

# DevOps und Cloud

## Synergieeffekte nutzen

- **Warum DevOps und Cloud zusammenwachsen**  
Die Basis effizienter Automatisierung
- **Sicherheit, Datenschutz und Compliance**  
Konzepte und Strategien im Cloud- und DevOps-Umfeld
- **Die Zukunft des IT-Betriebs**  
Wie sich Teamstrukturen, Rollen, Skills und Prozesse verändern müssen

### Extras

Wie DevOps, Cloud und Automatisierung die Zusammenarbeit in der IT verändern

+ Case Studies



# Editorial

In den vergangenen Jahren hat das Tempo in der Software-Entwicklung enorm angezogen. Release-Zyklen von wenigen Wochen, Tagen oder sogar Stunden sind heute keine Seltenheit mehr. Continuous Development, Integration und Deployment, also die kontinuierliche Entwicklung, Integration und Auslieferung von Applikationen sind wesentliche Erfolgsfaktoren, um am hart umkämpften Markt bestehen zu können. Diese Schnelligkeit ist am effektivsten erzielbar, wenn Software-Entwicklung und -Betrieb nach dem DevOps-Konzept zusammenarbeiten, und wenn jederzeit leistungsfähige und skalierbare Ressourcen aus der Cloud als Infrastructure as Code angesprochen werden können. In diesem Szenario haben auch manuelle Tests keinen Platz mehr, eine durchgängige Automatisierung der Tests und deren Integration in die Entwicklungs-Pipeline sind essentiell, um Qualität, Sicherheit und Compliance sicherzustellen.

DevOps ist jedoch kein reines Technik-Thema. Neben Architektur, Tools und Methoden muss auch die Kultur im Unternehmen stimmen. Manager müssen bereit sein, Verantwortung und Entscheidungsbefugnisse an die Teams zu übertragen. Die Teammitglieder müssen im Gegenzug lernen, mit dieser Verantwortung und dem daraus resultierenden Druck umzugehen.

Dieses eBook verdeutlicht, wie sich mit der Kombination aus DevOps, Cloud und Automation die Entwicklung und der Betrieb von Software schneller, sicherer und besser machen lassen, welche Teamstrukturen, Rollen und Skills dafür notwendig sind, und mit welchen Strategien Sie für Sicherheit, Datenschutz und Compliance im DevOps-Umfeld sorgen.

**Dr. Thomas Hafen**  
Freier Journalist

## © 2020 Heise Medien

Alle Rechte vorbehalten. Nachdruck, digitale Verwendung jeder Art, Vervielfältigung nur mit schriftlicher Genehmigung der Redaktion.

Heise Medien GmbH & Co.KG  
Abt. Heise Business Services  
Hans-Pinsel-Straße 10b  
85540 Haar bei München

Registergericht:  
Amtsgericht Hannover HRA 26709

Persönlich haftende Gesellschafterin:  
Heise Medien Geschäftsführung GmbH

Registergericht:  
Amtsgericht Hannover, HRB 60405

Geschäftsführer:  
Ansgar Heise, Dr. Alfons Schröder

Verantwortlich für den Inhalt:  
Heise Business Services  
Thomas Jannot, tj@heise.de

Haftung: Für den Fall, dass Beiträge oder Informationen unzutreffend oder fehlerhaft sind, haftet der Verlag nur beim Nachweis grober Fahrlässigkeit. Für Beiträge, die namentlich gekennzeichnet sind, ist der jeweilige Autor verantwortlich.

Haben Sie Fragen zu diesem eBook oder haben Sie Interesse an einer eigenen Produktion, dann schicken Sie bitte eine E-Mail mit dem Betreff „HBS-eBook“ an [hbs@heise.de](mailto:hbs@heise.de)

# Inhalt

<b>Warum DevOps und Cloud zusammenwachsen</b>	<b>4</b>
Die Erfolgsfaktoren	4
Basis für Prozessautomatisierung	5
DevOps, Cloud Computing & Automation: die sechs wesentlichen Vorteile	5
Fazit	6
<b>Sicherheit, Datenschutz und Compliance im Cloud und DevOps-Umfeld</b>	<b>7</b>
Risiko verborgene Schwachstellen	7
Data Protection by Design	8
Sicherheitsbewusstsein schaffen	8
Fazit	9
<b>Wie sich Teamstrukturen, Rollen, Skills und Prozesse verändern müssen</b>	<b>10</b>
Interdisziplinäre, cross-mediale Teams	10
Traditioneller IT-Betrieb verliert an Stellenwert	11
Was jetzt zu tun ist	13
Fazit	14
<b>Extras</b>	<b>15</b>
Whitepaper: Wie DevOps, Cloud und Automatisierung die Zusammenarbeit in der IT verändern	15
Case Studies: DevOps-Teams im Interview	19

## ÜBER DEN AUTOR



**Dr. Thomas Hafen** war über 15 Jahre als Redakteur, Moderator und Manager für verschiedene IT-Fachverlage tätig. Seine fachlichen Schwerpunkte liegen in den Bereichen Digitale Transformation, Cloud Computing und Advanced Analytics. Thomas Hafen lebt und arbeitet heute als freier Journalist und Moderator in München.



## Effiziente DevOps-Prozesse

# Warum DevOps und Cloud zusammenwachsen

Die Cloud virtualisiert IT-Ressourcen und stellt sie als Service zur Verfügung. Deshalb bildet sie die ideale Basis für effiziente DevOps-Prozesse und eine weitgehende Automatisierung des IT-Betriebs.

Schnelligkeit und Flexibilität sind wesentliche Faktoren, wenn es darum geht, in der digitalisierten Welt von heute wettbewerbsfähig zu bleiben. Im IT-Bereich ist es oftmals die Sollbruchstelle zwischen Softwareentwicklung und Softwarebetrieb, die schnelle und effiziente Prozesse verhindert. Häufig kommt es zwischen Entwicklungs- und Betriebsabteilung zu Interessenkonflikten und gegenseitigen Schuldzuweisungen. Aus diesem Grund fasst seit rund zehn Jahren das DevOps-Konzept immer mehr Fuß.

## Die Erfolgsfaktoren

Es ist wichtig zu verstehen, dass DevOps kein reines Technik-Thema ist. Vielmehr müssen Tools, Methoden, Architektur und Kultur zusammenspielen, um DevOps wirklich erfolgreich einsetzen zu können. Eine durchgängige Testautomatisierung und Qualitätssicherung sind weitere Faktoren für den Erfolg.

Während 2015 laut dem [RightScale 2016 State of the Cloud Report](#) erst 66 Prozent der Unternehmen DevOps-Prinzipien anwendeten, waren es der Marktforschungsseite [Statista](#) zufolge 2018 bereits 81 Prozent. DevOps vereint Softwareentwicklung und IT-Betrieb in einer Abteilung. Entwickler sind

”

Die Vereinigung von Softwareentwicklung und IT-Betrieb birgt handfeste Vorteile.

für die Bereitstellung ihrer Produkte und Lösungen mitverantwortlich und haben daher ein hohes Interesse daran, stabile lauffähige Lösungen herzustellen. Die bisher für den Betrieb zuständigen Mitarbeiter erhalten im Gegenzug ein sehr viel größeres Mitspracherecht in der Entwicklung und werden auch an deren Zielen gemessen. Die Vereinigung der beiden Welten führt laut Softwareanbieter [UpGuard](#) zu handfesten Vorteilen. Demnach konnten 63 Prozent der Unternehmen, die DevOps-Prinzipien implementiert haben, Softwarequalität und Release-Frequenz steigern. DevOps-Teams verbringen 60 Prozent weniger Zeit mit Support-Anfragen und dafür 33 Prozent mehr mit der Infrastrukturverbesserung als traditionelle Betriebsabteilungen. Laut einer [Umfrage, die Harvard Business Review im Auftrag von Google](#) durchgeführt hat, setzen bereits 48 Prozent der Unternehmen bei der Softwareentwicklung immer auf DevOps-Methoden, 21 Prozent tun dies teilweise, weitere acht Prozent planen den Einsatz.



Um die Effizienzgewinne auch wirklich realisieren zu können, setzt DevOps stark auf die Automatisierung der Entwicklungs-Pipeline und eine kontinuierliche Produktion (Continuous Development). Automations-Tools wie Jenkins oder Puppet, Source-Code-Management-Lösungen wie Git und Bitbucket oder mit Kubernetes gemanagte Container-Umgebungen spielen ihre Stärken besonders in der Kombination mit Cloud Computing aus. Die Cloud stellt Ressourcen virtualisiert als Service zur Verfügung. Infrastrukturkomponenten lassen sich daher als Infrastructure as a Code per Software definieren. Entwickler können damit direkt auf eine leistungsfähige und skalierbare Betriebsumgebung zugreifen, ohne lange Antrags- und Beschaffungswege für neue IT-Ressourcen in Kauf nehmen zu müssen.

## Basis für Prozessautomatisierung

Die Virtualisierung aller IT-Komponenten und deren softwarebasierte Konfiguration stellen darüber hinaus eine ideale Basis dar, um Prozesse nahezu oder sogar vollständig zu automatisieren. So können Unternehmen das DevOps-Prinzip zum „SmartOps“- oder gar „NoOps“-Betrieb weiterentwickeln. Autonom werden die Prozesse allerdings erst durch den Einsatz von Machine Learning und künstlicher Intelligenz. Die kognitive Automation des IT-Betriebs, auch AIOps genannt, reduziert den manuellen Monitoring-Aufwand, erhöht die Verfügbarkeit, erkennt und behebt Performance-Einbußen, bevor sie zu Problemen führen, und unterstützt IT-Verantwortliche dabei, bessere und schnellere Entscheidungen zu treffen.

