

# Bericht

# ACCELERATE

# State of DevOps 2022



Gesponsert von

## Inhalt

<b>01</b>	<b>Kurzfassung</b>	03
<b>02</b>	<b>Wie schneiden Sie ab?</b>	08
<b>03</b>	<b>Wie können Sie sich verbessern?</b>	
	Einführung	19
	Cloud	21
	SRE und DevOps	26
	Technische DevOps-Ressourcen	29
	Firmenkultur	37
<b>04</b>	<b>Warum die Sicherheit der Lieferkette zählt</b>	42
<b>05</b>	<b>Überraschende Erkenntnisse</b>	55
<b>06</b>	<b>Demografische und firmenbezogene Merkmale</b>	59
<b>07</b>	<b>Schlussgedanken</b>	67
<b>08</b>	<b>Danksagungen</b>	68
<b>09</b>	<b>Autoren</b>	69
<b>10</b>	<b>Methodik</b>	73
<b>11</b>	<b>Weitere Informationen</b>	76

# 01

## Kurzfassung



Derek DeBellis



Claire Peters

In den letzten acht Jahren haben wir jedes Jahr den Bericht „Accelerate State of DevOps“ erstellt, wofür wir 33.000 Expertinnen und Experten befragt haben. Dabei wollten wir herausfinden, wie sich aufgrund von Ressourcen und Best Practices die Geschäftsergebnisse voraussagen lassen, die wir als für DevOps elementar betrachten:

- Leistung der Softwarebereitstellung – die vier wichtigen Messwerte der Softwarebereitstellungsleistung: Bereitstellungshäufigkeit, Vorlaufzeit für Änderungen, Änderungsfehlerquote und Zeit bis zur Dienstwiederherstellung.
- Operative Leistung – der fünfte wichtige Messwert, Zuverlässigkeit.
- Leistung der Organisation – Wie gut erreicht Ihre Organisation ihre Leistungs- und Profitabilitätsziele?

Wir konzentrieren uns außerdem auf die Faktoren, die anderen Geschäftsergebnissen zugrunde liegen, z. B. Burnout und wie wahrscheinlich es ist, dass Mitarbeitende ihre Teams weiterempfehlen.



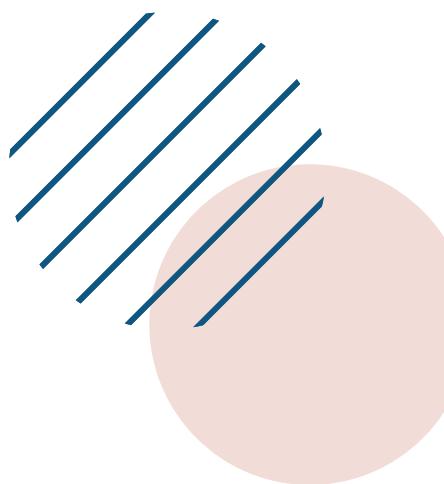
# Softwarelieferkette sichern

Im Jahr 2021 haben wir herausgefunden, dass die Softwarelieferkette unbedingt gesichert werden muss, um viele wichtige Geschäftsergebnisse erreichen zu können.

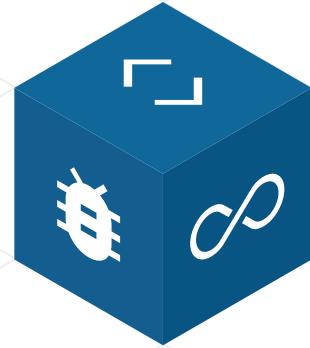
Dieses Jahr haben wir uns die Sicherheit der Softwarelieferkette noch genauer angesehen und sie zu einem der Kernthemen unserer Umfrage und unseres Berichtes gemacht. Wir haben das Framework [Supply Chain Levels for Software Artifacts \(SLSA\)](#) verwendet, um technische Best Practices zu erforschen, die dazu beitragen, die Softwarelieferkette sicherer zu gestalten. Zusätzlich haben wir das [Secure Software Development Framework \(NIST SSDF\)](#) des National Institute of Standards and Technology eingesetzt, um Einstellungen, Prozesse und nichttechnische Best Practices zu untersuchen, die mit dem Sichern der Softwarelieferkette im Zusammenhang stehen.

Wir haben herausgefunden, dass die wichtigsten Faktoren für Sicherheitsmaßnahmen bei der Anwendungsentwicklung einer Organisation kultureller und nicht technischer Natur sind: Bei einer von hohem Vertrauen geprägten Firmenkultur mit seltenen Schuldzuweisungen, bei der die Leistung im Fokus steht, ist es 1,6-mal wahrscheinlicher, dass das Unternehmen aufkommende Sicherheitspraktiken in überdurchschnittlichem Maße übernimmt, als wenn Misstrauen und Schuldzuweisungen vorherrschen und Macht oder Regeln im Fokus stehen. Außerdem haben wir erste Anzeichen gefunden, die darauf hindeuten, dass Sicherheitsscans, die vor der Bereitstellung ausgeführt werden, erfolgreich gefährdete Abhängigkeiten ermitteln, was zu weniger Sicherheitslücken im Produktionscode führt.

Wurden in puncto Anwendungsentwicklung zuverlässige Sicherheitsmaßnahmen eingeführt, korreliert dies mit weiteren Vorteilen. Wir haben festgestellt, dass es bei Teams, die sich darauf konzentrieren, solche Sicherheitsmaßnahmen einzuführen, seltener zum Burnout der Entwickelnden kommt. Bei Teams, die nur ein geringes Maß an Sicherheitsmaßnahmen implementiert haben, ist dagegen die Wahrscheinlichkeit, dass es zu vielen Fällen von Burnout kommt, 1,4-mal höher als im Fall von Teams mit einem hohen Maß an Sicherheitsmaßnahmen.<sup>1</sup> Auch empfehlen die Mitglieder der Teams, die sich darauf konzentrieren, Sicherheitsmaßnahmen zu implementieren, ihr Team deutlich häufiger an andere weiter. Außerdem lässt sich voraussagen, dass es sich auf die Leistung in der Organisation und die Leistung der Softwarebereitstellung positiv auswirkt, wenn SLSA-orientierte Sicherheitspraktiken eingesetzt werden. Damit dieser Effekt voll zum Tragen kommt, müssen Unternehmen jedoch leistungsfähige Continuous-Integration-Funktionen haben.



<sup>1</sup>Für diese Statistik definieren wir ein hohes Maß (z. B. an Sicherheitsmaßnahmen) als einen z-Score von  $\geq 1$  und ein geringes Maß als einen z-Score von  $\leq -1$ .



# Faktoren der Unternehmensleistung

Abgesehen von den oben erwähnten Sicherheitspraktiken lassen sich die Schlüsselvariablen, die sich auf die Unternehmensleistung auswirken, meist den folgenden Kategorien zuordnen:

## Organisations- und Teamkultur

Firmen mit einer vertrauensvollen Unternehmenskultur, in der es zu wenig Schuldzuweisungen kommt, so wie Dr. [Westrum](#) dies definiert hat, weisen tendenziell eine bessere Unternehmensleistung auf. Auch Organisationen mit Teams, die sich durch eine entsprechende Finanzierung und Führungskräftepatenschaften unterstützt fühlen, verzeichnen meist eine höhere Leistung. Wenn Teams stabil sind und deren Mitglieder das eigene Team positiv wahrnehmen (Wahrscheinlichkeit des Weiterempfehlens), trägt das meist ebenfalls in Form einer besseren Organisationsleistung Früchte. Auch bei Firmen mit flexiblen Arbeitsregelungen ist die Unternehmensleistung hoch.

## Zuverlässigkeit

Die Praktiken, die wir mit Reliability Engineering in Verbindung bringen (z. B. klare Ziele und herausragende Messwerte in Bezug auf die Zuverlässigkeit usw.), und das Maß, in dem die Belegschaft berichtet, dass sie ihre Erwartungen an die Zuverlässigkeit erfüllt hat, sind aussagekräftige Anzeichen einer guten Organisationsleistung.

## Cloud

Wir haben herausgefunden, dass sich auch aus dem Einsatz der Cloud Rückschlüsse auf die Unternehmensleistung ziehen lassen. Firmen mit Software, die direkt in der und für die Cloud entwickelt wurde, punkten in der Regel durch eine höhere Leistung der Organisation. Wenn Firmen private, öffentliche oder Hybrid-Clouds oder eine Kombination verschiedener Clouds einsetzen, spiegelt sich das in einer besseren Unternehmensleistung wider, als wenn nur lokale Server verwendet werden. Bei Unternehmen, die auf mehrere öffentliche Clouds setzen, ist es 1,4-mal wahrscheinlicher, dass ihre Leistung überdurchschnittlich ist, als bei den Firmen, die das nicht tun.

Was unser Dataset angeht, scheint sich der Einsatz der Cloud auch aufgrund anderer Faktoren auf die Unternehmensleistung auszuwirken. Ein Beispiel dafür ist die Sicherheit der Lieferkette. Diesbezüglich haben wir festgestellt, dass Organisationen, die öffentliche Clouds nutzen, auch eher SLSA-Praktiken implementieren – vielleicht, da zahlreiche Cloud-Anbieter diverse Bausteine für viele der SLSA-Praktiken bereitstellen und den Einsatz dieser Praktiken begrüßen, z. B. das Automatisieren von Builds und Bereitstellungen.<sup>2,3</sup> Im Großen und Ganzen geht es hier darum, dass Teams durch den Einsatz von Cloud-Plattformen viele Funktionen und Best Practices übernehmen bzw. einführen können, die schließlich zu einer höheren Firmenleistung führen.

<sup>2</sup> Jung, Sun Jae. „Introduction to Mediation Analysis and Examples of Its Application to Real-world Data.“ *Journal of Preventive Medicine and Public Health = Yebang Uihakhoe chi* Vol. 54,3 (2021): 166-172. doi:10.3961/jpmph.21.069.

<sup>3</sup> Carrión, Gabriel Cepeda, Christian Nitzl und José L. Roldán. „Mediation Analyses in Partial Least Squares Structural Equation Modeling: Guidelines and Empirical Examples.“ *Partial Least Squares Path Modeling*. Springer, Cham, 2017. 173-195.

