

# Site Reliability Engineering

Applikationen in einer agilen Business-Welt zuverlässig betreiben

**T Systems**

Let's power  
higher performance



# Inhalt

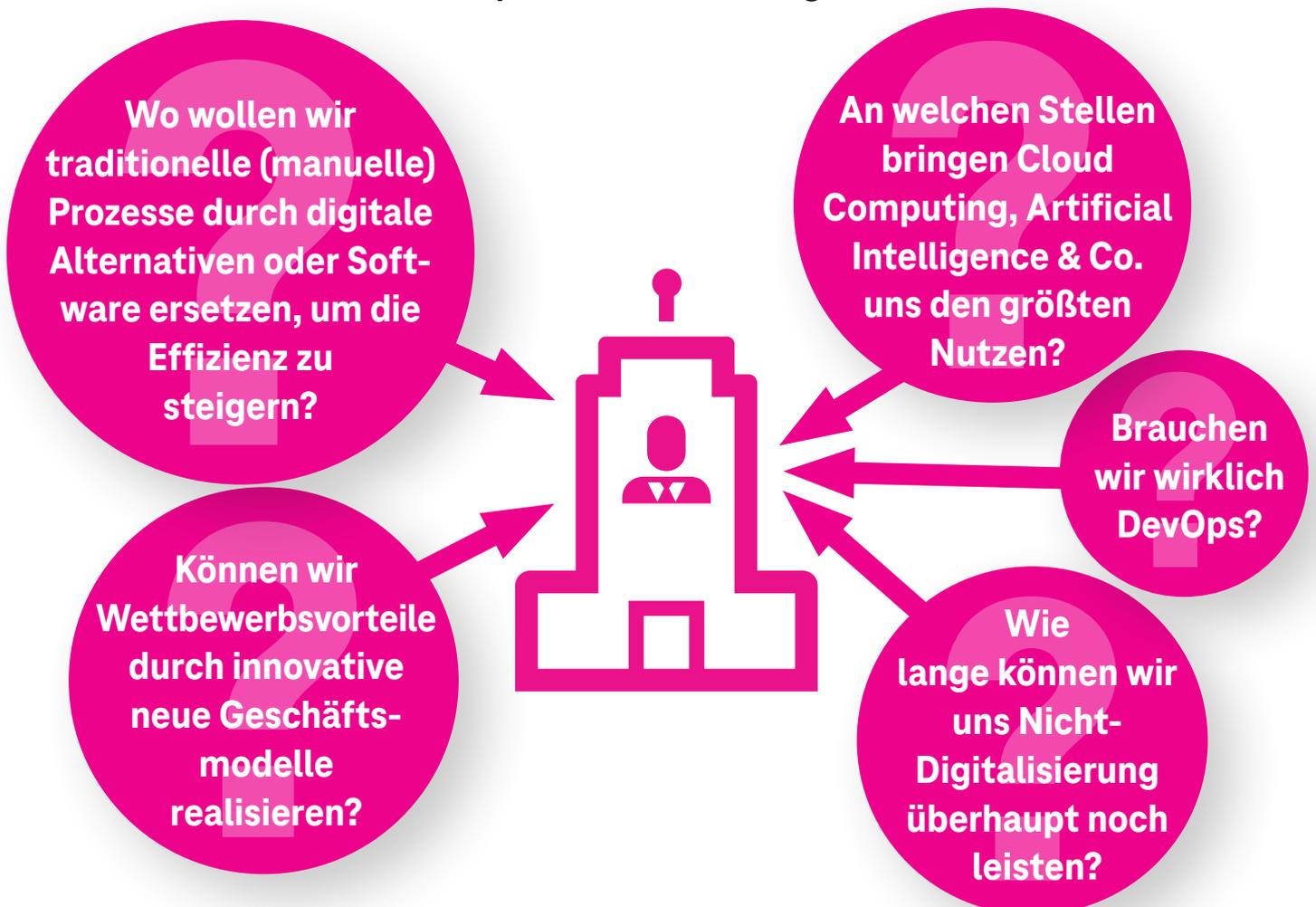
<b>Business-Trigger, Definition, Bedeutung DevOps für das Business</b>	<b>3</b>
<b>Status DevOps</b>	<b>7</b>
<b>Site Reliability Engineering</b>	<b>10</b>
<b>SRE-Werkzeuge, Service Level Agreements</b>	<b>13</b>
<b>Service Level Objectives</b>	<b>14</b>
<b>Service Level Indicators</b>	<b>15</b>
<b>Vorteile von SRE</b>	<b>16</b>
<b>Implementierung von DevOps/SRE</b>	<b>17</b>
<b>Einsatzbeispiel: Business-kritische Anwendungen der Deutschen Telekom</b>	<b>18</b>
<b>Fazit</b>	<b>19</b>
<b>Kontakt/Herausgeber</b>	<b>20</b>