## DevOps, Cloud Computing & Automation: die sechs wesentlichen Vorteile

Wenn IT-Prozesse mithilfe von DevOps-Prinzipien und Cloud-Technologien weitgehend automatisiert werden können, ergeben sich für Unternehmen im Wesentlichen folgende Vorteile:

### 1. Schnellere Marktreife von Produkten

Schlanke DevOps-Strukturen und die flexible Ressourcenbereitstellung über die Cloud ermöglichen es, Produkte sehr viel schneller an den Markt zu bringen, als dies mit herkömmlichen Methoden und Infrastrukturen möglich ist.

### 2. Weniger Komplexität

Automatisierung und Infrastructure as a Code vereinfachen den Systemaufbau und die Systemwartung.

### 3. Höhere Sicherheit

Automatisierte reproduzierbare Prozesse verhindern Flüchtigkeitsfehler und ermöglichen es, Security-Aspekte durch sämtliche Stadien der Softwareentwicklung hindurch zu berücksichtigen.

### 4. Weniger Ausfälle, kürzere Ausfallzeiten

Cloud-native Applikationen, die auf Microservices beruhen, sind wesentlich robuster als traditionelle monolithische Lösungen. Wenn DevOps-Teams Software direkt für den cloud-basierten Betrieb entwickeln, erhöht sich die Verfügbarkeit bei entsprechender Architektur signifikant.



### 5. Nahezu unbegrenzte Skalierbarkeit

Einer der Hauptgründe, warum Unternehmen auf die Cloud setzen, ist ihre Skalierbarkeit. Kapazität lässt sich mit wenigen Klicks erhöhen, bei nachlassendem Bedarf aber auch schnell wieder reduzieren. In Kombination mit DevOps wird Skalierbarkeit zu einem integralen Bestandteil der Anwendungsentwicklung. Sie ermöglicht die optimale Nutzung von On-Demand-Ressourcen, was die Infrastrukturkosten senkt und das Entwicklungstempo beschleunigt.

### 6. Höhere Innovationsgeschwindigkeit

Cloud-Ressourcen lassen sich schnell und einfach bereitstellen. Das ermöglicht die schnelle Umsetzung neuer Ideen und die Entwicklung neuer Geschäftsmodelle. Statt mühsam Services selbst zu entwickeln, können sich Unternehmen außerdem auf den Cloud-Plattformen aus einer Vielzahl bestehender Lösungen bedienen und sich so ganz auf ihre Kernkompetenzen konzentrieren.

## Fazit

Sowohl DevOps als auch Cloud Computing erhöhen die Wettbewerbsfähigkeit von Unternehmen. Während DevOps die Entwicklung und den Betrieb von Software optimiert, stellt die Cloud schnell, flexibel und bedarfsabhängig IT-Ressourcen als Service zur Verfügung. Allerdings spielen DevOps und Cloud erst gemeinsam ihre ganze Stärke aus. Ihr Zusammenspiel ermöglicht und beschleunigt nicht nur die Entwicklung neuer Produkte, sondern legt auch den Grundstein für eine durchgängig automatisierte Infrastruktur, bei der menschliche Eingriffe in den Betrieb nur noch in Ausnahmefällen notwendig sind. Diese Vorteile sind allerdings nicht ohne kulturellen Wandel in den Unternehmen und erhebliche Investitionen in Personal und Technologien zu haben. Angesichts zahlreicher Mitbewerber, die mit immer neuen Lösungen und Services in den Markt drängen, gibt es zu dieser grundlegenden Transformation jedoch keine wirkliche Alternative. ■

”

Die kognitive Automation des IT-Betriebs (AIOps) reduziert den manuellen Monitoring-Aufwand, erhöht die Verfügbarkeit und behebt Performance-Einbußen, bevor sie zu Problemen führen.



## Konzepte und Strategien

# Sicherheit, Datenschutz und Compliance im Cloud- und DevOps-Umfeld

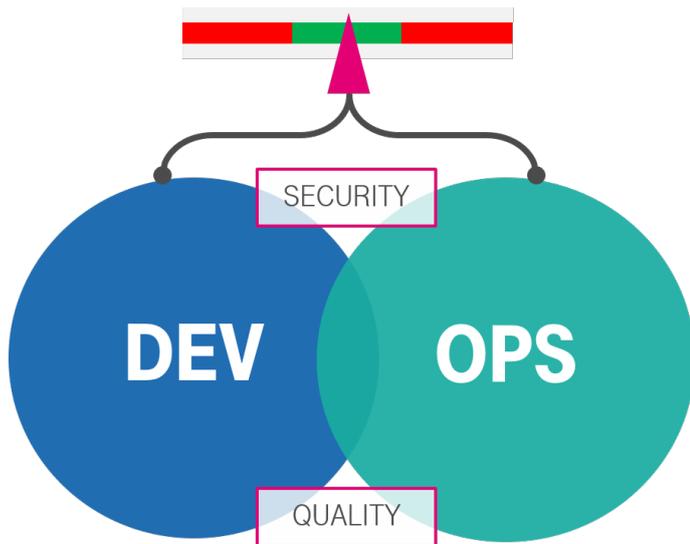
**DevOps und Automatisierung haben Qualität, Sicherheit und Compliance in der Softwareentwicklung deutlich verbessert. Doch Sicherheitslücken, Datendiebstahl und rechtliche Probleme lassen sich nur bei richtiger Implementierung zuverlässig vermeiden.**

Vor der Einführung von DevOps erfolgte die Softwareentwicklung üblicherweise in einem linearen Prozess, der aus vorab definierten und klar getrennten Phasen bestand. Für jeden Schritt war ein eigenes Team zuständig, das nach Abschluss seiner Arbeit das Projekt an die nächste Arbeitsgruppe übergab. In diesem als „Wasserfallmodell“ bekannten Paradigma gab es definierte Testabschnitte, die allerdings erst sehr spät in der Projektphase vorgesehen waren. Traten dabei Fehler, Sicherheitslücken, Datenschutz- oder Compliance-Probleme auf, kam es in der Regel zu erheblichen Verzögerungen bei der Fertigstellung, weil das Projekt oft um mehrere Stufen zurückgeworfen wurde. Bei der agilen Softwareentwicklung nach dem DevOps-Prinzip sind dagegen alle Phasen eng verzahnt und laufen teilweise parallel. Ein Team ist nicht mehr nur für einen separaten Abschnitt des Software-Lifecycle verantwortlich, sondern für die gesamte Entwicklungs-Pipeline bis hin zum Betrieb. Feedbacks von dort werden dann wieder in Sprintmeetings und Backlogs aufgenommen.

Dieser Ansatz hat die Softwareentwicklung erheblich beschleunigt. War es früher üblich, ein Major Release nur alle zwei Jahre herauszubringen und dieses höchstens alle sechs Monate zu aktualisieren, sind heute mehrere Updates pro Tag keine Seltenheit mehr. Zusätzlich beschleunigt wird dieser Prozess durch die Bereitstellung in der Cloud, die jederzeit auf Knopfdruck die notwendigen Ressourcen zum Beispiel für den raschen Aufbau und Wiederabbau von Testumgebungen zur Verfügung stellen kann. Diese Schnelligkeit ist mit manuellen Tests auf Sicherheit, Datenschutzkonformität und Compliance nicht vereinbar. Daher sind automatisierte, in die Pipeline integrierte Tests, wie sie für DevOps-Umgebungen typisch sind, unabdingbar.

## Risiko verborgene Schwachstellen

Während funktionale Fehler im Produktiveinsatz meist schnell entdeckt und behoben werden, bleiben Sicherheitslücken oder Compliance-Verstöße oftmals unbemerkt und führen erst Monate oder Jahre später zu Cyber-Attacken, Datenverlust und rechtlichen Auseinandersetzungen. So wurde beispielsweise bei einem Unternehmen der Automobilzulieferindustrie ein AKS-Cluster in Azure aufgesetzt und ohne Prüfung über das Internet erreichbar geschaltet. Der Prozess wurde automatisch als Blueprint für das restliche Unternehmen platziert und damit 1.000-fach ausgerollt, bevor der Fehler



Sicherheit muss ein integraler Bestandteil des DevOps-Prozesses werden. (Quelle: T-Systems Multimedia Solutions)

auffiel. Dieses Beispiel zeigt, dass es zu erheblichen Fehlern kommt, wenn der DevOps-Gedanke einschließlich Automatisierung nicht konsequent zu Ende gedacht und gelebt wird.

Um solche Probleme zu verhindern, müssen Security, Datenschutz und Compliance im Sinne eines DevSecOps natürliche Bestandteile des DevOps-Prozesses werden. Eine zentrale Rolle spielt dabei die CI/CD-Pipeline (Continuous Integration / Deployment), die die technische Basis von Entwicklung und Betrieb bildet. Sicherheitsmaßnahmen wie Code Scans und Penetrationstests sind in die Pipeline zu integrieren und regelmäßig durchzuführen. Nur so lassen sich Sicherheitslücken schnell und zuverlässig entdecken, Datenlecks vermeiden und die Einhaltung von Compliance-Vorgaben sicherstellen.