## Der Kontext zählt

Das DORA-Team berücksichtigt seit Langem, dass solche Effekte davon abhängig sind, unter welchen Bedingungen Teams arbeiten. Wir sind überzeugt, dass es wichtig ist, die Eigenschaften eines Teams (Prozesse, Stärken, Einschränkungen und Ziele) und die Umgebung zu kennen, in der es arbeitet. Beispielsweise kann eine technische Fähigkeit, die in einem Kontext von Vorteil ist, in einem anderen negative Konsequenzen nach sich ziehen. Dieses Jahr haben wir unseren Fokus darauf gelegt, diese angenommenen Bedingungen in Form von Interaktionen explizit zu modellieren; viele dieser Hypothesen werden durch die Daten des diesjährigen Berichts gestützt:

- Eine hohe Leistung bei der Softwarebereitstellung wirkt sich nur dann positiv auf die Firmenleistung aus, wenn die operative Leistung ebenfalls hoch ist. Eine schnelle Bereitstellung spielt vielleicht keine Rolle, wenn Ihr Dienst nicht die Erwartungen erfüllt, die die Nutzenden an seine Zuverlässigkeit stellen.
- Wenn Sie für die Softwarelieferkette Sicherheitskontrollen implementieren, wie die Kontrollen, die im SLSA-Framework empfohlen werden, wirkt sich das positiv auf die Leistung der Softwarebereitstellung aus – sofern die Continuous Integration bereits fest im Unternehmen verankert ist. Wenn keine Continuous-Integration-Funktionen etabliert sind, können die Softwarebereitstellungsleistung und die Sicherheitskontrollen miteinander in Konflikt stehen.
- Die Auswirkungen von SRE-Praktiken (Site Reliability Engineering) auf die Fähigkeit eines Teams, seine Ziele in Sachen Zuverlässigkeit zu erreichen, ist nicht linear. Wenn SRE zum Einsatz kommt, wirkt sich das erst dann positiv auf die Zuverlässigkeit aus, wenn ein Team bezüglich SRE ein bestimmtes Maß an Reife erlangt hat. Erst wenn ein Team diesen SRE-Reifegrad erlangt hat, können wir einen Zusammenhang zwischen SRE und dem Erreichen seiner Zuverlässigkeitsziele feststellen. Wenn ein Team zunehmend mehr SRE-Praktiken einsetzt, erreicht es irgendwann einen Wendepunkt, ab dem sich aus dem Einsatz von SRE-Praktiken die Zuverlässigkeit in hohem Maße prognostizieren lässt. Die verbesserte Zuverlässigkeit wirkt sich dann wiederum auf die Firmenleistung aus.
- Technische Fähigkeiten bauen aufeinander auf. Continuous Delivery und Versionsverwaltung verstärken sich gegenseitig dabei, bei der Softwarebereitstellung für ein hohes Maß an Leistung zu sorgen. Wenn Continuous Delivery, eine lose gekoppelte Architektur, die Versionsverwaltung und Continuous Integration gemeinsam verwendet werden, ist die Leistung der Softwarebereitstellung größer, als wenn diese Techniken nur einzeln eingesetzt werden.



Die Bedingungen, von denen die Bereitstellung abhängt, und die Notwendigkeit, die Arbeitsbedingungen der Teams zu verstehen, führen uns zu einer Schlussfolgerung, die folgender Erkenntnis aus dem Jahr 2021 ähnelt:

„Damit Verbesserungen auch sinnvoll sind, müssen Teams eine Philosophie der kontinuierlichen Verbesserung verfolgen. Mithilfe von Benchmarks können Sie messen, wo Sie aktuell stehen, und auf Basis der untersuchten Funktionen etwaige Einschränkungen ermitteln. Außerdem können Sie so testweise Verbesserungen vornehmen, um diese Einschränkungen zu beseitigen. Dieser Testprozess wird von Erfolgen und Misserfolgen geprägt sein, aber in beiden Fällen können Ihre Teams dann sinnvolle Maßnahmen ergreifen, da sie dadurch hilfreiche Erkenntnisse gewonnen haben.“

In der Tat konnten wir dieses Jahr einen Effekt ausmachen, der gut zu dieser übergreifenden Philosophie passt: Bei Teams, die die Notwendigkeit anerkennen, kontinuierlich Verbesserungen vorzunehmen, ist die Leistung in der Organisation tendenziell höher als bei denen, die das nicht tun.

Kurz gesagt müssen die Teams kontinuierlich Anpassungen vornehmen und auch verschiedene Praktiken zum Entwickeln von Software ausprobieren.

Das wissen wir, da Teams, die so vorgehen, im Großen und Ganzen eine höhere Unternehmensleistung aufweisen. Das trifft zwar nicht immer zu (was einer Organisation hilft, funktioniert nicht zwingend auch für eine andere), aber meistens. Wenn Sie selbst mit DevOps-Praktiken experimentieren, um nach und nach herauszufinden, was für Ihr Team funktioniert, sollten Sie auch mit gelegentlichen Fehlschlägen rechnen.

Dieses Jahr haben die Daten auch eine Reihe [überraschender Erkenntnisse](#) ans Licht gebracht, aber dazu möchten wir an dieser Stelle noch nicht ins Detail gehen.

Bei Teams, die die Notwendigkeit anerkennen, kontinuierlich Verbesserungen vorzunehmen, ist die Leistung in der Organisation tendenziell höher als bei denen, die das nicht tun.

# 02

## Wie schneiden Sie ab?



Derek DeBellis

Möchten Sie wissen, wie Ihr Team im Vergleich zu den Teams anderer Unternehmen der Branche abschneidet? Dieser Abschnitt enthält die neueste Benchmarkbewertung der DevOps-Leistung diverser Firmen. Dazu haben wir erforscht, wie die untersuchten Teams Softwaresysteme entwickeln, bereitstellen und betreiben. Dann haben wir die Teilnehmenden zu natürlichen Gruppen (Clustern) zusammengefasst, die die häufigsten DevOps-Leistungsmuster abdecken.

Dieses Jahr haben wir zwei verschiedene Clustering-Verfahren einbezogen. Das erste basiert auf historischen Präzedenzfällen. Bei diesem Clustering-Verfahren liegt der Fokus darauf, die Cluster anhand von vier Messwerten zu erstellen, die die Leistung der Softwarebereitstellung erfassen: Vorlaufzeit, Bereitstellungshäufigkeit, Zeit bis zur Dienstwiederherstellung und Änderungsfehlerquote. Diese werden weiter unten alle separat erklärt. Ziel dieses Verfahrens ist es, Sie dabei zu unterstützen, die derzeitige Leistung Ihres Teams zu quantifizieren, damit Sie diese mit der Leistung anderer Teams vergleichen können.

Beim zweiten Clustering-Verfahren wird noch die Zuverlässigkeit als fünfter Messwert berücksichtigt, den wir verwenden, um die operative Leistung nachzuvollziehen. Warum haben wir unsere Clusteranalyse um einen neuen Messwert ergänzt? Da wir immer wieder gesehen haben, wie wichtig dieser Messwert ist. In der Tat haben wir Hinweise, die darauf hindeuten, dass sich die Bereitstellungsleistung nachteilig auf die Unternehmensleistung auswirken kann, wenn nicht gleichzeitig eine starke operative Leistung vorliegt. Anders als bei unserem traditionellen Clustering-Verfahren handelt es sich hierbei um ein beschreibendes Verfahren, bei dem versucht wird, häufig vorkommende Muster von Teams in puncto Bereitstellungsleistung und operativer Leistung abzubilden. Daher ist es nicht immer offensichtlich, welches Clustering-Verfahren besser geeignet ist.

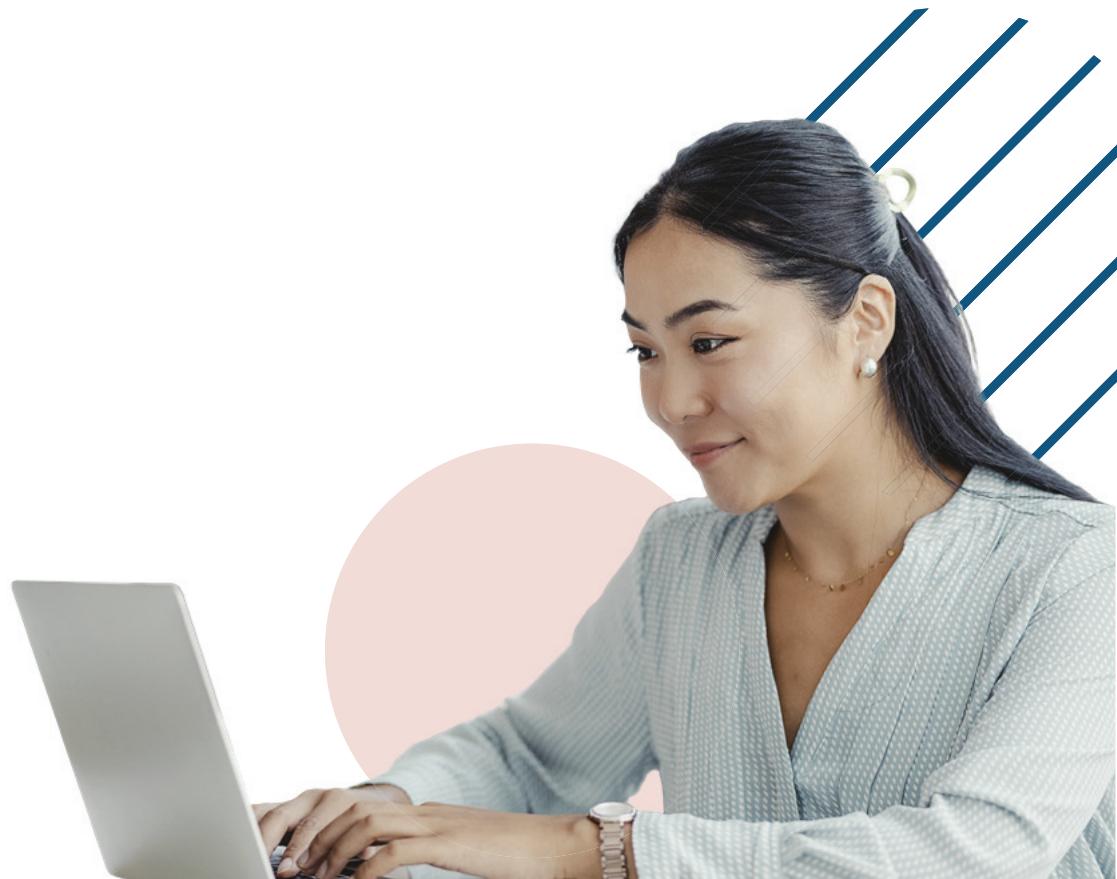
Schauen wir uns zuerst eine kurze Beschreibung der fünf Messwerte an, die wir verwendet haben, um die Softwarebereitstellungsleistung und operative Leistung nachzuvollziehen.