”

**Jedes Unternehmen ist heute ein digitales Unternehmen.**

**- Forrester, 2021 -**

Auch wenn die Aussage vermutlich nicht tatsächlich für alle Unternehmen gilt – so lässt sich doch festhalten: Kaum ein Unternehmen einer gewissen Größe kommt heute noch ohne eine digitale Strategie aus. Dazu gehört eine Vielzahl von Komponenten und Fragen:





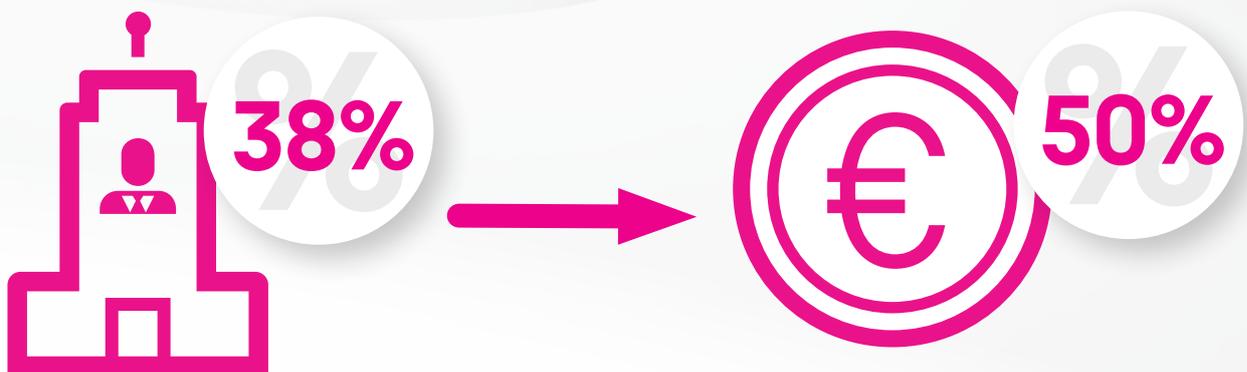
Schon heute ist offenbar: Größe und Industrieerfahrung sind nicht mehr die alleinigen Erfolgsfaktoren. Viel mehr sind in den letzten Jahren **Geschwindigkeit und Agilität** zu den neuen Business-Maximen avanciert. Unternehmen zeichnet heute die Fähigkeit aus, schnell auf Marktentwicklungen zu reagieren und in kürzeren Zyklen neue Services anzubieten. Schnelle Unternehmen sind erfolgreicher und hängen langsame ab. Dabei spielen die Digitalisierung und digitale Kompetenzen eine entscheidende Rolle.



Immer noch dreht sich in diesem Umfeld die Diskussion hauptsächlich um die Cloud. Doch tatsächlich ist die **Cloud nur der** (wenngleich essenzielle) **Startpunkt der Digitalisierung**. Die Mehrzahl der Unternehmen hat sich bereits für die Cloud entschieden und verfolgt Cloud-First-Strategien. Und das gilt nicht nur für Unternehmen: So steht Cloud-First beispielsweise auch im Mittelpunkt der Cloud-Strategie der Europäischen Kommission (veröffentlicht 2019). Sie wird dort als Wegbereiter für die übergreifende Digitalstrategie der Europäischen Kommission eingestuft.



Viele Industrieunternehmen streben eine dramatische Umstellung auf digitale Software und Dienstleistungen bis 2023 an.