## Dataprotection by Design

Dem Datenschutz muss darüber hinaus nach Vorgaben der Europäischen Datenschutz-Grundverordnung (DSGVO) bereits bei der Konzeption einer Software Rechnung getragen werden. Sie fordert in Artikel 25 die technische Implementierung von Datenschutzmechanismen (Data Protection by Design). Personenbezogene Daten sind nach diesen Vorgaben so weit wie möglich von vornherein zu pseudonymisieren, auf das Nötigste zu begrenzen, so kurz wie möglich zu speichern und durch geeignete Zugangsbeschränkungen abzusichern.



Der aktuelle Sicherheitsstatus muss immer transparent kommuniziert werden.

Auch ist die gesamte CI-/CD-Pipeline als Qualitätsinstrument zu verstehen. Je höher der Automatisierungsgrad, desto geringer die Gefahr von Sicherheitsproblemen. Grundlage für die Einhaltung von Security-Vorgaben sind bereits gehärtete Images im Vorfeld, die regelmäßig nach Vorgaben erstellt werden. Außerdem ist es streng verboten, außerhalb der Pipeline Code bereitzustellen oder sogar Konfigurationen zu ändern. Auch das erfolgt über die Pipeline – somit ist alles dokumentiert und im Störungsvorfall rasch nachvollziehbar.

## Sicherheitsbewusstsein schaffen

Häufig sind es aber nicht einmal Programmierfehler, die zu Sicherheitsproblemen führen, sondern Unachtsamkeit und der laxer Umgang mit sensiblen Informationen. Unter hohem Zeitdruck werden Passwörter oder Schlüssel ungeschützt per E-Mail und Chat ausgetauscht, starke Verschlüsselung und Mehrfaktorenauthentifizierung



der Schnelligkeit geopfert, Zertifikate selbst signiert oder Cloud-Speicher falsch konfiguriert. Zahlreiche Datendiebstähle gehen auf solche Nachlässigkeiten zurück. So verlor beispielsweise die Firma Uber persönliche Daten von mehr als 600.000 Fahrern und 57 Millionen Fahrgästen, [weil Entwickler mit ihrem Code unabsichtlich auch ihre Login-Daten auf GitHub veröffentlicht hatten](#). Laut der Sicherheitsfirma Digital Shadows waren fast 1,5 Milliarden interne Firmendokumente mit zum Teil hochsensiblen Inhalten frei über das Internet zugänglich, [weil unter anderem Amazon-Cloud-Speicher \(S3 Buckets\) falsch konfiguriert wurde](#).

Cloud-Dienste zur Verwaltung von Schlüsseln wie AWS KMS und Azure Key Vault oder Code-Module wie git-secrets und detect-secrets, die eine Veröffentlichung sensibler Daten automatisiert verhindern, können das Problem zwar lindern, doch ohne entsprechendes Sicherheitsbewusstsein der Entwickler sind technische Maßnahmen nur bedingt wirksam. Security Awareness muss daher ein integraler Bestandteil einer jeden DevOps-Kultur werden. Das Sicherheitsbewusstsein lässt sich über externe Coaches entwickeln oder direkt im Team verankern, indem ein Mitarbeiter als „Security Manager“ oder „Security Champion“ fungiert. Wichtig ist hier auch der Reifegrad der einzelnen Mitglieder in der Organisation. Es gibt bereits Reifegradmodelle, die sehr gut aufzeigen, wie jeder Einzelne seinen Arbeitsalltag gestaltet. Der Wandel von Low Performing zu Elite lässt sich darüber sehr gut messen. Auch die Rollen, die in Zukunft für Unternehmen relevant sind, ändern sich.

Der aktuelle Sicherheitsstatus muss dabei immer transparent kommuniziert werden. So können beispielsweise Fehlermeldungen in der Pipeline, die auf eine instabile und damit prinzipiell kompromittierte Software hinweisen, über ein gemeinsames Dashboard allen Entwicklern zugänglich gemacht

werden. Auch die Ergebnisse von Schwachstellen-Scans sollten jederzeit für das gesamte Team einsehbar sein. Diese Offenheit darf allerdings nicht zu gegenseitigen Schuldzuweisungen führen, sondern sollte als Chance verstanden werden, Dinge besser und sicherer zu machen.

## Fazit

Wer nach DevOps-Prinzipien entwickelte Software über die Cloud bereitstellt, muss Sicherheit, Datenschutz und Compliance neu denken. In der alten Welt von Wasserfallmodell und On-Premise-Rechenzentrum spielten Verfügbarkeit und Skalierbarkeit die entscheidende Rolle, die Sicherheit wurde am Perimeter mit Firewalls, De-militarisierten Zonen (DMZ) und Intrusion-Detection-/Prevention-Systemen (IDS/IPS) verteidigt. In der Cloud gibt es ebenso wenig einen definierten Perimeter wie fixe Rechenleistung oder periodische Wartungsarbeiten. Durch die freie Verfügbarkeit von Rechenleistung zu jeder Zeit lassen sich Anpassungen an den bereitgestellten Produkten schneller durchführen, Fehler werden schneller beseitigt. Daher sind Tests auch für einzelne Releases nicht mehr so umfangreich beziehungsweise zum größten Teil automatisiert.

Sicherheit, Datenschutz und Compliance sind daher mit anderen Methoden wie Privacy by Design, starker Verschlüsselung oder Mehrfaktorenauthentifizierung zu garantieren, Security-Tests kontinuierlich in die Entwicklungspipeline zu integrieren und so weit wie möglich zu automatisieren. Der Wert solcher Maßnahmen steht und fällt allerdings mit dem Bewusstsein der Teams für deren Bedeutung. Werden unter Zeitdruck oder aus Nachlässigkeit Sicherheitsmechanismen umgangen, Compliance-Vorgaben ignoriert und sensible Daten öffentlich zugänglich gemacht, nützen auch die besten Konzepte und Tools nichts. ■

## Die Zukunft des IT-Betriebs

# Wie sich Teamstrukturen, Rollen, Skills und Prozesse verändern müssen

**Cloud Computing, Agilität, DevOps und Automatisierung verändern den IT-Betrieb in fundamentaler Weise. Das hat nicht nur Auswirkungen auf die Entwicklung von Software und die Bereitstellung von Ressourcen, sondern betrifft vor allem die Kultur im Unternehmen.**

Der traditionelle IT-Betrieb ist von der klaren Trennung zwischen IT-Bereitstellung und deren Nutzung geprägt. Die IT-Abteilung war federführend für die Konzeption, den Aufbau, den Betrieb und die Absicherung der IT-Ressourcen verantwortlich – und ist es in vielen Unternehmen auch heute noch. Dafür stand ihr nahezu das gesamte IT-Budget zur Verfügung. Fachabteilungen hatten nur wenig Mitspracherecht. Sie mussten benötigte Ressourcen in einem aufwendigen Bestellprozess beantragen, und die Beschaffung dauerte oft Wochen oder Monate. Dieses Modell gerät derzeit von verschiedenen Seiten unter Druck. Cloud Computing macht traditionelle Rechenzentren weitgehend überflüssig, und damit verschwinden auch die typischen Silo-Organisationen und deren Verantwortungsbereiche. Die im Data Center übliche Trennung in Netz-, Hardware, Firewall-, Datenbank- und Betriebs-Teams ist im Cloud-Umfeld wenig sinnvoll. Stattdessen gilt es, für bestehende Software, vor allem aber auch für neue Cloud-native Applikationen und Container-Umgebungen ein ganzheitliches Betriebsmodell bereitzustellen,

das eine sichere und effiziente Nutzung garantiert. Man kann sogar so weit gehen zu sagen, Datenschutz und Security sind die einzigen verbleibenden Gründe, nicht in die Cloud zu gehen. Sobald sie gelöst sind, werden alle Anwendungen entweder abgelöst oder in die Cloud migriert. Die Möglichkeiten der Cloud werden stetig mit großen Entwicklungs- und Betriebsteams weiterentwickelt – da kann keine heimische IT-Abteilung im Rechenzentrum mithalten. Wenn dazu „DevOps Cloud native“-Teams ihre Performance voll ausspielen können, dann sind Innovationen bei hoher Effizienz (fast) keine Grenzen gesetzt.

## Interdisziplinäre, cross-funktionale Teams

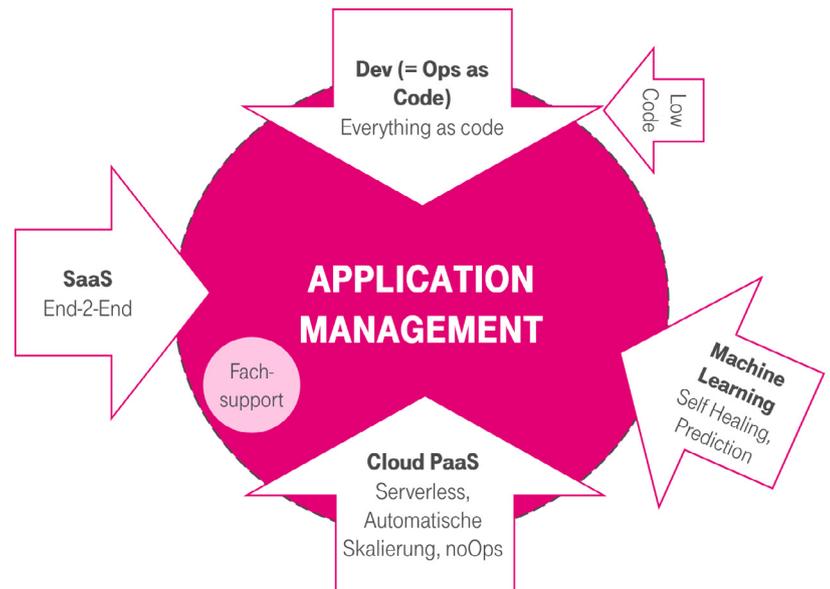
Der agile Ansatz, der sich aus der Softwareentwicklung immer mehr auch in andere Unternehmensbereiche ausbreitet, stellt ebenfalls das starre hierarchische Modell von IT-Bereitstellung und IT-Nutzung infrage. In agilen Organisationen wird die Trennung in spezialisierte Abteilungen und Kompetenzzentren wie Business Analytics, Entwicklung oder Qualitätssicherung immer unwichtiger. Stattdessen verantworten interdisziplinäre oder cross-funktionale Teams den gesamten Lebenszyklus eines Artefakts. Jedes Mitglied eines Teams ist für dessen wirtschaftlichen Erfolg mitverantwortlich und steuert nicht nur isoliertes Fachwissen oder

Spezialkenntnisse bei. Die wachsende Autonomie der Produktteams, die Lösungen nicht nur entwickeln, sondern auch betreiben, schmälert die Bedeutung der zentralen IT-Abteilung zusätzlich. Statt Anträge auszufüllen, beschaffen sie sich ihre Ressourcen selbst oder definieren sie sogar als Infrastructure as a Code direkt in ihrer Pipeline.