## Softwarebereitstellungsleistung und operative Leistung

Um den Anforderungen sich ständig ändernder Branchen gerecht zu werden, müssen Unternehmen ihre Software schnell und zuverlässig bereitstellen und betreiben. Je schneller Ihre Teams an der Software Änderungen vornehmen können, desto schneller können Sie Ihren Kundinnen und Kunden einen Mehrwert bieten, Tests durchführen und so wertvolles Feedback erhalten. Durch unsere Datenerhebung und Forschung über acht Jahre hinweg haben wir vier Messwerte entwickelt und validiert, die die Leistung der Softwarebereitstellung in Firmen messen. Seit 2018 verwenden wir noch einen fünften Messwert, um die operativen

Funktionen zu erfassen. Teams, die bei allen fünf Messwerten gut abschneiden, weisen auch eine außergewöhnlich hohe Unternehmensleistung auf. Wir fassen diese fünf Messwerte unter dem Begriff **Softwarebereitstellungs- und operative Leistung (Software Delivery and Operational Performance, SDO)** zusammen. Beachten Sie, dass bei diesen Messwerten Ergebnisse auf Systemebene im Fokus stehen. Dadurch können Fallstricke vermieden werden, die sich häufig ergeben, wenn Softwaremesswerte verfolgt werden. Diese können nämlich dazu führen, dass zulasten der Gesamtergebnisse Aufgabenbereiche gegeneinander ausgespielt werden und lokale Optimierungen vorgenommen werden.



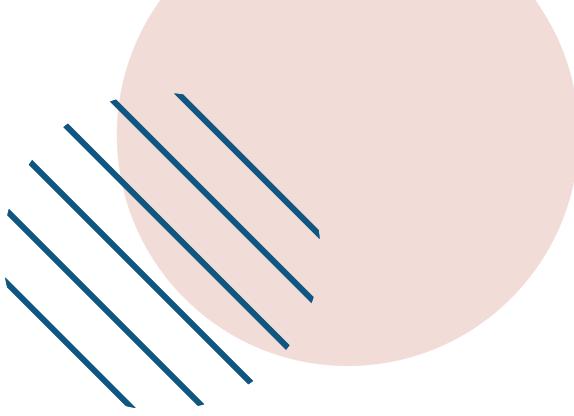
## Fünf Messwerte für die Bereitstellungs- und operative Leistung

Die vier Messwerte der Softwarebereitstellungsleistung lassen sich in Durchsatz und Stabilität ausdrücken.

Der Durchsatz wird anhand der **Vorlaufzeit für Codeänderungen** (also der Zeit vom Code-Commit bis zum Bereitstellen in der Produktion) und der **Bereitstellungshäufigkeit** gemessen. Wir messen die Stabilität anhand der benötigten **Zeit bis zur Dienstwiederherstellung** nach einem Vorfall und anhand der **Änderungsfehlerquote**.

Der fünfte Messwert erfasst dann die **operative Leistung** und ist ein Maßstab für moderne Betriebspraktiken. Wir legen der **operativen Leistung** den Messwert **Zuverlässigkeit** zugrunde. Mit Zuverlässigkeit ist gemeint, wie gut Ihre Dienste die Erwartungen Ihrer Nutzenden erfüllen, z. B. hinsichtlich Verfügbarkeit und Leistung. In der Vergangenheit haben wir die Verfügbarkeit, aber nicht die Zuverlässigkeit gemessen. Da die Verfügbarkeit jedoch beim Reliability Engineering einen besonderen Schwerpunkt darstellt, messen wir seit 2021 auch die Zuverlässigkeit. Dadurch sind bei unseren Messungen Verfügbarkeit, Latenz, Leistung und Skalierbarkeit breiter vertreten. Konkret haben wir die Teilnehmenden gebeten, zu bewerten, wie gut sie in der Lage sind, ihre Zuverlässigkeitsziele zu erreichen bzw. zu übertreffen. Wir haben festgestellt, dass Teams mit unterschiedlichen Quoten in puncto Bereitstellungsleistung dann bessere Ergebnisse erzielen (z. B. weniger Fälle von Burnout), wenn sie gleichzeitig auch die operative Leistung priorisieren.





## Das altbewährte Clustering-Verfahren nach Bereitstellungsleistung

Dieses Jahr haben wir beim Auswerten der Daten der Vier-Cluster-Lösung, die wir seit 2018 verwenden, festgestellt, dass sich aus diesen klar ein Niedrigleistungs-Cluster und ein Hochleistungs-Cluster bilden lassen. Jedoch wiesen die zwei Cluster, die wir normalerweise verwendet hätten, um die mittlere Leistung und die hohe Leistung voneinander abzugrenzen, nicht genug Unterschiede auf, um diese Aufteilung zu rechtfertigen. Außerdem deuteten die verschiedenen Indizes, die wir zum Auswählen der richtigen Clusterlösung verwendet haben, ausnahmslos

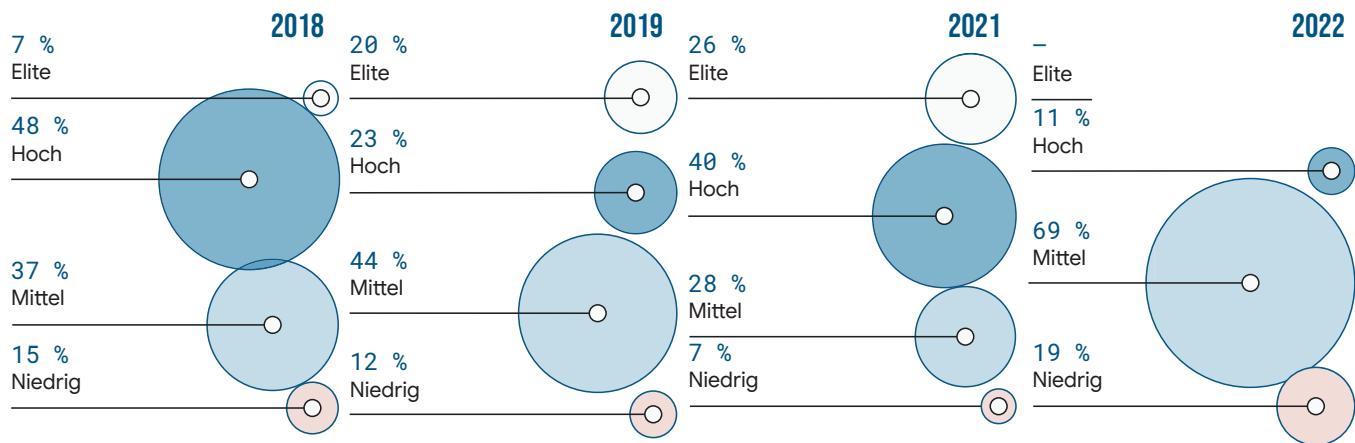
darauf hin, dass drei Cluster die Daten am besten abbilden – unabhängig vom verwendeten Clustering-Verfahren. In der Tabelle unten sind die Merkmale der Bereitstellungsleistung für die einzelnen Cluster beschrieben.

Der auffällige Unterschied zum Vorjahr ist, dass es dieses Jahr keinen Elite-Cluster gibt. Der diesjährige Hochleistungs-Cluster ist eine Kombination aus dem Hochleistungs-Cluster und dem Elite-Cluster des letzten Jahres. Wir haben uns dazu entschieden, den Elite-Cluster nicht separat auszuweisen, da der Cluster, der dieses Jahr die höchste Leistung abbildet, schlachtweg nicht genug der Merkmale des letzjährigen Elite-Clusters zeigt.

Messwert der Softwarebereitstellungsleistung	Niedrig	Mittel	Hoch
<b>Bereitstellungshäufigkeit</b> Für die primäre Anwendung oder den primären Dienst, an der bzw. an dem Sie arbeiten: Wie oft stellt Ihre Organisation Code für die Produktion bereit oder gibt diesen für Endnutzende frei?	Zwischen einmal im Monat und einmal alle sechs Monate	Zwischen einmal pro Woche und einmal im Monat	Nach Bedarf (mehrere Bereitstellungen pro Tag)
<b>Vorlaufzeit für Änderungen</b> Für die primäre Anwendung oder den primären Dienst, an der bzw. an dem Sie arbeiten: Wie lange ist Ihre Vorlaufzeit für Änderungen (d. h., wie lange dauert es vom Commit des Codes zum erfolgreichen Ausführen des Codes in der Produktion)?	Zwischen einem Monat und sechs Monaten	Zwischen einer Woche und einem Monat	Zwischen einem Tag und einer Woche
<b>Zeit bis zur Dienstwiederherstellung</b> Für die primäre Anwendung oder den primären Dienst, an der bzw. an dem Sie arbeiten: Wie lange dauert es im Allgemeinen bis zur Wiederherstellung des Diensts, wenn ein Dienstvorfall oder ein Fehler auftritt, der sich auf Nutzende auswirkt (z. B. ein ungeplanter Ausfall oder eine Beeinträchtigung des Diensts)?	Zwischen einer Woche und einem Monat	Zwischen einem Tag und einer Woche	Weniger als einen Tag
<b>Änderungsfehlerquote</b> Für die primäre Anwendung oder den primären Dienst, an der bzw. an dem Sie arbeiten: Welcher Prozentsatz der Änderungen an der Produktion oder der für Nutzende freigegebenen Version führt zu einem eingeschränkten Dienst, z. B. zu Dienstbeeinträchtigungen oder -ausfällen, und erfordert infolgedessen eine Korrektur, z. B. einen Hotfix, ein Rollback, einen Fix-Forward oder einen Patch?	46 %–60 %	16 %–30 %	0 %–15 %

Das deutet darauf hin, dass dieser Querschnitt keine Teams und Organisationen repräsentiert, bei denen es Mitarbeitende gibt, die den Eindruck haben, dass sie Fortschritte gemacht haben. Eine mögliche Hypothese, die wir aktuell nicht mit Daten unterstützen können, ist, dass in puncto Softwareentwicklung weniger Innovationen geschaffen wurden, was Praktiken, Tools und die Weitergabe von Daten angeht. Das könnte die Folge der immer noch andauernden Pandemie sein, die die Fähigkeit der Belegschaften einschränkt, Wissen und Praktiken team- und organisationsübergreifend weiterzugeben. Möglicherweise gab es weniger Möglichkeiten, sich mit Kolleginnen und Kollegen zu treffen und voneinander zu lernen, was wiederum zu einer langsameren Innovation geführt haben kann. Wir hoffen, noch tiefgreifendere Nachforschungen anstellen zu können, um herauszufinden, was dieser Entdeckung zugrunde liegt.