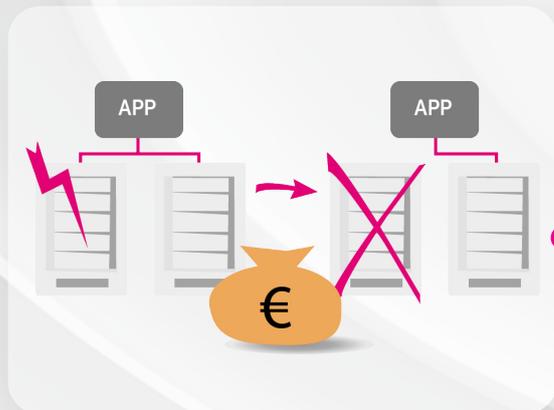
38% der Industrieunternehmen werden danach streben 50% der Einnahmen bis 2023 aus digitalen Technologien und Dienstleistungen zu generieren.



*McKinsey, Four myths about building a software business, 04/2021*

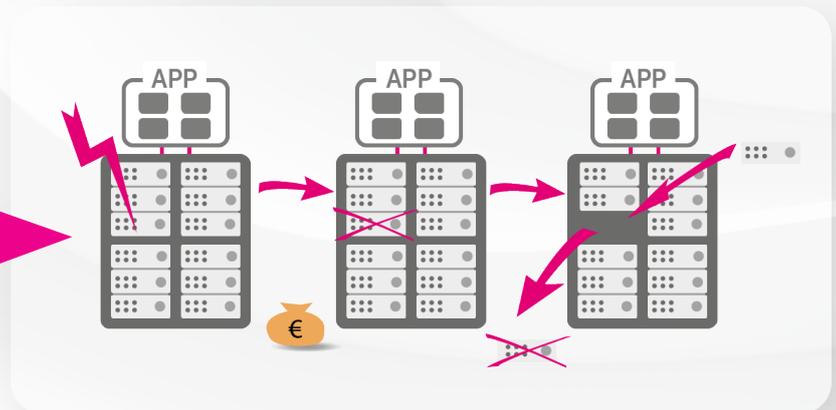
## Mode 1

Verfügbarkeit auf **Infrastrukturebene**



## Mode 2

Verfügbarkeit auf **Applikationsebene**



Doch Cloud-Infrastrukturen und -Plattformen sind nur „Enabler“. Unternehmen, die Applikationen in die Cloud migrieren, werden zunächst nicht viel davon haben – außer der Reduktion von Infrastrukturaufwänden und der Nutzung von Services aus dem Ökosystem der Cloud-Provider. **Differenzierung im Cloud-Zeitalter entsteht nicht auf der Infrastrukturebene, sondern auf der Applikationsebene** – bei der direkten Unterstützung von Businessprozessen. Die Cloud schafft die Basis für Agilität und Geschwindigkeit, doch das Herz der digitalen Wertschöpfung schlägt auf der Applikationsebene. Unternehmen kommen also nicht umhin, moderne Software-Kompetenz zu entwickeln. Mit anderen Worten:

## Wer Cloud sagt, muss auch DevOps sagen

– und sich zu einer entwicklerzentrierten Organisation entwickeln.

# Status DevOps

Egal ob Over-the-Air-Updates für Autos, immersive Einkaufserlebnisse oder AI-basierte Qualitätskontrolle in der Fertigung – immer mehr Produktmanager fordern und designen digitale Produkte auf Basis von Software, mit denen sie Kundenmehrwerte erzielen (wollen). Diese Produkte befinden sich – unabhängig von Branchen – in einem kontinuierlichen Entwicklungsprozess. Sie werden in physische Produkte eingebaut oder direkt als virtueller Service via Internet oder Mobilfunk geliefert.

**Die Kompetenz von Software-Bereitstellung und -Betrieb (Software Delivery and Operation – SDO) wird dadurch zu einem kritischen Erfolgsfaktor für Wettbewerbsdifferenzierung und Markterfolg.** Dabei spielen DevOps bzw. die DevOps-Fähigkeiten eines Unternehmens eine entscheidende Rolle. Und DevOps geht über einen technischen Blick hinaus: Mit DevOps adaptieren die IT-Mitwirkenden die Haltung der Business-Verantwortlichen als neuen „Mindset“. Sie sind nicht mehr nur für Bits und Bytes verantwortlich, sondern messen sich an der Kundenakzeptanz „ihres“ Produkts. Dies ist einer der Aspekte des Kulturwandels, der mit der Cloud einhergeht. DevOps führt Business-Ziele mit der Kompetenz eines Unternehmens zusammen, Software zu entwickeln und zu betreiben.