### Traditioneller IT-Betrieb verliert an Stellenwert

Das gilt allerdings noch nicht in vollem Umfang im Desktop-Bereich. Auch wenn Office 365 als Cloud-Modell hier schon einiges harmonisiert, obliegen Hardwarebeschaffung und Support nach wie vor der IT-Abteilung – es sei denn, die Hardwarebeschaffung wird im Sinne von Bring Your Own Device (BYOD) an die Mitarbeiter outsourct. In Zukunft ist die Aufgabe der IT nicht mehr nur die Bereitstellung von Diensten auf Request, sondern auch die proaktive Bereitstellung von Self-Service-Funktionen für die Fachabteilungen.

Die zunehmende Automatisierung des IT-Betriebs durch Cloud-Plattformen, DevOps und AIOps reduziert den Stellenwert des herkömmlichen IT-Betriebs zusätzlich. Robotic Process Automation entlastet Mitarbeiter von Routineaufgaben. Kognitive Monitoring-Lösungen erkennen und beheben Fehler selbstständig, bis hin zum nahezu autonomen „SmartOps“- oder „NoOps“-Betrieb. Auch traditionelle Frameworks wie die IT Infrastructure Library (ITIL) behalten in Cloud-Umgebungen ihre Bedeutung. So zählt etwa Forrester in seiner Studie „Top 10 Facts About Managed Public Cloud Services“ die Beherrschung von ITIL-Prozessen zu den Kernkompetenzen eines Managed-Service-Providers. In der



Der traditionelle IT-Betrieb gerät von verschiedenen Seiten unter Druck. (Quelle: T-Systems Multimedia Solutions)

aktuellen Version ITIL 4 trägt das Framework der Agile- und DevOps-Bewegung Rechnung, indem der ganzheitliche Ansatz zugunsten einer Value-Chain-Betrachtung aufgegeben wird.

Welche Auswirkungen diese Faktoren auf die Wettbewerbsfähigkeit haben, zeigt der „Accelerate State of DevOps Report 2019“. Er widmet sich insbesondere der Frage, was erfolgreiche Unternehmen von weniger erfolgreichen Betrieben unterscheidet. Der Studienhersteller DORA (DevOps Research and Assessment) hat dazu eine Matrix entwickelt, mit der sich die Leistung bei der Entwicklung und dem Betrieb von Software (Software Delivery and Operational Performance, SDO) messen lässt. Zu den Messkriterien gehören die Schnelligkeit der Softwareentwicklung, die Release-Frequenz, die Häufigkeit von Fehlern und die Zeit, bis diese behoben sind, sowie die allgemeine Verfügbarkeit der Lösungen. Die Unterschiede zwischen den

Unternehmen in der besten und der schlechtesten Kategorie sind enorm: Top-Performer können mehrmals am Tag ein neues Release ausliefern, benötigen für Änderungen weniger als einen Tag, beheben Fehler in unter einer Stunde und verzeichnen eine Fehlerrate von maximal 15 Prozent. Die Firmen, die am schlechtesten abschneiden, benötigen dagegen bis zu sechs Monate für ein neues Release und ebenso lange für Änderungen, brauchen zwischen einer Woche und einem Monat zur Behebung von Fehlern und verzeichnen eine Fehlerrate von bis zu 60 Prozent.

### Weitere Schlüsselergebnisse der Untersuchung

- Die Cloud ist ein entscheidender Differenzierungsfaktor. Erfolgreiche Unternehmen nutzen sehr viel häufiger alle fünf Cloud-Dimensionen, wie sie vom National Institute of Standards and Technology (NIST) definiert wurden (Selbstbedienung, plattformunabhängiger Zugang, Ressourcenteilung, Elastizität und kontinuierliche Qualitätssicherung).
- Erfolgreiche Unternehmen setzen weitaus stärker auf automatisierte Prozesse als weniger erfolgreiche Betriebe. Vor allem bei der Automatisierung von Build-Prozessen und Tests, aber auch bei der Bereitstellung und Integration der Software liegen die Top-Performer vorne. Die Testautomatisierung hat außerdem positive Effekte auf die kontinuierliche Integration von Komponenten (Continuous Integration) und die fortlaufende Bereitstellung von Anwendungen (Continuous Deployment).

- Eine modulare Softwarearchitektur mit geringen Abhängigkeiten verringert den Abstimmungsaufwand zwischen Entwicklerteams und erleichtert Testing und Deployment.
- Nur 40 Prozent der Befragten absolvieren wenigstens einmal im Jahr einen Disaster-Recovery-Test. Wenn Unternehmen solche Tests durchführen, steigt die Wahrscheinlichkeit einer besseren Serviceverfügbarkeit.
- Traditionelle Formen der Wissensvermittlung und -organisation wie Trainingszentren oder Centers of Excellence verlieren an Bedeutung. Erfolgreiche Unternehmen setzen deutlich mehr auf Arbeits- und Wissensgemeinschaften (Communities of Practice) und einen „Bottom-up“-Ansatz. Pilotprojekte scheitern in ihnen zudem deutlich seltener und werden besser ausgewertet.
- Ein gutes Change-Management mit kurzen Entscheidungswegen und schlanken Prozessen hat nicht nur positive Auswirkungen auf die Schnelligkeit und Stabilität in der Softwareentwicklung, es reduziert auch die Gefahr eines Burn-outs bei den Beschäftigten.

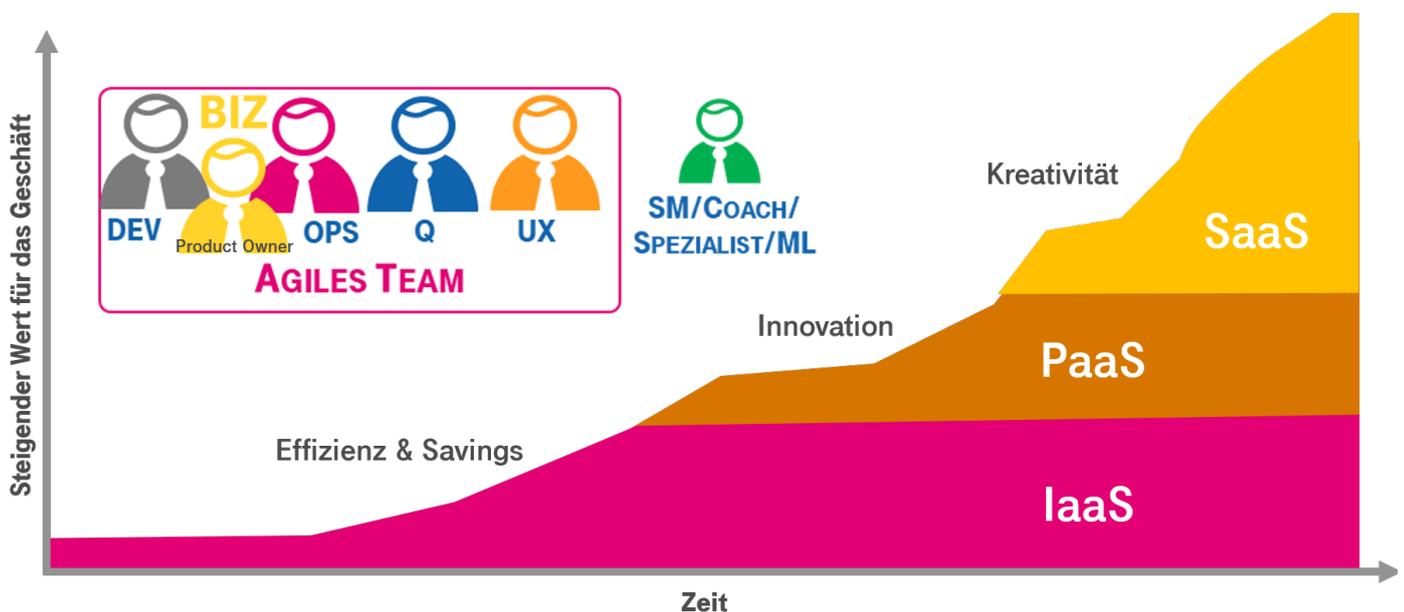
## Was jetzt zu tun ist

Unternehmen, die in der Transformation noch am Anfang stehen, sollten so schnell wie möglich DevOps-Prinzipien und agile Konzepte wie Scrum in ihre Prozesse integrieren. Laut einer Studie der Uni Koblenz setzen erst 40 Prozent agile Methoden in IT-nahen Themen ein. Für den Erfolg ist es wichtig, herkömmliche Macht- und Befehlsstrukturen zu durchbrechen. Der externe oder interne Kunde wird zum Product Owner, der eng mit dem Entwicklerteam zusammenarbeitet und die geschäftlichen Ziele definiert. Starre Abteilungen werden durch agile arbeitende interdisziplinäre Scrum-Teams ersetzt. Der Scrum Master darf dabei nicht als Ab-

teilungsleiter im herkömmlichen Sinn missverstanden werden. Er gibt keine Anweisungen, vielmehr ist es der Moderator und Controller, der für eine reibungslose Zusammenarbeit im Team sorgt.