Trotz allem zeigt ein Vergleich des Niedrigleistungs-, Mittelleistungs- und Hochleistungs-Clusters aus diesem Jahr mit den Clustern aus dem letzten Jahr, dass es zu einer Verschiebung in Richtung einer etwas höheren Bereitstellungsleistung kam. Es zeigt sich, dass die Cluster dieses Jahres zwischen zwei Clustern des Vorjahrs liegen. Der Hochleistungs-Cluster von 2022 liegt zwischen dem Hochleistungs-Cluster und dem Elite-Cluster von 2021. Der Niedrigleistungs-Cluster von 2022 scheint zwischen dem Niedrigleistungs-Cluster und dem Mittelleistungs-Cluster von 2021 zu liegen. Die Verschiebung des Niedrigleistungs-Clusters nach oben deutet darauf hin, dass zwar die Obergrenze der Bereitstellungsleistung herabgesetzt, aber eben auch die Untergrenze angehoben wurde.

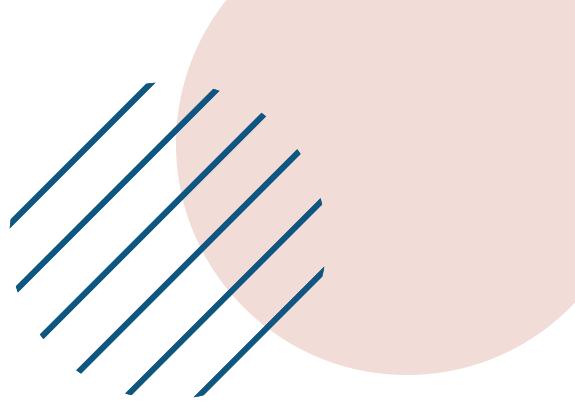


Die prozentuale Aufschlüsselung der obigen Tabelle zeigt, dass der Prozentsatz der leistungsstarken Mitarbeitenden auf ein 4-Jahres-Tief gefallen ist, während der Prozentsatz der leistungsschwachen Mitarbeitenden stark angestiegen ist – von 7 % im Jahr 2021 auf 19 % in diesem Jahr. Mehr als zwei Drittel der diesjährigen Teilnehmenden sind dem mittleren Cluster zuzuordnen. Der deutliche Rückgang an leistungsstarken Mitarbeitenden und Mitarbeitenden auf Elite-Niveau kann vermuten lassen, dass viele der diesjährigen Teilnehmenden in Organisationen oder Teams arbeiten, die entweder noch keine DevOps-Kultur – wie sie gerade rund um viele moderne Teams entsteht – geschaffen haben oder gerade dabei sind.

Vielleicht konzentrieren wir uns zu sehr auf die Unterschiede zwischen 2021 und 2022, statt die Ähnlichkeiten hervorzuheben. Die Cluster aus 2021 und 2022 weisen viele gemeinsame Merkmale auf, einschließlich einer großen Kluft zwischen leistungsstarken und leistungsschwachen Mitarbeitenden. Beispielsweise führen leistungsstarke Mitarbeitende schätzungsweise<sup>1</sup> 417-mal mehr Bereitstellungen durch als leistungsschwache Mitarbeitende.



<sup>1</sup> Im Abschnitt „Methodik“ erfahren Sie, wie wir auf diesen geschätzten Wert gekommen sind.



## Clusteranalyse der Bereitstellungs- und operativen Leistung

Wir haben uns dazu entschieden, eine Clusteranalyse der drei Kategorien durchzuführen, die von den fünf Messwerten dargestellt werden sollen: **Durchsatz** (eine Kombination aus „Vorlaufzeit von Codeänderungen“ und „Bereitstellungshäufigkeit“), **Stabilität** (eine Kombination aus „Zeit bis zur Dienstwiederherstellung“ und „Änderungsfehlerquote“) und **operative Leistung** („Zuverlässigkeit“).

Der Grund dafür war die entscheidende Rolle, die die operative Leistung bei unseren Modellen spielt. Im Fall von Organisationen, die keine solide operative Leistung aufweisen, wirken sich Durchsatz und Stabilität weniger stark auf die Unternehmensleistung aus. Wir sind der Meinung, dass wir ein lückenhaftes Bild der DevOps-Leistungslandschaft zeichnen würden, wenn wir die dafür entscheidende operative Leistung nicht miteinbeziehen würden.

Das Sichten der Daten hat uns veranlasst, eine Vier-Cluster-Lösung zu wählen. Hier sehen Sie die Namen der vier Cluster und wie sie aufgeschlüsselt werden:

Cluster	Stabilität		Operative Leistung	Durchsatz		% der Teilnehmenden
	Zeit bis zur Dienstwiederherstellung	Änderungsfehlerquote	Zuverlässigkeit	Vorlaufzeit	Bereitstellungs-häufigkeit	
Wird gestartet	Zwischen einem Tag und einer Woche	31 %–45 %	Erwartungen werden manchmal erfüllt	Zwischen einer Woche und einem Monat	Zwischen einmal pro Woche und einmal im Monat	28 %
Im Fluss	Weniger als eine Stunde	0 %–15 %	Erwartungen werden meist erfüllt	Weniger als einen Tag	Nach Bedarf (mehrere Bereitstellungen pro Tag)	17 %
Abflauend	Weniger als einen Tag	0 %–15 %	Erwartungen werden meist erfüllt	Zwischen einer Woche und einem Monat	Zwischen einmal pro Woche und einmal im Monat	34 %
Wird eingestellt	Zwischen einem Monat und sechs Monaten	46 %–60 %	Erwartungen werden meist erfüllt	Zwischen einem Monat und sechs Monaten	Zwischen einmal im Monat und einmal alle sechs Monate	21 %

Wenn Sie zwei Teams besuchen würden, die zum selben Cluster gehören, wirken diese jedoch wahrscheinlich sehr unterschiedlich und unsere Beschreibung deckt sich eventuell nicht mit Ihrer Beobachtung. Die Beschreibungen der Cluster weiter oben im Text stellen einen Versuch dar, unseren Erfahrungsschatz aus der Arbeit mit unzähligen Teams zu nutzen, um Ihnen die beschriebenen Muster im Datenmaterial verständlich zu machen. Außerdem kann es sein, dass ein Team, wenn Sie es zu zwei verschiedenen Zeitpunkten besucht hätten, nicht im selben Cluster geblieben wäre. Ein möglicher Grund dafür könnte sein, dass das Team entweder leistungsstärker oder leistungsschwächer geworden ist. Eine andere Möglichkeit besteht darin, dass das Team zu einem Bereitstellungsmuster gewechselt ist, das besser zum aktuellen Status seiner Anwendung bzw. seines Dienstes passt. Beispielsweise hat ein Team vielleicht zu einem früheren Zeitpunkt der Anwendungs- bzw. Dienstentwicklung eher explorativ gearbeitet (Cluster „Wird gestartet“). Während es langsam die passende Nische findet, legt es jedoch möglicherweise seinen Fokus auf die Zuverlässigkeit (Cluster „Im Fluss“ oder „Abflauend“).

## Jeder Cluster weist eindeutige Merkmale auf und macht einen erheblichen Teil der Antworten aus.

Was all unsere Größen angeht, schneidet der Cluster **Wird gestartet** Cluster weder gut noch schlecht ab. Teams in diesem Cluster stecken möglicherweise noch in einem sehr frühen Entwicklungsstadium ihres Produktes, Features oder Dienstes. Die Teammitglieder sind dann eventuell weniger auf die Zuverlässigkeit fokussiert, da sie primär versuchen, Feedback einzuholen und zu verstehen, wie gut ihr Produkt für seinen aktuellen Markt geeignet ist – allgemeiner gesprochen arbeiten sie noch verstärkt explorativ.

Der Cluster **Im Fluss** schneidet hinsichtlich aller Merkmale gut ab: hohe Zuverlässigkeit, hohe Stabilität, hoher Durchsatz. Nur 17 % der Teilnehmenden befinden sich in diesem Zustand, in dem alles im Fluss ist.

Teilnehmende im Cluster **Abflauend** nehmen nur hin und wieder Bereitstellungen vor, aber wenn sie es tun, sind sie sehr wahrscheinlich damit erfolgreich. Mehr als ein Drittel der Antworten entfallen auf diesen Cluster, sodass er der größte und repräsentativste unseres Querschnitts ist. Dieses Muster ist wahrscheinlich für ein Team typisch, das sich schrittweise verbessert. Das Team und seine Kundinnen und Kunden sind jedoch größtenteils mit dem aktuellen Status der Anwendung bzw. des Produktes zufrieden. Natürlich kann dieses Muster aber auch bei anderen Teams auftreten.

Und zu guter Letzt verbirgt sich hinter dem Cluster **Wird eingestellt** ein Team, das an einem Dienst oder einer Anwendung arbeitet, der bzw. die ihm und seinen Kundinnen und Kunden noch wichtig ist, sich aber nicht mehr in der aktiven Entwicklungsphase befindet.