## DevOps-Prinzipien

Bisher waren Entwicklung und Betrieb unterschiedliche Welten. Unterschiedliche Werkzeuge, Methoden und Ziele bestimmten die Arbeit der beiden Domänen. Unternehmen, die cloud-nativ denken, schließen mit DevOps die Lücken zwischen Entwicklung und Betrieb. DevOps setzt auf fünf zentrale Maximen:

Aufbrechen  
von Organi-  
sationssilos

Weit-  
reichende  
Automati-  
sierung

Akzeptanz  
von Fehlern

Erhebung  
von Daten

Kleine  
Entwicklungs-  
schritte

Die Prämisse, an der sich Business-Verantwortliche orientieren, ist die schnelle Bereitstellung von Kundenmehrwerten, die letzten Endes in der ultimativen Währung der digitalen Ära münden: **Kundenzufriedenheit**. Kundenzufriedenheit, ein klassisches Business-Ziel, wird durch DevOps auch zum Maßstab des Erfolgs von IT. Die Relation „Höhere Kundenzufriedenheit durch DevOps“ wird aber nur aufgehen, wenn DevOps mit einem hohen Maß an Professionalität gelebt wird.

Doch welche Faktoren bestimmen die Qualität der SDO-Fähigkeiten eines Unternehmens? Der Accelerate State of DevOps-Report benennt **vier Faktoren**:

Wie schnell werden Code-Änderungen umgesetzt?

**LEAD TIME OF CODE CHANGES**

Wie schnell lässt sich ein Service wiederherstellen bei fehlerhaftem Neu-Code?

**TIME TO RESTORE SERVICE**

Wie häufig wird geänderter Code ausgerollt?

**DEPLOYMENT FREQUENCY**

Wie oft treten solche fehlgeschlagenen Code-Änderungen auf?

**CHANGE FAILURE RATE**

Obwohl DevOps eigentlich – wie der Name schon sagt – einen umfassenden Ansatz kennzeichnet, der Entwicklung und Betrieb von Software als Einheit sieht, lässt sich beobachten, dass – selbst bei den Spitzenreitern – bis zu 15% aller Rollouts nicht erfolgreich sind.

15%

Die Quote erhöht sich auf bis zu 30% bei den Unternehmen der nachfolgenden Rubriken.

30%



DORA, DevOps Research & Assessment, Sept. 2021

Unternehmen aus der Spitzengruppe „Elite“ rollen täglich mehrmals neuen Code aus. Die Zeit für Änderungen liegt **unter einer Stunde** – wie auch die Zeit zur Wiederherstellung eines Service. Maximal

**jeder 7.**

Rollout geht *schief*.

### Fehlgeschlagene Code-Änderungen

"Welcher Prozentsatz der Änderungen an der primären Anwendung oder dem Service, an dem Sie arbeiten, führt zu einer Verschlechterung oder einem Ausfall und erfordert anschließend eine Korrektur (z. B. einen Hotfix, Rollback, Fix Forward, Patch)?"

Elite Performers

0% - 15%

high to low Performers

16% - 30%

Auch DevOps muss sich also dem zentralen SDO-Dilemma stellen: entweder wird der Code schneller mit niedriger Qualität und höheren Risiken ausgeliefert oder die Risiken werden reduziert und die Qualität erhöht, wodurch sich die Bereitstellungszeiten verlängern (geringere Time to Market).

Wie lässt sich dieses Dilemma lösen?

Hier kommt Site Reliability Engineering ins Spiel ...

# Site Reliability Engineering

Geringere Service-Qualität oder reduzierte Wettbewerbsfähigkeit – es ist eine unangenehme Entscheidung, die Unternehmen vor diesem Hintergrund fällen müssen.