Agile Strukturen stellen hohe Anforderungen an die Kommunikations- und Feedback-Kultur in einem Unternehmen. Während in herkömmlichen Strukturen wichtige Entscheidungen dem Management vorbehalten waren, können nun die Teams Entscheidungen treffen und Maßnahmen einleiten. Der Zugriff auf Cloud-Ressourcen ist ein gutes Beispiel dafür. Hier entscheidet das Team darüber, wie das Budget zu verwenden ist, um die Ziele zu erreichen, und verantwortet damit gemeinsam mit dem Product Owner die Kosten. Auch die Einstellung neuer Teamkollegen wird vom Team mitbestimmt und nicht mehr unbedingt allein durch die Führungskraft.

Der geschäftliche Mehrwert erhöht sich durch mehr Freiheit für Kreativität und Innovation im Team.  
(Quelle: T-Systems Multimedia Solutions)



Das bedeutet für traditionelle Führungskräfte natürlich ein starkes Umdenken und zum Teil auch einen Machtverlust. Doch Mitarbeiter müssen ebenfalls umdenken: DevOps bedeutet nicht Disziplinlosigkeit – im Gegenteil. Durch das Einschwören auf gemeinsame Ziele wird der Druck auf die einzelnen Teammitglieder eher größer als geringer. Auch das Anforderungsprofil für einen DevOps Engineer unterscheidet sich deutlich von dem eines klassischen Systems Engineers. Ein rein technisches Verständnis von Servern, Speichergeräten oder Netzwerk-Switches genügt nicht mehr, künftig werden alle Mitarbeiter einer DevOps-Einheit jenseits der Programmiersprachen auch Fach- und Domänenwissen beherrschen müssen. Ein annähernd gleiches kulturelles Mindset ist Voraussetzung für den Teamerfolg. Das muss vom Management unterstützt und gewollt sein.

„ Ein wichtiger Erfolgsfaktor ist Transparenz: Leistungen und Aufwände sind klar zu kommunizieren, Ergebnisse kontinuierlich zu messen und zu dokumentieren.

Ein weiterer wesentlicher Faktor für den Erfolg ist Transparenz. Leistungen und Aufwände müssen jederzeit klar kommuniziert, Ergebnisse kontinuierlich gemessen und dokumentiert werden. Neben bekannten agilen Tools wie Jira oder YouTrack ist dafür eine neue Art des Monitorings notwendig: Statt der bloßen Überwachung technischer Para-

meter sollte eine möglichst realistische Einschätzung des Produktivbetriebs das Ziel sein. Die gemessenen Kennwerte müssen dabei einen echten geschäftlichen Mehrwert liefern. Die Schwellwertauslastung einer CPU und ähnliche technische Metriken tun das nicht!

Schließlich verändert sich auch das Verhältnis zwischen internen oder externen Kunden und dem Team, das dessen Anforderungen umsetzt. Während im alten Wasserfallmodell Lasten- und Pflichtenheft die Arbeit vorgaben, befüllt heute der Product Owner das Backlog flexibel und in Absprache mit dem Team. Beide Seiten tragen Verantwortung für die Leistung und die Stabilität des Produkts. Entwickler können sich nicht mehr darauf berufen, dass sie lediglich den Vorgaben des Kunden gefolgt sind. Diese gemeinsame Verantwortung muss sich auch in den Vertragswerken zwischen Kunde und Dienstleister niederschlagen.

## Fazit

Märkte und Geschäftsmodelle verändern sich durch die Digitalisierung rasant. In dieser Welt kann nur bestehen, wer schnell auf Veränderungen reagieren und rasch neue Produkte auf den Markt bringen kann. Bei Qualität und Verfügbarkeit dürfen jedoch keine Kompromisse gemacht werden, sonst leiden Sicherheit und Kundenzufriedenheit. Für Unternehmen bedeutet das, sämtliche Prozesse und Strukturen auf den Prüfstand zu stellen, agile Methoden und DevOps-Prinzipien zu etablieren und Prozesse wo immer möglich zu automatisieren. Doch erst in Kombination mit der Cloud können diese Veränderungen ihre volle Wirkung entfalten. Im Budget werden Mittel frei für kreativere Aufgaben als das „Verwalten“ von IT-Ressourcen. Die Organisation kann sich so von einer Power-Organisation zu einer Performance-Organisation entwickeln. ■

# Wie DevOps, Cloud und Automatisierung die Zusammenarbeit in der IT verändern

Interview mit den DevOps- und Cloud-Experten Olaf Garves und Martin Lange

Die IT-Welt verändert sich. Dazu tragen vor allem die Themen Cloud, DevOps und Automatisierung bei. Doch wie arbeiten Teams jetzt erfolgreich zusammen? Wie erkennt eine Organisation ihren Reifegrad in dieser veränderten Situation? Welche neuen Rollen entwickeln sich? Dazu haben wir die Experten der T-Systems Multimedia Solutions GmbH befragt.

## Welche Erfolgsfaktoren sehen Sie in einer veränderten IT-Welt?

**Olaf Garves:** Die Kerngeschäftsprozesse eines Unternehmens, sind immer mehr mit der IT verknüpft. Das heißt, der Anspruch, dass die Technik im Hintergrund funktionieren muss, ist für die Leistungserbringung unbedingt zu erfüllen. Digitale Zuverlässigkeit muss gewährleistet sein und dazu gehört, Anwendungen sicher und stabil verfügbar zu machen und gleichzeitig die Möglichkeit zu bieten, schnell und kundenorientiert Funktionen und Features ändern zu können. Technische SLAs allein sind dabei heute nicht mehr zeitgemäß, um den Erfolg zu messen. Durch DevOps, Cloud und Automatisierung verändern sich die Anforderungen erheblich. Das agile Team konzentriert sich auf Unternehmensziele.

Um den businessrelevanten Output zu messen, sind folgende KPIs geeignet:

- Grad der Zielerreichung im Business Case (Marktanteile, Umsatz, Profit, Image, Kundengewinnung, gesellschaftliche Ziele/Werteerhaltung, Nachhaltigkeit)
- Zufriedenheit des Product Owners
- Zufriedenheit des Nutzers (Endkunde, eigene Mitarbeiter, Partner)
- Software-Durchsatz, Stabilität
- Team Happiness
- Feature-Anteil, Mean Time-to-Market (in Anlehnung an Mean Time To Repair, ein wichtiger KPI in der Instandhaltung)
- Velocity
- Qualität des Produkts in Bezug auf Skalierbarkeit, Performance, Stabilität, Verfügbarkeit und Robustheit
- Team-Reifegrad
- Reifegrad der Organisation

## Wie erkennt eine Organisation den Reifegrad Ihrer DevOps- & Cloud-Aktivitäten?

**Martin Lange:** Die Festlegung des Reifegrades kann in vier Dimensionen bestimmt werden: Kultur, Methoden, Tools und Architektur – diese werde ich kurz erläutern:

„Ich habe gar **nicht genug Mitarbeiter**, um wöchentliche Releases qualitätsgerecht zu liefern!“

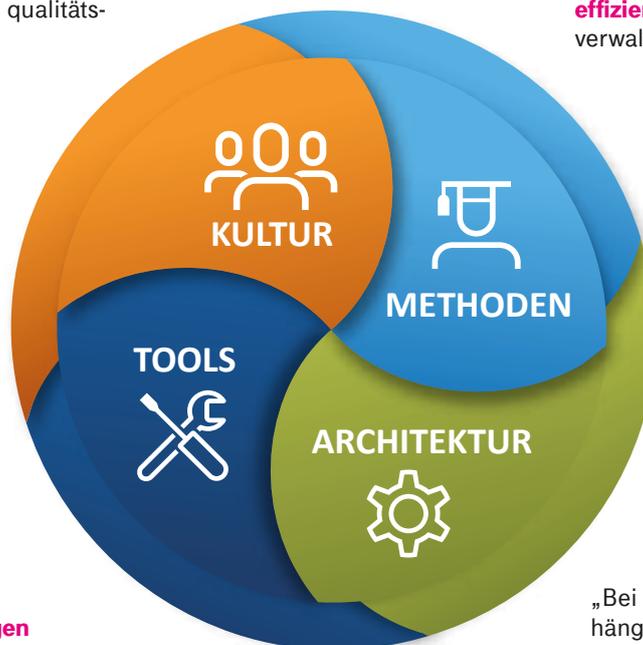
„Wie kann ich meine **Infrastruktur effizient und flexibel** aufbauen und verwalten?“

„Die Ziele von DevOps sind doch jedem klar. **Was muss da noch erklärt werden?**“

„Wie kann ich die **Software-Installationsaufwände** signifikant reduzieren?“

„Wie bekomme ich **heute entwickelte Features morgen in die Produktion?**“

„Bei DevOps geht es um Tools. Der Erfolg hängt in erster Linie von der **richtigen Toolauswahl** ab ...“





### Kultur:

Die Organisationskultur kann von Silo-orientierter Organisation bis hin zur Sharing Culture gehen. Als Beispiel der klassische IT-Architekt: Er rettet jedes Projekt, egal in welcher Schieflage es sich befindet. Das ist sehr gut, führt auf Dauer dennoch zu Wissens-Silos. Wenn der Kollege nicht mehr Teil der Organisation ist, gibt es einige harte Nüsse zu knacken oder die Wissens-Domain verwaist komplett.