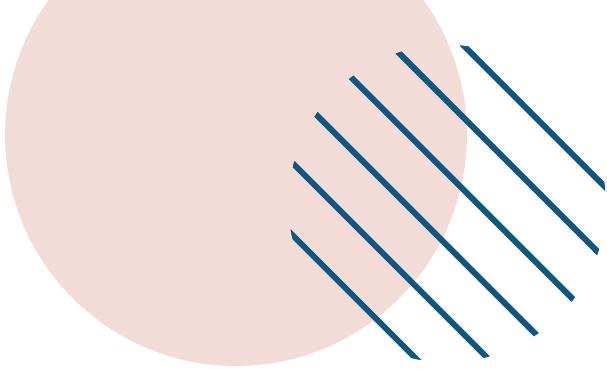
Diese Cluster unterscheiden sich merklich voneinander – hinsichtlich der von den Teams eingesetzten Praktiken und ihren technischen Fähigkeiten. Da die Teams des Clusters **Im Fluss** so gut abschnitten, was Softwarebereitstellung und operative Leistung angeht, haben wir uns entschieden, zu erforschen, worin sie sich hinsichtlich ihrer Praktiken und technischen Fähigkeiten von den Teams der anderen Cluster unterscheiden. Wir haben herausgefunden, dass sich im Vergleich zu den anderen Clustern die Teams des Clusters **Im Fluss** mehr auf folgende Bereiche konzentrieren:

- Lose gekoppelte Architekturen: das Ausmaß, in welchem Teams umfangreiche Änderungen am Design ihres Systems vornehmen können, ohne darauf angewiesen zu sein, dass andere Teams Änderungen an ihren eigenen Systemen vornehmen
- Flexibilität bieten: wie flexibel die Arbeitsregelungen einer Firma sind
- Versionsverwaltung: wie Änderungen am Anwendungscode, an der Systemkonfiguration, an der Anwendungskonfiguration usw. gemanagt werden
- Continuous Integration (CI): wie häufig Entwicklungszweige im Hauptentwicklungszweig zusammengeführt werden
- Continuous Delivery (CD): Funktionen, die darauf ausgerichtet sind, Änderungen sicher, nachhaltig und effizient in die Produktion zu überführen

Merkwürdigerweise fokussieren sich die Teams des Clusters **Im Fluss** tendenziell weniger auf die Dokumentation. Letztes Jahr haben wir herausgefunden, dass Dokumentationspraktiken sowohl bei der Softwarebereitstellungsleistung wie auch bei der operativen Leistung (zusammen SDO) eine zentrale Rolle spielen. Wie kann es sein, dass sich die Teams des Clusters **Im Fluss** durch eine hohe SDO-Leistung auszeichnen, ohne dass ein starker Fokus auf die Dokumentation gesetzt wird? Zum einen führen viele Strategien zu einer hohen SDO-Leistung, die das Augenmerk nicht auf die Dokumentation legen. Zum anderen refaktorieren die Teams des Clusters **Im Fluss** möglicherweise kontinuierlich ihren Code, um auf einen selbst dokumentierenden Prozess hinzuarbeiten, sodass weniger der Dokumente erforderlich sind, die wir in unserer Umfrage beschrieben haben.

Der Cluster **Abflauend**, auf den der größte Anteil der Teilnehmenden entfällt, setzt sich tendenziell aus Mitarbeitenden größerer Organisationen zusammen, die dazu tendieren, weniger cloudnativ zu sein als die anderer Cluster. Stellen Sie sich hier eine alteingesessene Firma vor, die auf einige eingerostete Prozesse setzt, es aber damit letzten Endes trotzdem schafft, ihren Endnutzerinnen und -nutzern eine stabile, zuverlässige und wertvolle Erfahrung zu bieten. Teams dieses Clusters haben eine leistungsorientierte, generative Firmenkultur.<sup>2</sup> Eines der interessantesten Merkmale des Clusters „Abflauend“ ist, dass seine Teams einen niedrigen Durchsatz und eine sehr positive Arbeitskultur (was Westrum als „generative“ Arbeitskultur definiert) aufweisen. Diese Kombination ist unüblich. Es kommt öfter vor, dass sich der Durchsatz und die Arbeitskultur proportional zueinander verhalten (hoch/positiv oder gering/negativ). Wir hoffen, dass wir in Zukunft weitere Untersuchungen vornehmen können, um die Beziehung zwischen Durchsatz und Arbeitskultur besser zu verstehen.

<sup>2</sup> (Westrum).



Zusätzlich haben wir noch untersucht, wie diese Cluster hinsichtlich dreier Ergebnisse abschneiden: Burnout, Unternehmensleistung und ungeplante Arbeit. Was wir herausgefunden haben, hat sich nicht mit unseren Erwartungen gedeckt. Teams des Clusters **Wird eingestellt** schnitten in Bezug auf die Firmenleistung besser ab als die anderen Cluster. Wenn wir uns die Merkmale dieses Clusters genauer ansehen (geringe Stabilität und geringer Durchsatz), steht das scheinbar im Widerspruch zu den meisten bisherigen Entdeckungen des DORA-Teams. Statt das jedoch als Anomalie abzutun, die der Zufälligkeit geschuldet ist (was durchaus denkbar wäre), möchten wir hier einige mögliche Erklärungen beleuchten.

Beim Versuch, diese Ergebnisse zu verstehen, sollten wir allerdings eine wichtige komplementäre Entdeckung im Hinterkopf behalten. Der Cluster **Wird eingestellt** zeichnet sich durch eine starke Unternehmensleistung aus, für die aber ein hoher Preis gezahlt wird: Diese Teams weisen die meisten Burnout-Fälle auf. Darüber hinaus haben sie den Eindruck, dass bei ihnen die größte Gefahr besteht, Fehler zu machen. Zusätzlich ächzen sie unter der größten Menge an ungeplanter Arbeit. Zusammengenommen lässt sich aus diesen Ergebnissen ableiten, dass die Zuverlässigkeit möglicherweise ausreicht, um eine hohe Unternehmensleistung zu erzielen. Aber wenn Geschwindigkeit und Stabilität fehlen, zahlt Ihr Team einen hohen Preis, da es mit Burnouts und ungeplanter Arbeit zu kämpfen hat.

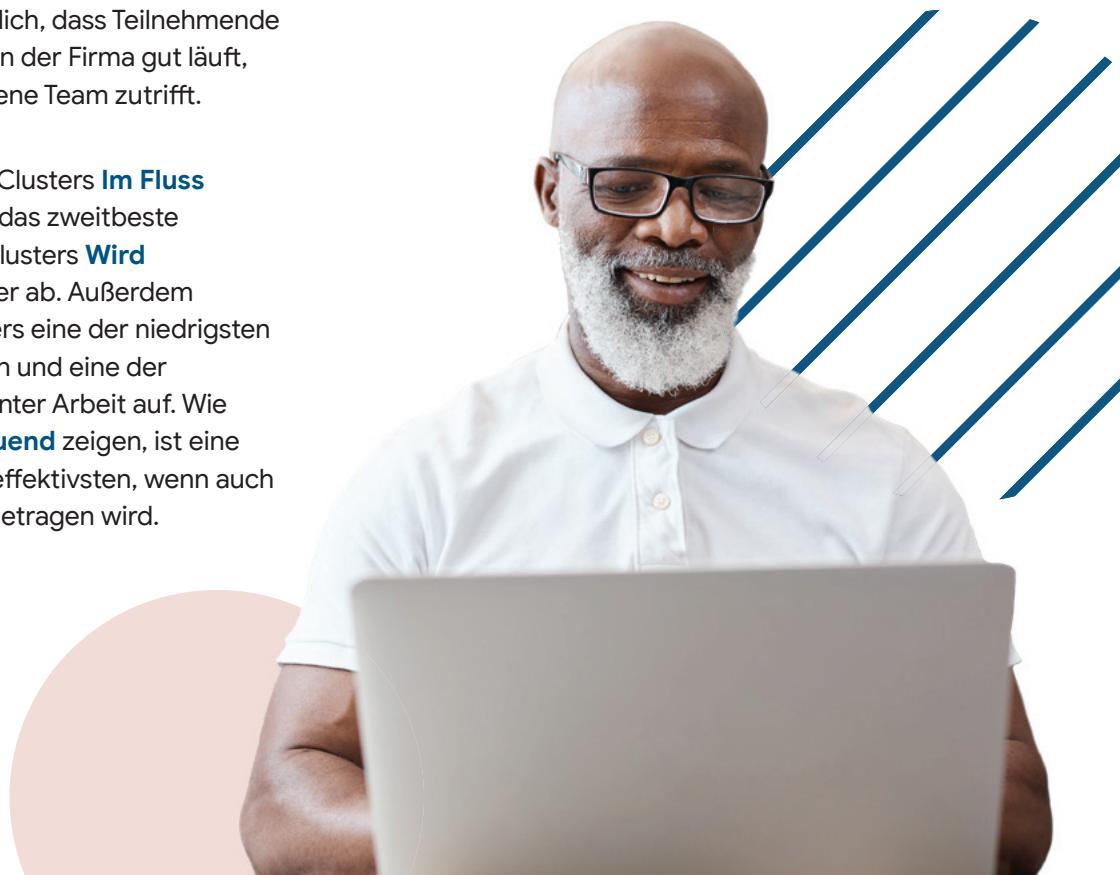
Wir haben noch weitere Hypothesen aufgestellt, um zu erklären, warum der Cluster **Wird eingestellt** eine höhere Unternehmensleistung aufweist, als die anderen Cluster, insbesondere als der Cluster **Im Fluss**. Nachfolgend finden Sie eine kurze Liste.

- Es spielen Features eine Rolle, durch die sich diese vier Cluster auszeichnen, die aber in unseren Daten möglicherweise nicht vorkommen. Beispielsweise könnte eine entsprechende Firmengröße eine unzureichende Reife gut genug ausgleichen. Der Cluster **Im Fluss** enthält tendenziell eher Teams aus kleineren Firmen, was anzeigen könnte, dass sich deren Produkte noch in prägenderen Phasen befinden.
- Die Mitglieder des Clusters **Im Fluss** neigen dazu, für kleinere Firmen zu arbeiten, die möglicherweise weniger an bisherige Prozesse sowie eine vorhandene Infrastruktur gebunden sind und somit ausgefeilte DevOps-Prozesse haben. Dennoch zeigen die Daten, dass die Größe einer Organisation positiv mit der Firmenleistung korreliert – und das aus Gründen, die möglicherweise kaum etwas mit Technologie zu tun haben.
- Die Teams des Clusters **Im Fluss** tendieren dazu, die Grundsätze zu missachten, die in der generativen Firmenkultur nach Westrum beschrieben sind. Wir haben herausgefunden, dass sich das oft nachteilig auf die Unternehmensleistung auswirkt.
- Die Organisationen in den verschiedenen Clustern unterscheiden sich womöglich darin, wie sie Zuverlässigkeit definieren und wie sie diese überwachen. Das Gleiche gilt für die Definition ihrer Ziele in puncto Organisationsleistung.