Die Ergebnisse des obigen Reports legen nahe, dass DevOps in der Realität immer noch einen starken Fokus auf der **Entwicklung** von Software hat. Mit der Definition eines fünften Erfolgsfaktors „Betriebs-Performanz“ trug der Report 2021 erstmals dem Aspekt des **Betriebs** cloud-nativer Applikationen stärker Rechnung:



\* vor dem Hintergrund der ständigen Änderungen

Die Betriebsseite muss Lösungen finden, um vom Tempo der Entwicklung nicht abgehängt zu werden. Sie muss einen stabilen und zuverlässigen Betrieb ermöglichen. **Falls dies nicht gelingt, führt (das gelebte) DevOps Unternehmen in ein systemisches Problem.**



*Die Reliability Pyramid bildet verschiedene Komponenten mit steigender Wertschöpfung für den zuverlässigen Betrieb eines Produkts/Service ab.*

Betriebsteams, die in der DevOps-Welt angekommen sind, haben diese Herausforderung bereits erkannt. Traditionelle Betriebsteams, die jährlich ein oder zwei Updates ausrollen, sind auf die Herausforderung noch nicht vorbereitet. Der kulturelle Wandel der Cloud schlägt damit auch hinter die Kulissen durch, mitten in die IT-Abteilung. Wenn Software-Entwickler zu Treibern der Unternehmensentwicklung werden, muss auch **der Betrieb von Applikationen neu gedacht werden**. Die Betriebsteams werden ebenfalls zu Treibern der Unternehmensentwicklung mit einer Verantwortung für den Business-Erfolg.

Mit dem Site Reliability Engineering (SRE) reagieren Betriebsteams auf die DevOps-Herausforderung. Die anstehenden Betriebs Herausforderungen werden durch die Einführung von Engineering-Praktiken (aus der Entwicklung) angegangen. Zugleich brechen Betriebsteams über SRE die etablierten Silos auf, initiieren so einen Austausch und die Kooperation über verschiedene Einheiten hinweg und schaffen gemeinsame Ziele. SRE greift auf dieselben Maximen zurück, die auch DevOps nutzt – damit wird tatsächlich eine gemeinsame Basis zwischen Betrieb und Entwicklung geschaffen. Wenn Betriebsteams die SRE-Prinzipien leben, resultiert daraus eine signifikante Erhöhung der SDO-Performanz.

SRE trägt maßgeblich zum Erfolg von DevOps-Praktiken bei:



mehr Performanz

&



bessere Business-Beiträge

**Site Reliability Engineering überträgt die Prinzipien von DevOps für den Betrieb. So bilden SRE-Teams DevOps-Prinzipien ab:**



**Aufbrechen von Organisationssilos**

Verwendung von identischen Tools für Entwicklung und Betrieb, geteilte Verantwortung für den Erfolg eines Produkts



**Akzeptanz von Fehlern**

Definition und Kontrolle eines „Error Budgets“, frühzeitige Problemlösung („shift left“)



**Kleine Entwicklungsschritte**

Validierung von Updates durch (begrenzte) Canary (Test) Deployments, automatisierte Rollbacks



**Weitreichende Automatisierung**

Identifikation der häufigsten manuellen Tätigkeiten/Abläufe, mehr als die Hälfte der Arbeitszeit wird für die Entwicklung von Verfahren aufgewendet, die 'manuelle Tätigkeiten reduzieren.



**Erhebung von Daten**

Einsatz von Service Level Indicators und Objectives, Telemetrie und Monitoring

# SRE-Werkzeuge

SRE nutzt drei zentrale Werkzeuge, um den Erfolg zu steuern:



**Service Level  
Agreements**



**Service Level  
Objectives**



**Service Level  
Indicators**

Die Idee hinter diesen drei Werkzeugen ist es, einen Bezug zwischen der Arbeit der Betriebsteams und den Business-Zielen des Unternehmens herzustellen.



[Google Cloud, SRE fundamentals: SLIs, SLAs and SLOs](#)

## Service Level Agreements

Service Level Agreements sind ein Klassiker in der IT-Leistungserbringung. Sie kennzeichnen ein Leistungsversprechen über einen bestimmten Zeitraum hinweg. Ein Dienstleister, der einen IT-Service bereitstellt und sein Kunde vereinbaren ein Leistungsziel, das in der Regel in Prozenten angegeben wird (z.B. 99,5 Prozent Serviceverfügbarkeit während eines Monats). Die Nichterfüllung der SLA hat in der Regel negative Konsequenzen für das Unternehmen oder die Einheit, die für den SLA verantwortlich ist – meist Pönalezahlungen, die explizit im SLA niedergelegt sind. Der SLA ist ein starkes Steuerungsinstrument für die Serviceerbringung.