Tipp: Ziel einer Organisation sollte eine Sharing Culture sein. Das Teilen von Wissen ist das höchste „Gut“. Voraussetzung hierfür ist, dass die richtigen Methoden, Tools und Architekturen vorhanden sind.



### Methoden:

DevOps als Methode ist in vielen Organisationen bereits an der Tagesordnung. Aber wie kann ich herausfinden, ob Teams bspw. in SCRUM effizient arbeiten? Agiles Zusammenarbeiten ist nicht für jeden gleich vom ersten Tag an normal. Es gibt einige Schritte, die getan werden müssen, damit es gut läuft. Z. B. werden anfangs StandUps und Retrospektiven eingeführt. Trotzdem wird es noch eine RACI-Matrix geben, die den Teams dabei hilft, die neue Verantwortung greifen zu können.

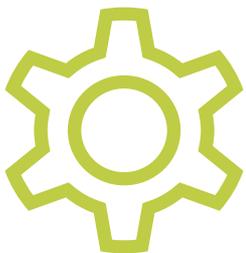
Tipp: Schritt für Schritt werden Teams und Organisationen in der Lage sein, die neuen Methoden vollumfänglich zu nutzen und damit auch ihre Effektivität zu steigern. Außerdem ist die eigene Organisationsstruktur wichtig. Ob Lean Management oder Spotify, finden Sie Ihr geeignetes Modell.



### Tools:

Der Wandel in der Organisation durch den richtigen Einsatz von Kollaborationswerkzeugen ist nicht zu unterschätzen. Nehmen wir O365 mit OneDrive als Beispiel. Durch OneDrive ist das gemeinsame Arbeiten an Dokumenten ein Kinderspiel. Lokale Kopien werden nicht mehr benötigt. Das Gleiche gilt für zentrale Repositories wie bspw. Git. Diese Werkzeuge ermöglichen es jederzeit Wissen zu teilen und alle Informationen in der Organisation zur Verfügung zu stellen.

Tipp: Erstellen Sie eine gemeinsame kulturelle Basis. Mit dieser wird ein neues Tool nicht nur ein weiteres Stück Geschichte in der Organisation sein, sondern gelebte Realität.



### Architektur:

Gerade durch Cloud und Hyperautomatisierung ändern sich die Regeln für die IT-Abteilungen in der Organisation. Hier wird in Zukunft mehr erwartet als das Bereitstellen von Diensten auf Basis von Requests. Fachabteilungen erwarten mehr Self-Services und schnellere Reaktionen auf ihre Anforderungen, um mit den aktuellen Markttrends mithalten zu können. Die CI/CD-Pipeline ist der direkte Weg in die Cloud. Alle Architekturen sollten sich danach richten, damit alle Komponenten automatisiert und sicher deployt werden.

Tipp: Ermöglichen Sie Ihrem Team wegweisende CI/CD-Blueprints zu erstellen und vernetzen Sie es innerhalb einer aktiven Community. Erarbeiten Sie Angebote und bieten Sie Self-Service-Optionen an, ohne die Kreativität der Einzelnen einzuschränken. Nehmen Sie ständig Feedback auf und setzen Sie dieses als wiederverwertbaren Standard um. Erlauben Sie Anpassungen und machen Sie diese anschließend transparent.

## Können Sie uns das Wichtigste noch einmal zusammenfassen?

**Martin Lange:** Gern. Der Reifegrad einer Organisation lässt sich durch unterschiedliche Faktoren bestimmen. Es gibt keine Organisation, die in allen Bereichen 100% erreichen kann. Hier sind viele kleine Schritte in der Organisation durchzuführen, um sich von einer Power Organisation zu einer Performance Organisation zu entwickeln. In dieser Grafik sieht man, was die Unterschiede sind:

	POWER-ORIENTIERT	REGELORIENTIERT	PERFORMANCE-ORIENTIERT
	auf Anfrage folgt Reaktion		von sich aus denkende und handelnde Organisation
<b>Zusammenarbeit</b>	Kooperation nur da, wo es zwingend notwendig ist.	Kooperation wird akzeptiert.	Kooperation wird aktiv gelebt.
<b>Verteilen von Informationen</b>	Nachrichten werden ohne Zusammenhang verteilt.	Informationen werden vernachlässigt, Regeln zählen.	Es gibt spezielle Trainings für die Verteilung von Wissen, um einen größtmöglichen Gewinn durch die Nachricht zu erzielen.
<b>Wahrnehmen von Verantwortlichkeiten</b>	Verantwortung wird nicht wahrgenommen, da es dafür keinen Grund gibt.	Verantwortung wird in der definierten Rolle wahrgenommen.	Verantwortung und Risiken werden geteilt. Erfolge gemeinsam gefeiert.
<b>Gegenseitige Unterstützung</b>	Nahezu ausgeschlossen.	Wird im Sinne der Rolle toleriert.	Unterstützung wird aktiv in der Organisation gelebt. Wer hilft, ist Teil des Teams.
<b>Fehlerkultur</b>	Für Fehler wird immer ein Sündenbock gesucht.	Fehler führen zum Ausschluss aus der Organisation bzw. zu Sanktionen.	Fehler sind im Unternehmen erwünscht. Das Teilen der Erfahrungen gehört zum Alltag.
<b>Umgang mit Veränderungen</b>	Veränderungen werden in der Organisation nicht umgesetzt.	Veränderungen führen zu Problemen in der Organisation.	Veränderungen werden als normal angesehen und in der Organisation umgesetzt.

## Ohne eine Strategie ist das doch gar nicht umsetzbar, oder?

**Martin Lange:** Das ist richtig. Eine klare Cloud-/DevOps-Strategie muss in diesem Zuge auf jeden Fall aufgesetzt und verfolgt werden. Hier helfen die vier Dimensionen von DevOps, um eine Reifegradanalyse zur Bestandsaufnahme für die eigene Organisation durchzuführen. Auf Basis der Ergebnisse können dann die zur Strategie passenden Schritte abgeleitet werden. Doch auch die Cloud löst nicht alle Probleme automatisch. Skripte/ oder PaaS-bezogene Konzepte zur Automatisierung aus dem herkömmlichen Rechenzentrum müssen zu großen Teilen neu geschrieben werden.

[Wir helfen Ihnen gern bei der Ermittlung Ihres DevOps-Reifegrades: Fragen Sie uns direkt an!](#)

## In Zeiten von Fachkräftemangel: Welche Rollen werden jetzt unverzichtbar?

**Olaf Garves:** System- und Security-Engineers zum Beispiel können zum DevOps-Engineer entwickelt werden. Dieser übernimmt alle Themen von der Automatisierung, Anwendungswartung, über Test und Build bis hin zu CI/CD-Anpassungen und Software-Stack-Aufbau und -Migration. Kurzum, er ist dafür verantwortlich, dass jederzeit deployt werden kann. Inwieweit andere Rollen sich ändern, haben wir in dieser Übersicht zusammengefasst:

	Product Owner	Scrum Master	Entwickler	Software-/ System-/ Cloud-Architekt	Security-Engineer	DevOps-Engineer	System-Engineer
<b>Rollen abnehmend</b>					✓		✓
<b>Rollen unverändert</b>	✓	✓	✓	✓			
<b>Rollen zunehmend</b>						✓	
<b>Tätigkeiten abnehmend</b>	Projektleitung	Teamführung	Entwicklung von Null an		Verantwortung für Security und Datenschutz		Operation, AM, Support, Analyse, Proaktive Maßnahmen, Monitoring, (Data Science), Infrastrukturaufbau
<b>Tätigkeiten zunehmend</b>	Produktnutzen-Vermittler, Management-Ansprechpartner Fachseiten-Schnittstelle	Teamförderung, Reibungsverluste mindern	Anwendung Orchestrierung, Konfiguration, Testautomatisierung, Code Refactoring	Analysieren, Evaluieren, SW-Modernisierung, Architektur, Capacity Management, Cloud-Infrastruktur, Service Design, DR, Skalierung, Container, Sec & Datenschutz	Berater und Coach; DevSecOps-Team übernimmt Verantwortung	Automatisierung, Anwendungswartung, Test, Build, CI/CD Anpassung, SW-Stack Aufbau/ Migration inkl. DB, jederzeit Deployment	
<b>Softskills</b>	■■■■	■■■■■	■■■■	■■■■	■	■■■■	■■■■
<b>Technik-Skills</b>	■	■	■■■■■	■■■■■	■■■■	■■■■■	■■■■■
<b>Business-Alignment</b>	■■■■■	■■■	■■■	■■■■	■■■■	■■■■	■
<b>Fähigkeiten/Qualifikationen</b>	■■■■■	■■■■	■■■	■■■■	■■■■	■■■	■■■■
<b>Erfahrung Jahre</b>	■■■	■■■	■■■	■■■■■	■■■■	■■■	■■■
<b>Entwicklungspotenzial</b>	Product Owner	Scrum Master/ Product Owner	DevOps-Engineer/ Architekt/CI/CD/ Berater	SW/System/ Cloud-Architekt	Security Consulting/ Security Coach	DevOps-Engineer/ Entwickler/ Architekt/ technischer Berater	DevOps-Engineer