- Die Teams des Clusters **Wird eingestellt** weisen zwar auf kurze Sicht eine hohe Unternehmensleistung auf, aber wir fragen uns, wie sie langfristig abschneiden würden. Führen hier Burnout-Erkrankungen zu einer hohen Fluktuation? Schaffen sie es, ihre Prozesse zu skalieren?
- Wir stellen unsere Fragen auf verschiedenen Ebenen. Die Fragen zu den technischen Fähigkeiten (z. B. eine lose gekoppelte Architektur) werden auf Teamebene gestellt, die Fragen zur Unternehmensleistung dagegen auf Organisationsebene. Organisationen haben oft viele Teams und es ist möglich, dass Teilnehmende anerkennen, dass es zwar in der Firma gut läuft, dies aber nicht auf das eigene Team zutrifft.

Ferner erzielten die Teams des Clusters **Im Fluss** hinsichtlich der Firmenleistung das zweitbeste Ergebnis – nur die Teams des Clusters **Wird eingestellt** schnitten hier besser ab. Außerdem weisen die Teams dieses Clusters eine der niedrigsten Raten an Burnout-Erkrankungen und eine der geringsten Mengen an ungeplanter Arbeit auf. Wie die Cluster **Im Fluss** und **Abflauend** zeigen, ist eine DevOps-Philosophie dann am effektivsten, wenn auch der Zuverlässigkeit Rechnung getragen wird.

Wir freuen uns darauf, weiter nach Möglichkeiten zu suchen, Abweichungen innerhalb der Branche zu beschreiben. In Zukunft möchten wir die operative Leistung weiterhin als eine für unsere Verständnis dieser Abweichungen relevante Größe berücksichtigen. Darüber hinaus möchten wir stark normative und bewertende Cluster vermeiden (z. B. den Elite-Cluster) und uns stattdessen darauf konzentrieren, einfach nur beschreibend zu arbeiten und so gängige Konstellationen der SDO-Leistung zu ermitteln.



# 03

## Wie können Sie sich verbessern?



Derek DeBellis

### Wie können Sie sich in Bezug auf mehrere unterschiedliche Ergebnisse verbessern?

Sinn und Zwecke des Berichts „State of DevOps“ ist es, dass Sie und Ihr Team einen evidenzbasierten Leitfaden erhalten, damit sich Ihr Team leichter auf die DevOps-Praktiken und -Fähigkeiten konzentrieren kann, mit denen es die Ergebnisse erreichen kann, die Ihnen wichtig sind. Dieses Jahr haben wir in unsere Nachforschungen auch die Sicherheit und die Ergebnisse, die sich Teams wünschen, einbezogen. Früher haben wir uns auf die Ergebnisse der Softwarebereitstellungs- und operativen Leistung (SDO) und die Firmenleistung fokussiert. Das tun wir immer noch. Wir wollten aber auch Themen wie Burnout, ungeplante Arbeit, Fehleranfälligkeit und die Wahrscheinlichkeit, dass Teams weiterempfohlen werden, erforschen – nicht nur um die SDO- und Firmenleistung zu verbessern, sondern auch um der Themen selbst willen. Infolgedessen haben wir dieses Jahr darauf geachtet, dass wir die Praktiken und Fähigkeiten explizit erwähnen, die sich anscheinend auf diese Ergebnisse auswirken.





Dieses Jahr haben wir unser Forschungsmodell angepasst, um einer Theorie besser Rechnung zu tragen, die DORA zugrunde liegt: Was DevOps angeht, gibt es keinen allgemeingültigen Ansatz. In der Praxis haben wir die Erfahrung gemacht, dass es notwendig ist, die Arbeitsbedingungen eines Teams zu kennen, um passende Empfehlungen aussprechen zu können. Eine Praxis, die für ein Team vorteilhaft ist, kann sich bei einem anderen Team nachteilig auswirken. Beispielsweise halten wir seit Langem an der Hypothese fest, dass sich technische Fähigkeiten (wie eine lose gekoppelte Architektur, die Entwicklung nach Baumschema, Versionsverwaltung und Continuous Integration) positiver auf die Leistung der Softwarebereitstellung auswirken, wenn auch Continuous Delivery eingesetzt wird. Dieses Jahr haben wir das sowie andere Wechselwirkungen explizit modelliert. Dabei zielen wir auf ein besseres Verständnis ab. Wir möchten nicht nur Antworten auf die Frage „Was wirkt sich auf was aus?“ erhalten, sondern auch die Frage „Unter welchen Bedingungen sind diese Effekte vorhanden, werden sie verstärkt oder abgemildert?“ beantworten. All diese Konditionalitäten zu verstehen, hat sich als schwieriges und kniffliges Unterfangen erwiesen, aber wir freuen uns, an dieser Stelle einige der ersten Erkenntnisse dazu mit Ihnen zu teilen.

Sie finden die Forschungsmodelle von diesem Jahr und den Vorjahren [auf unserer Website](#).

## Über die „Four Keys“ hinaus

Wie lassen sich mithilfe der DORA-Messwerte die Leistung von Entwicklung und Betrieb verbessern? Beim Unternehmen Liberty Mutual Insurance führt ein funktionsübergreifendes Team aus Softwareentwicklerinnen und -entwicklern regelmäßig eine Leistungsüberprüfung durch und verwendet dazu die vier „[Four Keys](#)“-Messwerte von DORA. Zum Beispiel hat uns Jenna Dailey, Senior Scrum Master bei Liberty Mutual, erzählt, dass dort ein spezielles Team die Forschungsergebnisse von DORA verwendet hat, um den anderen Mitarbeitenden dabei zu helfen, einen Engpass zu erkennen, zu einem testbasierten Entwicklungsansatz zu wechseln und die Gesamtleistung des Unternehmens zu verbessern.

Weitere Informationen dazu, wie Liberty Mutual Datenmaterial und DORA-Messwerte eingesetzt hat, um die Qualität und Bereitstellung ihrer Software zu verbessern, erhalten Sie in der aktuellen Ausgabe der [Tomorrow Talks](#) des Unternehmens.



Eric Maxwell

## Cloud

Wie auch in den letzten Jahren schon steigt der Einsatz von Cloud-Computing weiterhin kontinuierlich an. Tatsächlich liegt der Prozentsatz der Personen, die angegeben haben, dass sie eine oder mehrere öffentliche Clouds verwenden, inzwischen bei 76 % – gegenüber 56 % im Jahr 2021. Die Anzahl der Teilnehmenden, die angegeben haben, dass sie keinerlei Clouds verwenden, einschließlich der Personen, die angegeben haben, dass sie auch keine private Cloud nutzen, beläuft sich nur noch auf 10,5 % – im Vorjahr waren es noch 21 %. 26 % der befragten Personen verwenden aktuell mehrere öffentliche Clouds – im Vorjahr waren es nur 21 %; die Hybrid-Cloud-Nutzung hat dagegen um 25 % zugelegt und beträgt jetzt 42,5 %. Außerdem war bei der Verwendung privater Clouds ein kleiner Anstieg auf 32,5 % zu beobachten – gegenüber 29 % im letzten Berichtsjahr.

36 % 25 %

Zuwachs bei der  
Verwendung der  
öffentlichen Cloud

Zuwachs bei der  
Hybrid-Cloud-Nutzung



Wenn Cloud-Computing eingesetzt wird, wirkt sich das positiv auf die gesamte Firmenleistung aus. Bei den Teilnehmenden, die mit Clouds arbeiten, **war die Wahrscheinlichkeit**, dass sie ihre Ziele in puncto Firmenleistung übertreffen, um 14 % höher als bei den Kolleginnen und Kollegen, die keine Clouds einsetzen.

Wie die Untersuchungen der letzten Jahre bereits ergeben haben und auch dieser Bericht belegt, wirkt es sich positiv auf die Leistung der gesamten Firma aus, wenn Cloud-Computing genutzt wird. Bei den Teilnehmenden, die mit Clouds arbeiten, **war die Wahrscheinlichkeit**, dass sie ihre Ziele in puncto Firmenleistung übertreffen, um 14 % höher als bei den Kolleginnen und Kollegen, die keine Clouds einsetzen. Unsere Untersuchung zeigt, dass das Cloud-Computing Teams befähigt, sich in Bereichen wie Sicherheit und Zuverlässigkeit der Softwarelieferkette hervorzutun –, und *das sind die Dinge*, die die Betriebsleistung beflügeln.

Überraschenderweise zeigen die Ergebnisse, dass zwischen Nutzenden aller Cloud-Arten – öffentlicher, privater, Hybrid- und Multi-Clouds – und ihrer Änderungsfehlerquote ein negativer Zusammenhang besteht, also eine erhöhte Änderungsfehlerquote

vorliegt. Dazu müssen wir weitere Nachforschungen anstellen. Statt über die Gründe dafür zu spekulieren, haben wir uns vorgenommen, das im Rahmen zukünftiger Forschungsprojekte zu untersuchen. Bis auf ein paar Ausnahmen hat sich gezeigt, dass sich die Verwendung cloudnativer Anwendungen (Anwendungen, die ursprünglich für die Cloud entworfen und erstellt wurden) positiv auf alles auswirkt, was wir untersucht haben.

Der Einsatz einer Cloud-Computing-Plattform, sei es einer öffentlichen oder einer privaten, leistet einen positiven Beitrag zu den Ergebnissen einer Firma in Sachen Unternehmenskultur und Arbeitsumgebung, zum Beispiel eine generative Kultur, weniger Burnout-Fälle, mehr Stabilität und eine höhere Zufriedenheit unter den Beschäftigten. In Bezug auf diese kulturellen Ergebnisse schnitten Cloud-Nutzerinnen und -Nutzer um 16 % besser ab.

## Cloud-Nutzung nimmt weiterhin Jahr für Jahr zu

	2022	Prozentuale Änderung gegenüber 2021
Hybrid-Cloud	42,47 %	25 %
Öffentliche Cloud und/oder mehrere öffentliche Clouds	76,08 %	36 %
Private Cloud	32,55 %	12 %
Keine Cloud	10,55 %	-50 %



Die Verwendung von Hybrid- und Multi-Clouds sowie privater Clouds scheint sich **negativ** auf die Leistungskennzahlen der Softwarebereitstellung – MTTR, Vorlaufzeit für Änderungen und Bereitstellungshäufigkeit – auszuwirken, **es sei denn**, die Teilnehmenden schnitten gut ab, wenn es um die **Zuverlässigkeit** ging.