# Service Level Objectives

Verfügbarkeit (eines Service) ist die zentrale Voraussetzung des Business-Erfolgs – das ist die Maxime von SRE. Verfügbarkeit im Sinne von SRE bedeutet, dass ein System zu einem bestimmten Zeitpunkt in der Lage ist, seine beabsichtigte Funktion zu erfüllen. Dieser Blickwinkel kann retrospektiv (als Reporting) eingesetzt werden und auch als Prognose für die Zukunft. Service Level Objectives sind **Zielwerte für die Leistungserfüllung** innerhalb eines SLA. Die SLO konkretisieren die Vereinbarungen des SLA. Und sie legen bei den Teams, die den Service bereitstellen, einen konkreten Erwartungswert für die Erfüllung der Business-Funktion fest, z.B. mit welcher Geschwindigkeit Anfragen in einem Shop-System abgearbeitet werden. Die Service Level Objectives dienen als Vorgaben, die erfüllt werden müssen, um den SLA zu erfüllen (und kennzeichnen gleichzeitig die Qualität des Betriebs in Abhängigkeit von Business-Kennzahlen).

Ohne einen solchen SLO kann ein Unternehmen keine Aussage über die Zuverlässigkeit eines Service machen. Aus dem SLO lassen sich dann auch die Anforderungen an Kosten und Entwicklungsgeschwindigkeit des Service ableiten. Eine zu hohe Systemverfügbarkeit erzeugt unnötige Kosten und Aufwände. Eine zu niedrige verhindert die Erfüllung der Business-Ziele.

Wir werden mit Ihnen zusammenarbeiten, um Ihre Anwendung zu unterstützen, solange Sie einen zuverlässigen Code entwickeln.



## SRE

- Verfügbarkeit
- Latenz
- Leistung
- Effizienz
- Change Management
- Incident Management
- Kapazitätenplanung

SLO



## Produkt

- Architektur
- Funktionelles Design
- Entwicklung von Features
- Testing

# Service Level Indicators

Mit den Service Level Indicators messen Betriebsteams ihre Arbeit und prüfen so die Erfüllung der SLO bzw. in der Summe die Erfüllung des SLA. Sie stellen sich auf die Nutzerseite und testen das Verhalten des Systems. Je häufiger das Verhalten des Systems den festgelegten Erwartungen entspricht, desto besser. Liegt aber die Zuverlässigkeit des Systems unter der vereinbarten Qualität, muss das Betriebsteam gegensteuern, um den SLA zu erreichen. Die Messbarkeit der Qualität eines Service ist – wie in den DevOps-Maximen vorgesehen – von zentraler Bedeutung auch für SRE. Unternehmen können für die Erfassung von SLIs auch auf cloud-interne Tools zurückgreifen. Bei AWS lassen sich SLIs via Tool automatisch messen. Es erstellt automatisierte Reports in Echtzeit.

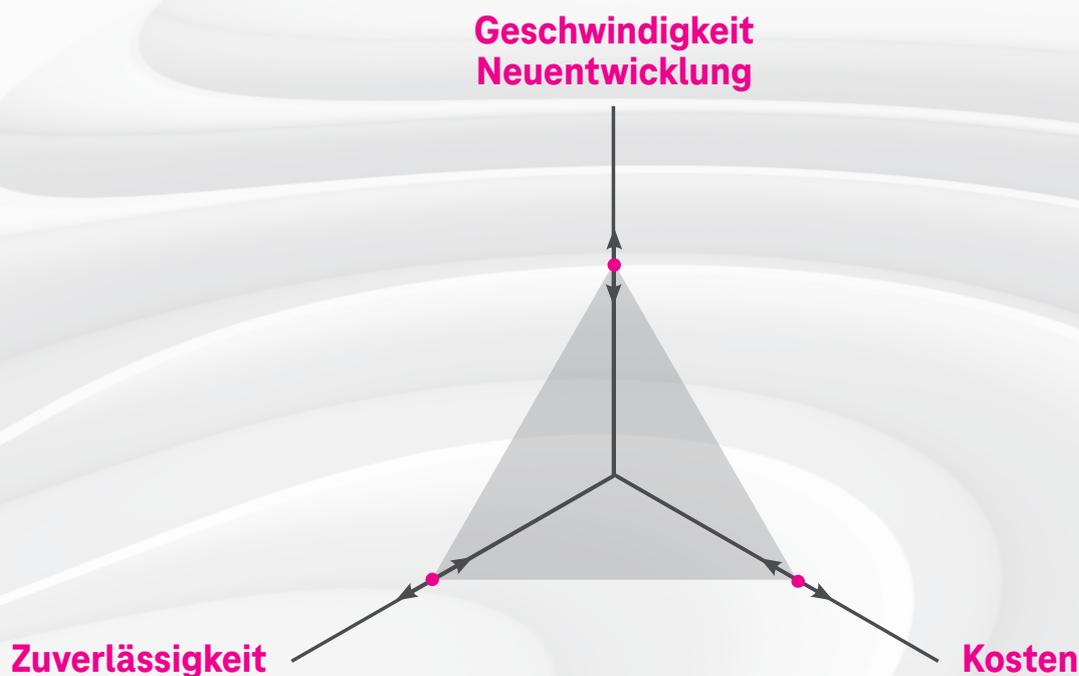
Beim Neudesign eines Systems sollte auch eine Vereinbarung für SLI und SLO als Teil der Systemanforderungen vorgesehen werden. Unternehmen, die SRE einführen wollen, sollten in einem ersten (priorisierten) Schritt derartige SLI und SLO für ihre bestehenden Systeme definieren.