## Welche Qualifikationen müssen Mitarbeiter für diese neuen Rollen mitbringen?

**Olaf Garves:** Digitale Zuverlässigkeit, die der Kunde fordert, ist nicht nur ein technisches Feature. Anforderung sind mit Geschäftsverständnis zu bewerten und dann entsprechend umzusetzen. Es ist also nicht mehr nur von Bedeutung, dass IT-Mitarbeiter die richtigen Qualifikationen und Technik-Skills mitbringen. Der Engineer kann sich z.B. nicht mehr hinter der Technik verstecken. Vielmehr müssen IT-Mitarbeiter zukünftig vor allem Werte einer offenen Kultur mit ins Team einbringen, sie müssen interagieren und kommunizieren. Skills, die jetzt wichtig werden, sind Wissensaustausch, Zusammenarbeit, agile Methoden, Selbstorganisation, gemeinsame E2E-Verantwortung, Transparenz, Innovationsbereitschaft und Identifikation.

## Unsere Autoren



**Martin Lange**  
*Leiter Cloud Transformation*

Martin Lange ist als Teamleiter „Cloud Transformation“ verantwortlich für die strategische Weiterentwicklung des Cloud-Geschäfts der T-Systems Multimedia Solutions Kunden. Bereits seit 2005 ist er im Konzern der Deutschen Telekom in verschiedenen Rollen tätig. Schon während des Studiums arbeitete er als Software-Entwickler und -Architekt. Nach mehreren Jahren als technischer Berater und Projektleiter im Bereich Cloud Computing war Martin Lange von 2015 bis 2018 Produktmanager für ein zentrales IaaS Public Cloud-Angebot bei der T-Systems International GmbH.



**Olaf Garves**  
*Leiter Application Management Telco & Mobility Solutions*

Olaf Garves leitet das Servicefeld „Application Management Telco und Mobility Solutions“ bei T-Systems Multimedia Solutions. 1996 begann er als Projektleiter im iTV und Bildtelefonie Umfeld und führte anschließend auch mehrere Jahre eine SW-Entwicklungseinheit. 2003 setzte er den Grundstein für eine Support-Abteilung, die erfolgreich Web-Lösungen für verschiedenste Kunden betreut. Seit 2013 beschäftigt er sich mit DevOps und hat seinen Bereich entsprechend agiler Anforderungen organisatorisch und prozessual weiterentwickelt. Dabei steht die Mitarbeiterförderung im Fokus.

# Case Studies: DevOps-Teams im Interview

Jahrelange Kundenerfahrung zeigt, wie DevOps-Teams erfolgreich zusammenarbeiten.

Es gibt nicht nur ein DevOps – jeder braucht sein individuelles DevOps. Dies zeigen auch unsere Kundenbeispiele aus verschiedenen Branchen. In diesen Interviews lesen Sie, wie die DevOps-Teams zusammenarbeiten, welche Herausforderungen es zu meistern gilt und wie Innovationen vorangetrieben werden.

## DevOps-Team im Interview

Deutscher Automobilhersteller

### Wie definieren Sie DevOps?

**Wenke Hübler:** DevOps bedeutet für uns zum einen Zusammenarbeit über jegliche Bereichsgrenzen hinweg, mit dem Fokus das Projektziel des Kunden als Team gemeinsam zu erreichen. Zum anderen sollte ein ausgereiftes vollautomatisches Deployment und eine automatisierte Qualitätssicherung selbstverständlich sein, um Anforderungen wie beispielsweise schnelles Time-to-Market gerecht zu werden. Meines Erachtens sind Transparenz über die Geschehnisse im Projekt, vor allem über die, wo es Probleme, Hindernisse, Widerstand gibt, ein wichtiger Baustein für den Erfolg. Dies betrifft natürlich auch Projektziele und das vorhandene Projektbudget. Anderenfalls sind die Wege bis zur Problemlösung deutlich verlängert und die Reaktionsgeschwindigkeit des Teams somit eingeschränkt. Außerdem erfordert DevOps neben dem Willen über den „Tellerrand“ hinauszudenken, die Fähigkeit den Gesamtkontext der Anwendung und ihren Lebenszyklus zu verstehen. Ich glaube, dass am Ende auch immer der persönliche Anspruch dazugehört, die Projekte wirklich Ende-zu-Ende zu denken.

### Wie leben Sie DevOps konkret im Projekt?

**Thomas Friedland:** Das Projektteam arbeitet in dreiwöchigen Sprints, um schnell auf Anforderungen reagieren zu können. Über das gemeinsam genutzte Tool RocketChat ist ein schnelles Debugging jederzeit möglich, wodurch Laufzeiten von Fehlertickets kurzgehalten werden können. Wir haben ein standardisiertes Deployment der Anwendungen, damit die entwickelnden Abteilungen Einblick in die Deployment-Prozesse sowie die Logfiles zur Laufzeit bekommen. Das Deployment war nicht von Beginn an beim Projekt vorhanden, sondern wir haben zügig unsere Ideen in den regelmäßig stattfindenden Innovationsworkshops mit dem Unternehmen platziert, immer in Bezug auf den Kundennutzen. Neben dem Content Management System Adobe Experience Manager stellen wir auch MicroServices in der AWS Cloud bereit und setzen Testautomatisierung um.

### Woran merken Sie, dass Ihre Innovationen beim Kunden ankommen?

**Thomas Friedland:** Vor allem der Aufbau in der AWS Cloud hat beim Kunden einen guten Ruf. Immer mehr Fachbereiche fragen bei uns an, weitere Microservices in der Cloud in Betrieb zu nehmen. Aktuell sind wir dabei, das komplette CMS in die Cloud zu heben. Davon verspricht sich der Kunde mehr Flexibilität der Infrastruktur im Vergleich zum bisherigen dedicated hosting. Auch werden wir beim Kunden in vielen Querschnittsthemen hinzugezogen, was wir als Indikator für die gute Zusammenarbeit nehmen.

### Welche Schlüsse haben Sie aus dem Projekt auch im Hinblick auf den DevOps-Gedanken gezogen?

**Wenke Hübler:** Das Commitment auf der Führungsebene hat sich unheimlich positiv auf das Projekt und das Team ausgewirkt. Es war nicht „dein“ oder „mein“ Projekt, sondern es war immer „unser“ Projekt. So stand der Kunde im Mittelpunkt und es gab keine internen Nebenkriegsschauplätze. Es ist wichtig, miteinander über Hürden, Schwierigkeiten, mangelnde Ressourcen, Budgetfragen zu sprechen, ehrlich zu sein, Erwartungen zu äußern und auch um Hilfe zu bitten.



Wenke Hübler  
Head of Adobe Driven Technology



Thomas Friedland  
Senior System Architect

## DevOps-Team im Interview

Transport-/Logistikunternehmen

### Wie leben Sie DevOps konkret im Projekt?

**Lucas Fiedler:** Die besondere Herausforderung war, neben dem Commitment aller, also auch der Management-Sponsoren, ein Arbeitsmodell umzusetzen, welches auch unternehmensübergreifend funktioniert. Aus unserer Unternehmenssicht machen wir kein echtes DevOps, weil die Fachanwendungen durch verschiedene externe Dienstleister des Kunden erstellt werden. Wenn dieses durch alle gewünscht wird und ein Zielbild existiert, dann geht das Zusammenwachsen auf Augenhöhe schneller als bei Teams, welche zur Zusammenarbeit gezwungen werden. Ein motivierender Faktor ist dabei natürlich die Intention, Kundenziele immer besser und schneller zu erreichen, aber auch Neues zu lernen und sich weiterzuentwickeln. Damit wächst das Vertrauen und das befördert die Effektivität der Zusammenarbeit, was wiederum die Ergebnisse verbessert. Eine Besonderheit unseres DevOps-Teams ist, dass „Ops“ die Verantwortung für die Entwicklung der CI/CD-Pipeline hat. Das ist deshalb sinnvoll, da „Dev“ die Fachanwendung entwickeln soll und der Betrieb nach wie vor für den sicheren 24/7 Betrieb der Anwendung zuständig ist. Das Team muss sicherstellen, dass nur geprüfte SW in die Produktion ausgerollt wird.

### Welche Herausforderungen gab es im Cloud-Umfeld zu meistern?

**Martin Schern:** Bei der Einführung der Public Cloud war vor allem auf die Prozesse des Kunden zu achten. Diese zum Teil auch neu erstellten Prozesse mussten in Einklang mit den Compliance-Richtlinien des Unternehmens gebracht werden. Echte Agilität ist nur möglich, wenn die Prozesse und Verantwortlichkeiten geklärt sind. Um einen unkontrollierten Einsatz der Public Cloud-Lösung und Wildwuchs zu vermeiden, ist strategisches Vorgehen elementar. Die Möglichkeiten und Herausforderungen, die das Team im Bereich Security und IT-Governance lösen mussten, haben wir in diesem Webinar erläutert: [Cloud und Compliance unter einem Hut](#).