## Hybrid- und Multi-Cloud-Einsatz steigert Firmenleistung

Die Signale, dass sich der Einsatz von Hybrid- und mehreren öffentlichen Clouds positiv auf Firmen auswirkt, sind nach wie vor ausgeprägt. Unternehmen, die mehrere Clouds verwendeten, wiesen eine 1,4-mal höhere Firmenleistung auf als jene, die keine Clouds verwendeten. Jedoch scheint es sich negativ auf einige Leistungskennzahlen der Softwarebereitstellung auszuwirken – die durchschnittliche Zeit bis zur Problembehebung (Mean Time To Resolution, MTTR), die Vorlaufzeit für Änderungen und die Bereitstellungshäufigkeit –, wenn Hybrid- und Multi-Clouds sowie private Clouds zum Einsatz kommen, außer die Teilnehmenden schnitten gut ab, wenn es um die Zuverlässigkeit ging.

Diese Entdeckung liefert also einen weiteren Beweis dafür, wie wichtig eine widerstandsfähige SRE-Strategie ist und welche bedeutende Rolle die Zuverlässigkeit bei der Softwarebereitstellung spielt.

2021 haben wir die Teilnehmenden gefragt, was der Hauptgrund dafür ist, dass sie mehrere öffentliche Clouds nutzen. 2022 haben wir sie dagegen gefragt, welche Vorteile sie daraus ziehen, auf mehrere Cloud-Anbieter zu setzen. Verfügbarkeit war hier der am häufigsten genannte Vorteil. Das stimmt auch mit der Aufmerksamkeit und dem Fokus überein, die bzw. den die Branche dem Thema Zuverlässigkeit hat zukommen lassen, denn nur wenn Dienste verfügbar sind, können diese auch zuverlässig sein. Über 50 % der Fachkräfte haben angegeben, dass sie auf die einzigartigen Vorteile verschiedener Cloud-Anbieter zurückgreifen.

## Realisierte Vorteile dank mehrerer Cloud-Anbieter

Verfügbarkeit	62,61 %
Individuelle Vorteile jedes Anbieters werden genutzt	51,59 %
Vertrauen wird auf mehrere Anbieter aufgeteilt	47,54 %
Notfallwiederherstellung	43,48 %
Einhaltung gesetzlicher Vorschriften	37,97 %
Verhandlungstaktik oder Beschaffungsanforderung	19,13 %
Sonstiges	4,06 %



> 50 %

der Teilnehmenden haben angegeben, dass sie mehrere Cloud-Anbieter nutzen.

Wenn Sie Ihre Firmenleistung erhöhen möchten, sollten Sie unbedingt die fünf Merkmale des Cloud-Computings verwenden. Am Ende einer sich anschließenden langen Kausalkette haben Sie dann die Firmenleistung verbessert.

## Die fünf Merkmale des Cloud-Computings

Wie auch bei unserem vorherigen Forschungsansatz haben wir nicht nur versucht, herauszufinden, ob die befragten Personen Cloud-Computing-Technologien verwenden, sondern auch, wie sie diese nutzen. Das haben wir erreicht, indem wir nach den fünf unverzichtbaren Merkmalen des Cloud-Computings gefragt haben, so wie sie das National Institute of Standards and Technology (NIST) definiert hat.

**On-Demand-Self-Service-Bereitstellung:** Nutzende können Rechenressourcen nach Bedarf automatisch bereitstellen, ohne menschliche Interaktion mit dem Anbieter.

**Breiter Netzwerkzugriff:** Die Ressourcen sind allgemein verfügbar und ein Zugriff ist über verschiedene Plattformen wie Smartphones, Tablets, Laptops und Workstations möglich.

**Ressourcen-Pooling:** Die Ressourcen des Anbieters werden in einem mandantenfähigen Modell zu Pools zusammengefasst, aus denen physische und virtuelle Ressourcen nach Bedarf dynamisch zugewiesen und neu zugewiesen werden.

Kundinnen und Kunden haben in der Regel keine direkte Kontrolle über den genauen Standort der bereitgestellten Ressourcen, können ihn jedoch auf einer höheren Abstraktionsebene angeben, z. B. Land, Bundesland oder Rechenzentrum.

**Schnelle Elastizität:** Ressourcen können elastisch und schnell nach Bedarf bereitgestellt (heraufskaliert) und freigegeben (herunterskaliert) werden. So hat es den Anschein, dass sie unbegrenzt sind und jederzeit in beliebiger Menge in Anspruch genommen werden können.

**Gemessener Dienst:** Cloud-Systeme kontrollieren und optimieren die Ressourcennutzung automatisch. Dazu wird eine Messfunktion auf einer Abstraktionsebene verwendet, die zum Diensttyp passt, z. B. Speicher, Verarbeitung, Bandbreite oder aktive Nutzerkonten. Die Ressourcennutzung kann mit dem Ziel der Transparenz überwacht, gesteuert und gemeldet werden.

Dieser Bericht bestätigt die Forschungsergebnisse von DORA der letzten drei Jahre, sodass wir schlussfolgern können, dass es sich positiv auf die Softwarebereitstellungs- und operative Leistung einer Firma auswirkt, wenn sie diese fünf Merkmale aufweist. Wir haben außerdem festgestellt, dass Firmen eine bessere Unternehmensleistung erzielen, wenn sie diese Merkmale aufweisen, da dadurch Prozesse angestoßen werden, die sich positiv auf das Unternehmen auswirken. Wenn Sie dafür sorgen, dass Ihr Unternehmen die fünf Merkmale des Cloud-Computings aufweist, ist das der erste Schritt in einem langwierigen Prozess, der am Ende zu einer höheren Unternehmensleistung führt.

2022 haben wir festgestellt, dass sich Firmenteams zunehmend die Alleinstellungsmerkmale des Cloud-Computings zunutze machen. Seit nunmehr vier Jahren in Folge stellen wir fest, dass die fünf Merkmale des Cloud-Computings zunehmend implementiert werden. Das Ressourcen-Pooling verzeichnete mit 14 % den größten Zuwachs, während die schnelle Elastizität, das Feature, das letztes Jahr am zweitmeisten verwendet wurde, mit 5 % das geringste Plus aufwies.



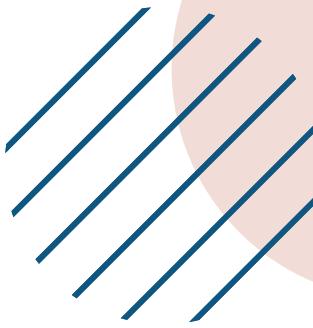
NIST	2021	2022	Prozentuale Änderung
Breiter Netzwerkzugriff	74 %	80 %	8
Schnelle Elastizität	77 %	81 %	5
On-Demand-Self-Service-Bereitstellung	73 %	78 %	7
Gemessener Dienst	78 %	83 %	7
Ressourcen-Pooling	73 %	83 %	14

14 %

Anstieg beim  
Ressourcen-Pooling



Dave Stanke



## SRE und DevOps

Ein erfolgreiches Technologieteam trägt mehr zu seiner Firma bei, als nur hochwertigen Code zu programmieren. Seine Mitglieder stellen auch sicher, dass die Dienste, die sie bereitstellen, verfügbar und leistungsstark bleiben und im Laufe der Zeit auch sonst den Erwartungen der Nutzenden entsprechen. Zuverlässigkeit ist ein vielseitiges Maß, das beschreibt, wie gut ein Team diesen Verpflichtungen nachkommt. Deswegen haben wir auch dieses Jahr die Zuverlässigkeit als Faktor bei der Softwarebereitstellung und im Betrieb untersucht.

Site Reliability Engineering (SRE) ist ein impulsgebendes Konzept für den laufenden Betrieb, das von Google entwickelt wurde und inzwischen von vielen Firmen eingesetzt wird. Bei SRE wird der Fokus auf empirisches Lernen, funktionsübergreifende Zusammenarbeit, umfangreiche Automatisierung und Messmethoden einschließlich Service Level Objectives (SLOs) gesetzt. Bei anderen modernen Praktiken für den laufenden Betrieb kommen ähnliche Methoden zum Einsatz, aber es werden andere Namenskonventionen verwendet. Um also das Ausmaß dieser Praktiken so objektiv wie möglich beurteilen zu können, haben wir bei unserer Umfrage darauf geachtet, in dem Text, den wir den Teilnehmenden vorgelegt haben, neutrale, beschreibende Formulierungen zu verwenden. Wir erheben auch Daten zu den Ergebnissen des Reliability Engineering: zum Umfang, indem Firmenteams in der

Lage sind, ihre Zuverlässigkeitsziele zu erreichen. Sowohl die Eingaben als auch die Ausgaben – Ergebnisse von SRE-Praktiken und Zuverlässigkeit – spiegeln sich in unserem Vorhersagemodell wider – zusammen mit anderen DevOps-Funktionen.

### Zuverlässigkeit ist unerlässlich

Unter den Teams, die wir befragt haben, sind die SRE-Praktiken weit verbreitet: Die Mehrheit der Teilnehmenden nutzt eine oder mehrere der Praktiken, nach denen wir gefragt haben. Diese Daten, die solch eine Bandbreite von Teams abbilden, lassen eine nuancierte Beziehung zwischen Zuverlässigkeit, Softwarebereitstellung und den Ergebnissen erkennen: Wenn die Zuverlässigkeit gering ist, erlaubt die Leistung der Softwarebereitstellung keinerlei Vorhersagen über den Unternehmenserfolg. Wenn die Zuverlässigkeit jedoch höher ist, kommt der positive Einfluss, den die Softwarebereitstellung auf den Unternehmenserfolg hat, langsam zum Tragen.

**Wenn es an Zuverlässigkeit fehlt, erlaubt die Softwarebereitstellungsleistung keinerlei Vorhersagen über den Unternehmenserfolg.**

**Wenn Sie in SRE-Maßnahmen investieren, verbessert sich die Zuverlässigkeit zwar, aber erst, nachdem in Bezug auf die Einführung ein Schwellenwert erreicht wurde.**

Dieses Phänomen ist mit der Verwendung des SRE-Fehlerbudget-Frameworks stimmig: Wenn ein Dienst unzuverlässig ist, ziehen dessen Nutzende keinen Vorteil daraus, wenn sie ihren Code schneller in so ein wackeliges Gerüst übertragen können.