# Vorteile von SRE

Mit der Etablierung der SLX-Methodik gewinnen Betriebsteams und ihre Kunden Klarheit über die erwarteten und zu erbringenden Leistungen und lösen sich von nicht quantifizierten Diskussionen („Die User Experience ist schlecht“). Beide Seiten – sowohl Kunde als auch Leistungserbringer – sind gefordert, Business-Erwartungen zu messbar zu definieren und regelmäßig zu überprüfen. Damit entsteht ein Rahmenwerk für einen erfolgreichen Betrieb von cloud-nativen Applikationen.

SRE ergänzt DevOps-Initiativen in Unternehmen optimal, so dass tatsächlich eine bidirektionaler Austausch auf Augenhöhe zwischen Entwicklung und Betrieb entsteht. So spiegeln erfahrene SRE-Teams beispielsweise ihre Betriebskompetenz in die Entwicklungsteams zurück und helfen bei der Identifikation von Bottlenecks in der Architektur. Die Applikationszuverlässigkeit und die Entwicklungsgeschwindigkeit werden sauber gegeneinander ausbalanciert, so dass – gemessen an den Zielen der Business-Einheit – das optimale Ergebnis für das Unternehmen entsteht. Die Business-Ziele werden zur Maßgabe für Entwicklung und Betrieb. Sie geben vor, wo ggfs. Schwerpunkte gesetzt werden.

Mit SRE hat der Betrieb eine Antwort auf die steigende Geschwindigkeit der Neuentwicklungen und kann die Konsequenzen der Business-Ziele quantifizieren. Durch den hohen Automatisierungsgrad im Betrieb werden CI/CD-Pipelines optimal unterstützt.





## Implementierung von DevOps/SRE

Prinzipiell existieren für Unternehmen, die DevOps und SRE einführen wollen, zwei grundlegende Möglichkeiten. Im „Make“-Fall etablieren Unternehmen cross-funktionale Teams, die die übergreifende DevOps-Verantwortung für einen Service/ein Produkt übernehmen. Dies ist der Ansatz aus dem Lehrbuch. In der Realität wird es aber nötig sein, die bislang autark und in ihren Silos agierenden Teams aus Entwicklung und Betrieb kulturell und methodisch zusammenzuführen. Spezifische Betriebsteams sind dann nicht mehr nötig. Das Unternehmen muss dann aber ein DevOps-Upskilling sicherstellen. Dabei kann es auch auf externe Expertise zurückgreifen. Das kann entweder über Schulungen geschehen oder ein „Training on the Job“, indem beispielweise externe Experten temporär in die Betriebsteams integriert werden und dort ihr Wissen weitergeben.

Alternativ dazu können sich Unternehmen auch für einen „Buy“-Ansatz entscheiden und die Ops-Seite von DevOps komplett an einen erfahrenen Partner auslagern. Auch hier gilt, dass eine enge Abstimmung zwischen den organisatorisch getrennten Einheiten erfolgen muss. Zudem muss eine einheitliche Tool- und Methodenbasis geschaffen werden, denn auch in diesem Fall teilen sich die beiden Einheiten die Verantwortung für die Zuverlässigkeit des Service. Der SLO wird in diesem Konstrukt zu einem entscheidenden Steuerungsinstrument (s. Grafik S. 14).

