Eindruck hinsichtlich der Expertise und Erfahrung eines Service Providers liefern können. Generell sollten Entscheider prüfen, wie viel Erfahrung ein Service Provider vorweisen kann und wie viele Ressourcen zur Verfügung gestellt werden können - zur Not auch vor Ort. Dabei ist es notwendig, neben der Zertifizierung auch auf tatsächliche

#04

Projekterfahrungen beim Aufbau und beim Betrieb zu achten. Hier gibt es durchaus große Unterschiede, inwieweit die Kubernetes-Expertise auch bis ins Produktivstadium hinein gegeben ist.

Auch in Hinblick auf das gewünschte Deployment-Modell sollte geprüft werden, ob ein Service Provider hier über die notwendigen Kenntnisse oder auch eigene Tools verfügt, um den Betrieb in Art und Umfang dann entsprechend sicher aufbauen und bereitstellen zu können. Darüber hinaus zeichnen sich exzellente Managed Service Provider und IT-Operations-Teams dadurch aus, dass sie auch weitere Aufgaben im Sinne des Kunden übernehmen:

- **Transformation:** Soll Kubernetes auch für bestehende Applikationen oder Plattformen im Sinne der Migration von virtuellen Maschinen hin zu Containern zum Einsatz kommen, dann empfiehlt es sich hier einen Service Provider auszuwählen, der auch im Bereich der klassischen Applikationen über Expertise verfügt, um die Modernisierung mit entsprechender Validität durchführen zu können.
- **DevOps:** Aufgrund der Komplexität wird die Zusammenarbeit auf Augenhöhe und auch in interdisziplinären Teams mit Partnern immer wichtiger. Daher bieten viele Service Provider auch eine Zusammenarbeit nach dem Vorbild von DevOps an. Das bedeutet, dass Personal des Service Providers eng mit den Kunden-Teams zusammenarbeitet und sich die Verantwortung für den Erfolg im Betrieb partnerschaftlich geteilt wird.
- **Compliance Management:** Die Erfüllung gesetzlicher Bestimmungen und unternehmenseigener Compliance-Standards im Hinblick auf den Kubernetes-Betrieb ist ein weiterer Aufgabenbereich, um den sich Managed Service Provider bzw. die IT-Operations-Teams kümmern sollten.

#05

Empfehlung

#05

Kubernetes wird sich in den nächsten Jahren stetig weiterentwickeln und auch Anwendungsfälle unterstützen, welche heute nur bedingt abbildbar sind. Unternehmen sollten daher nicht den Trend verschlafen und den Einsatz von Kubernetes prüfen und die Einführung vorbereiten. Gerade in Hinblick auf Hybrid- und Multi-Cloud-Lösungen bietet sich der Einsatz an und eröffnet neue Möglichkeiten. Denn der Einsatz von Containern und Kubernetes ist gerade seitens der Corporate-IT ein echter Innovationsbeitrag für die Umsetzung von unternehmensweiten Digital- und Innovationsprojekten.

Dennoch ist der Einsatz im Unternehmen keine triviale Aufgabe. Hier empfiehlt es sich zumindest in der Anfangsphase, auf erfahrene Dienstleister zu setzen. Weiterhin ist es sinnvoll, die folgenden Punkte zu berücksichtigen:

- **Providerwahl & Architektur-Konzept:** Bevor CEOs und CIOs mit der Umstrukturierung der IT-Landschaft beginnen, sollten geeignete Kubernetes Provider evaluiert und identifiziert werden. Gerade vor dem Hintergrund der Architekturkonzepte von Hybrid- und Multi-Cloud-Ansätzen ist die Auswahl der passenden Anbieter wichtig, um eine Ergänzung des Portfolios und die optimale Betriebsgrundlage für die zukünftigen Workloads sicherzustellen.
- **Workloads evaluieren:** Unternehmen sollten auch die bestehende Anwendungslandschaft bei der Evaluierung der Einsatzbereiche von Kubernetes berücksichtigen. Oftmals gibt es Workloads, die einfach von virtuellen Maschinen in Container migriert werden können.
- **Angebot in Deutschland nutzen:** Nicht nur internationale Großkonzerne bieten Kubernetes Lösungen und Dienste an. Gerade in Deutschland und Europa haben sich viel Unternehmen mit Expertise und Produkten am Markt etabliert, die sich nicht verstecken müssen. Daher sollten Entscheider ein besonderes Augenmerk auf die Anbieterlandschaft in Deutschland und Europa werfen.
- **Evolutionspfad – Was braucht es für den Kubernetes Betrieb?** Die zentrale Orchestrierung und ein einheitliches Management der zahlreichen Cluster auf unterschiedlichen Infrastrukturen im Unternehmen wird zur Kernherausforderung. Daher sollten die richtigen Tools, Produkte und Service Provider ausgewählt werden, die hier einheitliche Standards schaffen können unter Einhaltung von Compliance-Richtlinien.