### Woran merken Sie, dass Ihre Innovationen beim Kunden ankommen?

**Martin Schern:** An dem begeisterten Gesichtsausdruck des Kunden, wenn auch die neuen Technologien und Möglichkeiten der Cloud umgesetzt werden können. Anwendung per lift & shift in die Cloud zu bringen, das kann prinzipiell jeder. Die eigentlichen Innovationen besteht darin, wenn die neuen Möglichkeiten der Cloud, wie z.B. PaaS und SaaS-Dienste eng mit der Anwendung verwoben werden. Hierzu gehören auch Anwendungsbeispiele von Machine Learning und Analytic Services. Auf Basis der in die Cloud migrierten Daten können hier zum Teil vollkommen neue Erkenntnisse über das Verhalten der Applikationen und Nutzer gewonnen werden. Diese Erkenntnisse können dann gezielt für die kontinuierliche Verbesserung genutzt werden und führen nicht selten zu ganz neuen Marktmöglichkeiten.

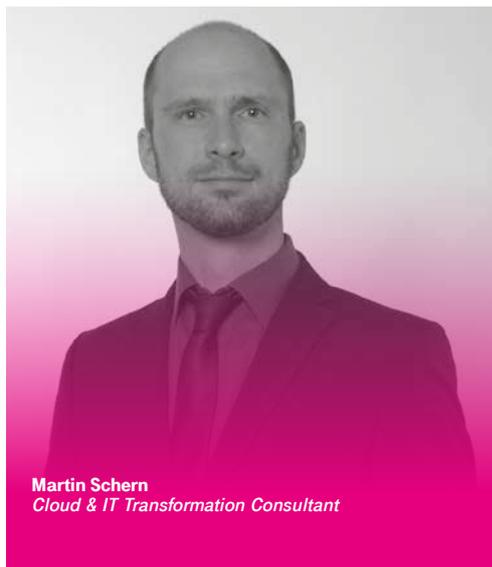
### Welche Schlüsse haben Sie aus dem Projekt auch im Hinblick auf den DevOps-Gedanken gezogen?

**Lucas Fiedler:** Wir haben uns die Frage gestellt, wie man die durch DevOps vorgelebte bereichsübergreifende Zusammenarbeit auf Querschnittsbereiche erweitern kann. Denn, was nützen mir die minutenschnellen Deployments, wenn ich einen Freischaltungsprozess habe, der händisch prozessiert werden muss? Oder warum kann ich nicht die Vertragsabteilung in die agile Zusammenarbeit miteinbeziehen, um frühzeitig Vertrauen auf allen Seiten aufzubauen? Unsere Gedanken gingen sogar noch weiter: Unternehmen sollten sich nicht ausschließlich als Konkurrenz sehen und so täglich Aufwände in Kundenbindung und Fachkräftesuche investieren. Stattdessen könnte der Standort Deutschland gemeinsam gestärkt werden und die Umsetzung bzw. digitale Unterstützung von industriellen Produktideen und Bürgerservices vorangetrieben werden. Natürlich ist Konkurrenz gut, ebenso, dass es verschiedene Clouds gibt. Ziel bei allen Aktivitäten sollte aber sein: Wie findet man schnell eine partnerschaftlich, individuell erstellte Lösung für den Kunden. Mit reiner Infrastruktur und dessen Betrieb wird kein Geld mehr verdient werden, sondern letztendes mit hochmotivierten MitarbeiterInnen, die gemeinsam gerne miteinander arbeiten. Und dies innerhalb von einem vielseitigen Cloud-Ökosystem, welche alle Technik-Möglichkeiten bietet, die man sich nur wünschen kann.

[Wir unterstützen Sie dabei, Ihre Unternehmensprozesse bedarfsgerecht auf die Cloud vorzubereiten und etablieren diese. Fragen Sie uns gleich an.](#)



Lucas Fiedler  
DevOps & Cloud Consultant



Martin Schern  
Cloud & IT Transformation Consultant

## DevOps-Team im Interview

Deutsche Krankenkasse

### Wie definieren Sie DevOps im Team mit dem Kunden?

**Ronny Lehmann:** Für uns bedeutet DevOps interdisziplinäre Zusammenarbeit aller Fachbereiche. Kunden, Entwicklung, Betrieb, Test und Redaktion agieren als EIN Team mit einem gemeinsamen Ziel. Dafür haben wir im Laufe der Jahre, die wir nun schon diese Plattform betreuen, bereichsübergreifende, verbindlich gelebte Prozesse etabliert. Eine typische Silo-bildung haben wir nicht mehr, weil unsere regelmäßigen Abstimmungen dies von vornherein verhindern und die direkte Kommunikation unter den Mitarbeitern jederzeit und problemlos funktioniert.

### Wie leben Sie DevOps konkret im Projekt?

**Raik Schmole:** Neben den gemeinsamen Meetings zur Abstimmung gehören auch prozessunterstützende Tools wie JIRA oder Confluence unbedingt dazu. Dann haben wir zum Beispiel die gesamte Entwicklungsumgebung in das Konfigurationsmanagement aufgenommen und die Deployment-Prozesse automatisiert. Dafür mussten wir gemeinsame Tools für Konfiguration und Automatisierung festlegen. Durch die intensive Zusammenarbeit hat sich inzwischen ein sehr großes fachbereichsübergreifendes Wissen und Problemverständnis aufgebaut. So versteht beispielsweise die Entwicklung die betrieblichen Notwendigkeiten und Anforderungen und kann diese bereits von Anfang an berücksichtigen.

### Was sind aus Ihrer Sicht notwendige Rahmenbedingungen für erfolgreiches DevOps?

**Ronny Lehmann:** Das Wichtigste bei DevOps ist das enge Vertrauensverhältnis zwischen unserem Kunden und uns als Dienstleister. Weiterhin war es aus meiner Sicht wichtig, dass alle Beteiligten bei dem Programm - auch jene auf Kunden-seite - die Vorteile von DevOps verstanden haben und dasselbe Ziel verfolgen. Das gesamte Team muss mithelfen, dass Konventionen aufgebrochen werden. Die Mitarbeiter und Teilprojektleiter gehören unterschiedlichen Profitcentern an und Teilteams haben teilweise auch unterschiedliche Verträge und Kundenansprechpartner. Das darf keine Barriere darstellen. Deswegen ist bei uns die Kundenseite in den DevOps-Abstimmungen miteinbezogen. Der DevOps-Gedanke muss also auch von den Führungskräften gefördert und gefordert werden. Außerdem braucht es abgestimmte Prozesse, um die zunehmende Automatisierung zu beherrschen.

### Woran merken Sie, dass Ihre Innovationen beim Kunden ankommen?

**Ronny Lehmann:** Die Krankenkasse hat ihr Angebot an digitalen Services in der Onlinegeschäftsstelle rasant weiterentwickelt und ausgebaut. Das wäre ohne engere Abstimmungszyklen und Automatisierung gar nicht möglich. Wir sichern so die höhere Flexibilität und Agilität, und können gleichzeitig Stabilität und Sicherheit gewährleisten. Insgesamt haben wir über die Jahre durch unsere gut geplanten DevOps-Maßnahmen gegenüber der Krankenkasse nur eine geringfügige Kostensteigerung. Die wesentlich höheren Release-Zyklen machen es möglich, dem Kunden schneller neue Geschäftsprozesse und Applikationen bereitzustellen.

### Können Sie ein Resümee zu dem Projekt bisher mit einem Ausblick auf den DevOps-Gedanken in der Zukunft ziehen?

**Raik Schmole:** Als nächstes wird der Ausbau der Testautomatisierung mit den dazugehörigen Prozessen den Schwerpunkt bilden. Das wollen wir durch ein weiterführendes Zusammenwachsen der jetzigen Teilteams zu einem einzelnen homogenen Team erreichen. Längerfristig planen wir mit dem Kunden, die Release-Zyklen ganz aufzubrechen und Adhoc-Micro-changes zu ermöglichen. Dazu wird aber auf der organisatorischen Seite, zum Beispiel bei den fachlichen Freigabe-Prozessen, noch einiges zu tun sein.



Ronny Lehmann  
Service Manager



Raik Schmole  
System Engineer

# Über T-Systems Multimedia Solutions

Die T-Systems Multimedia Solutions begleitet Großkonzerne und mittelständische Unternehmen bei der digitalen Transformation. Der Marktführer mit einem Jahresumsatz von 176 Millionen Euro im Jahr 2019 zeigt mit seiner Beratungs- und Technikkompetenz neue Wege und Geschäftsmodelle in den Bereichen Industrial IoT, Customer Experience, New Work sowie Digitale Zuverlässigkeit auf. Mit rund 2100 Mitarbeitern an sieben Standorten bietet der Digitaldienstleister ein dynamisches Web- und Application-Management und sorgt mit dem ersten zertifizierten Prüflabor der Internet- und Multimediabranche für höchste Softwarequalität, Barrierefreiheit und IT-Sicherheit.

Ausgezeichnet wurde T-Systems Multimedia Solutions mehrfach mit dem Social Business Leader Award der Experton Group sowie dem iF Design Award und gehörte 2017 zu den Gewinnern des Outstanding Security Performance Awards. Zudem wurde das Unternehmen mit Hauptsitz in Dresden mehrmals als einer von Deutschlands besten Arbeitgebern mit dem Great Place to Work Award gekürt sowie als bester Berater 2020 vom Wirtschaftsmagazin brand eins ausgezeichnet.

[Weitere Informationen finden Sie hier.](#)