Die Realität in Unternehmen bewegt sich zwischen diesen beiden Polen. Erfahrene Dienstleister reagieren darauf mit modularen SRE-Support-Angeboten, bei dem sie auch nur Teilaufgaben übernehmen. Diese können sich beispielsweise an der Reliability Pyramid (S. 10) orientieren. In jedem Fall ist aber ein gelebter Shift-Left-Ansatz notwendig – das Betriebsteam muss frühzeitig einbezogen werden. Ideal ist die Einbindung in eine Cloud Center of Excellence wie im folgenden Beispiel.

# Einsatzbeispiel: Business-kritische Anwendungen der Deutschen Telekom

Die Deutsche Telekom hat im Rahmen ihrer Cloud-First-Strategie einige businesskritische Anwendungen in die AWS Cloud verlagert. Diese werden dort in einem agilen Modell kontinuierlich weiterentwickelt. Gleichzeitig will die Deutsche Telekom, dass die Services, die die Kundenschnittstelle bedienen, zuverlässig rund um die Uhr verfügbar sind. Die Services leisten Beiträge zu zwei herausragenden Zielen der Geschäftstätigkeit: Kundenzufriedenheit (durch Self Service) und Umsätze (durch Online-Sales).

Um einen zuverlässigen Betrieb sicherzustellen, hat das Cloud Center of Excellence der Deutsche Telekom IT ein SRE-Team von T-Systems integriert. Die Experten von T-Systems steuern ein an den Bedürfnissen des Großunternehmens ausgerichtetes Site Reliability Engineering bei. Sie arbeiten auf Basis von gemeinsamen Zuverlässigkeits- und Qualitätsstandards mit den Entwicklungseinheiten zusammen. Im Einzelnen sind dies gemeinsame CI/CD-Standards und SAFe als agile Methodik, aber auch die Privacy und Security Assessment Standards der Deutschen Telekom. Das T-Systems überwacht die Systeme 24/7 und verantwortet die Incident und Security Incident Response. Dazu greift das Team auf ein Set etablierter Methoden zurück (s. Box). Über das AWS-Know-how des T-Systems SRE-Teams ist auch der Einsatz AWS-spezifischer Tools wie AWS Code möglich.

## Exemplarische SRE Toolbox für AWS bei T-Systems

### Monitoring



Grafana

### Alerting



OpsGenie

### Loadtesting



K6

### CICD



Magenta CICD  
(GitLabCI)

### IaC



Cloudformation,  
Terraform

### Automation



Python/Lambda,  
Bash

### Documentation



GitLab pages

### Native AWS Dienste (Beisp.)



AWS Codebuild



## Fazit

Die Digitalisierung zeichnet den Weg zu einer höheren Business-Agilität vor. Das erfordert aber konsequenterweise den Einsatz agiler Entwicklungsmethoden. Doch auch hier dürfen Unternehmen nicht stehenbleiben, denn kurze Entwicklungszyklen fordern den Betrieb heraus. Wie kann in agilen Rahmenbedingungen neben Innovation auch die Zuverlässigkeit der Services gesichert werden? Die Antwort liegt in der Einführung von Site Reliability Engineering, das DevOps-Prinzipien für den Betrieb abbildet. Die Einführung von SRE erlaubt eine neue Kultur der Zusammenarbeit über klassische Einheitengrenzen hinweg und ein hohes Maß an Automatisierung. Silodenken wird durch gemeinsame Business-Ziele abgelöst. Nur dann werden Unternehmen die Früchte der Digitalisierung auch ohne bitteren Nachgeschmack genießen können.



**Sie wollen SRE einführen  
oder die Methodik näher  
kennenlernen?**

**Sprechen Sie uns an.**

### **Ihr Kontakt**

- Deutschland: 0800 33 09030
- International: 00800 33 090300
- Email: [info@t-systems.com](mailto:info@t-systems.com)
- Internet: [www.t-systems.com](http://www.t-systems.com)

### **Herausgeber**

T-Systems International GmbH  
Hahnstr. 43d  
60528 Frankfurt am Main  
Deutschland