Über Claranet/

Claranet unterstützt Unternehmen mit innovativen Cloud-Hosting 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. Claranet wurde in Gartners „Magic Quadrant 2018“ für „Data Centre Outsourcing and Hybrid Infrastructure Managed Services“ in Europa positioniert. Dies schließt an die Positionierung des Unternehmens als ein ‚Leader‘ in Gartners Magic Quadrant für „Managed Hybrid Cloud Hosting“ in Europa in fünf aufeinanderfolgenden Jahren an. Mit über 2.200 Mitarbeitern realisiert Claranet große Hosting-Lösungen in 43 Rechenzentren sowie auf Public Cloud-Umgebungen wie zum Beispiel von AWS, Google und Azure. Außerdem ist Claranet Gründungsmitglied des Kubernetes Certified Service Provider (KCSP) Programms und wurde von Crisp Research in der Vergleichsstudie „Vendor Universe Cloud Computing 2018“ als „Accelerator“ in dem Segment „Managed Kubernetes & Container Services“ ausgezeichnet.

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

claranet

Über Crisp Research/

Die Crisp Research AG ist ein unabhängiges IT-Research- und Beratungsunternehmen. Mit einem Team erfahrener Analysten, Berater und Software-Entwickler bewertet Crisp Research aktuelle und kommende Technologie- und Markttrends. Crisp Research unterstützt Unternehmen bei der digitalen Transformation ihrer IT- und Geschäftsprozesse.

Crisp Research wurde im Jahr 2013 von Steve Janata und Dr. Carlo Velten gegründet und fokussiert seinen Research und Beratungsleistungen auf „Emerging Technologies“ wie Cloud, Analytics oder Digital Marketing und deren strategische und operative Implikationen für CIOs und Business Entscheider in Unternehmen.

<https://www.crisp-research.com>



Über die Autoren/



bjorn.boettcher@crisp-research.com

Björn Böttcher

Senior Analyst & AI & Data Practice Lead

Björn Böttcher ist Senior Analyst bei Crisp Research. Er leitet als "AI & Data Practice Lead" die Research- und Beratungsaktivitäten zu den Themen Analytics, BI, datenbasierte Geschäftsmodelle und Künstliche Intelligenz. Als Gründer der ersten deutschen User Groups für Amazon AWS und Microsoft Azure zählt Björn Böttcher zu den Pionieren des Cloud Computing in Deutschland. Als Veranstalter der ersten deutschen Cloud-Konferenzen und Lehrbeauftragter für Informatik und Computational Web in der Parallel Computing Group der TU Hamburg hat er wesentliche Beiträge zur Entwicklung der Cloud-Community geleistet.

Björn Böttcher verfügt über 10 Jahre Berufserfahrung in der IT-Industrie in der Rolle des Software-Architekten und des IT-Strategieberaters. Zuletzt arbeitete er am Deloitte Analytics Institute und verantwortete dort die Entwicklung und Umsetzung datenbasierter Geschäftsmodelle für Unternehmen aus der Finanz-, Automotive- und Logistik-Branche. Björn Böttcher hat einen Abschluss als Dipl.-Informatikingenieur der Technischen Universität Hamburg-Harburg. Er hat als Autor eine Vielzahl von Fachbeiträgen publiziert und trägt als Key Note-Speaker und Experte aktiv zu den Debatten um neuen Markttrends, Standards und Technologien bei.



max.hille@crisp-research.com

Max Hille

Senior Analyst & Cloud Practice Lead

Maximilian Hille ist Senior Analyst und Practice Lead bei Crisp Research. Als Cloud Practice Lead leitet er alle Research- und Beratungsaktivitäten zu den Themen Cloud-Architektur, Cloud-Native Technologies, Managed Cloud Services, Digital Workplace und Mobility.

Zuvor war er Research Manager in der „Cloud Computing & Innovation Practice“ der Experton Group AG. Maximilian Hille studierte Wirtschaftswissenschaften mit dem Schwerpunkt Wirtschaftsinformatik.

Seine Schwerpunktthemen sind Cloud Platforms, Cloud Architecture Design, Hybrid & Multi Cloud Computing, Cloud-Native Architectures, Digital Workplace, Collaboration, Enterprise Mobility und Mobile Business. Maximilian Hille war Jurymitglied bei den Global Mobile Awards 2016, 2017 und 2018.

Kontakt/

Crisp Research AG
Weißenburgstraße 10
D-34117 Kassel

Tel +49-561-2207 4080
Fax +49-561-2207 4081
info@crisp-research.com

<http://www.crisp-research.com/>
https://twitter.com/crisp_research



Copyright/

Alle Rechte an den vorliegenden Inhalten liegen bei der Crisp Research AG. Die Daten und Informationen bleiben Eigentum der Crisp Research AG.

Vervielfältigungen, auch auszugsweise, bedürfen der schriftlichen Genehmigung der Crisp Research AG.



Crisp Research AG
Weißenburgstraße 10
D-34117 Kassel

Tel +49-561-2207 4080
Fax +49-561-2207 4081
info@crisp-research.com

<http://www.crisp-research.com/>
https://twitter.com/crisp_research

The logo for Crisp Research. It features the word "crisp" in a bold, lowercase, sans-serif font. The letter "c" is stylized with a blue circular graphic element. Below "crisp", the word "RESEARCH" is written in a smaller, uppercase, sans-serif font.

crisp
RESEARCH