Site Reliability Engineers machen schon lange geltend, dass die Zuverlässigkeit das wichtigste „Feature“ jedes Produktes ist. Unsere Forschung stützt die Beobachtung, dass es eine notwendige Bedingung darstellt, die Versprechen gegenüber den Nutzenden zu halten, damit das Unternehmen von einer verbesserten Softwarebereitstellung profitieren kann.

## J-Kurve berücksichtigen

Welche Herausforderungen erwarten Sie auf dem Weg zur Zuverlässigkeit? In ihrer O'Reilly-Publikation „Enterprise Roadmap to SRE“<sup>1</sup> haben die an der DORA-Umfrage beteiligten Mitarbeiter James Brookbank und Steve McGhee über ihre Erfahrungen reflektiert, die sie beim Implementieren von SRE in etablierten Organisationen gemacht haben. Sie haben ihrer Leserschaft auch empfohlen, „die J-Kurve von Veränderungen zu berücksichtigen“. Die „J-Kurve“, die bereits im Bericht „State of DevOps“ aus dem Jahr 2018 beschrieben wurde, ist ein Phänomen, bei dem betriebliche Transformationen tendenziell zu frühen Erfolgen führen, diese jedoch von ertragsarmen Zeiträumen oder sogar Regressionen

abgelöst werden. Die Unternehmen, die diese Herausforderungen überwinden, pendeln sich danach jedoch häufig auf einer (jetzt wieder) hohen Leistungsstufe ein, die auch von Dauer ist.

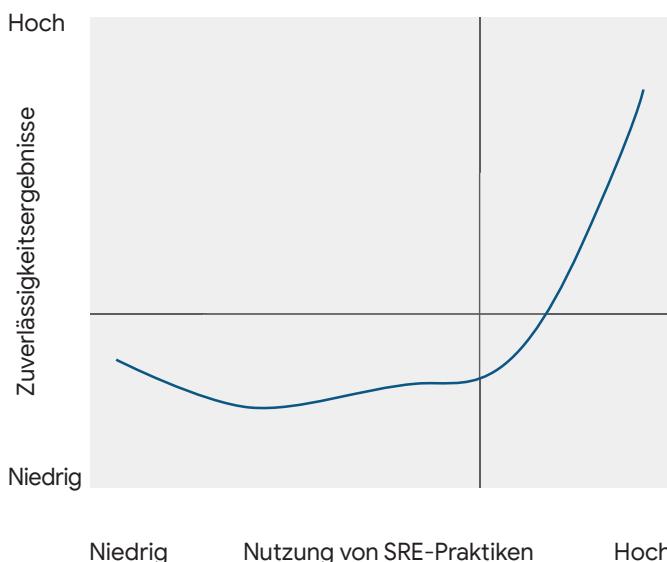
Unsere diesjährige Forschung zeigt ein J-Kurvenmuster, das sich über alle Technologieteams hinweg erstreckt, die wir untersucht haben: Wenn Teams weniger Reliability-Engineering-Praktiken einsetzen – was darauf hindeutet, dass sie noch am Anfang ihrer SRE-Reise stehen –, erlauben diese Praktiken keine Vorhersagen bezüglich besserer Zuverlässigkeitsergebnisse. Wenn Teams zunehmend mehr SRE-Praktiken einsetzen, kommen sie jedoch irgendwann an einen Wendepunkt, ab dem sich aus dem Einsatz von SRE-Praktiken in hohem Maße die Zuverlässigkeit und daraus wiederum die Unternehmensleistung prognostizieren lässt.



<sup>1</sup> <https://sre.google/resources/practices-and-processes/enterprise-roadmap-to-sre/>.

## Verlässliche Teams erstellen verlässliche Dienste: Eine generative Teamkultur erlaubt die Prognose einer höheren Zuverlässigkeit.

Teams, die bei der Einführung von SRE noch am Anfang stehen, sollten sich darauf einstellen, dass sie auf dieser Reise auch Rückschläge erleben werden. Es kann ein langer Weg sein, bis die Unternehmenskultur, die Prozesse und die Tools alle auf die neuen Grundprinzipien ausgerichtet sind. Aber die Teammitglieder können darauf vertrauen, dass ihr Unterfangen wahrscheinlich von Erfolg gekrönt sein wird, wenn sie genug Zeit mitbringen und kontinuierlich Investitionen leisten.



Teams, die die Anfangsphase der Einführung von SRE-Praktiken überstehen, weisen eine zunehmende Verbesserung bei Zuverlässigkeitsergebnissen auf

## In Menschen, Prozesse und Tools investieren

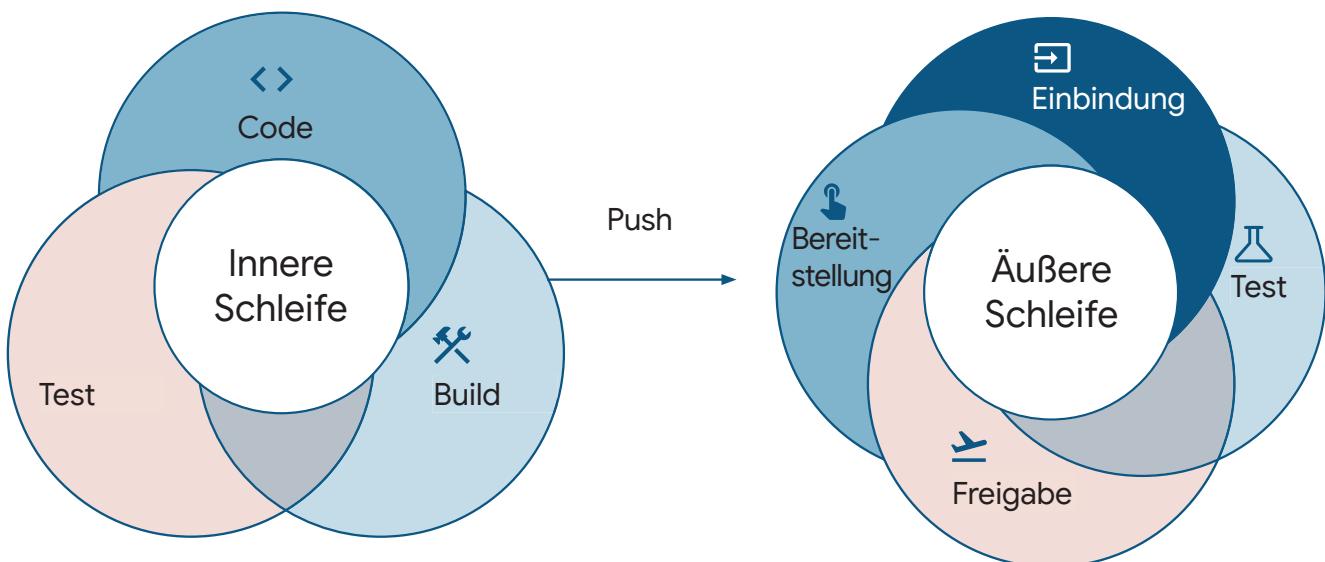
Um Zuverlässigkeit zu erreichen, braucht es Menschen und in vielerlei Hinsicht veranschaulicht das der SRE-Ansatz. Eines der Kernprinzipien von SRE ist es, dass nicht die internen Monitoring-Daten sondern die Wahrnehmung der Nutzenden der wahre Maßstab der Zuverlässigkeit ist. Daher überrascht es vielleicht kaum, dass die Zuverlässigkeit durch eine positive Teamdynamik befördert wird. Wir haben festgestellt, dass Teams mit einer „generativen“ Kultur, die also von Vertrauen und Zusammenarbeit geprägt ist, mit einer höheren Wahrscheinlichkeit SRE-Praktiken verwenden sowie gute Zuverlässigkeitsergebnisse erzielen. Auch stabile Teams mit denselben Mitgliedern über eine lange Zeit erzielen eine höhere Zuverlässigkeit, wenn es um Dienste geht, die für Nutzende sichtbar sind. Und wie auch bei DevOps im Allgemeinen wirkt es sich positiv auf die Reliability-Engineering-Bemühungen aus, wenn die Anstrengungen der Mitarbeitenden von Prozessen und Tools unterstützt werden. Praktiken wie das Verwenden von Cloud-Computing und Continuous Integration führen zu besseren Zuverlässigkeitsergebnissen.

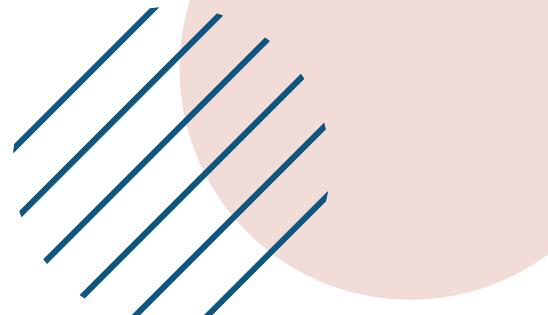


Eric Maxwell

## Technische DevOps-Fähigkeiten

Dieses Jahr haben wir eine Reihe technischer Fähigkeiten untersucht, um die Firmenergebnisse zu verstehen, die von verschiedenen technischen Praktiken beeinflusst werden. Wir haben zwei ausgeprägte Phasen der Softwareentwicklung betrachtet: die „innere Entwicklungsschleife“, die Entwickleraufgaben wie Coding, Tests und die Übergabe an die Versionsverwaltung umfasst, sowie die „äußere Entwicklungsschleife“, die Aktivitäten wie das Integrieren von Code, die automatisierte Code Review, die Testausführung, Bereitstellung und die Freigabe umfasst.





Bei leistungsstarken Mitarbeitenden, die ihre Zuverlässigkeitssziele erreichen, ist es **1,4-mal wahrscheinlicher, dass sie CI einsetzen.**

Unsere Studien haben ergeben, dass Firmen, die sich bei den Entwicklungsaufgaben der inneren und äußeren Schleife hervortun, Code schneller und zuverlässiger bereitstellen können. Die Fähigkeiten, die am meisten zu einer hohen Leistung beitragen, sind die Versionsverwaltung, Continuous Integration, Continuous Delivery und eine lose gekoppelte Architektur.

Bei leistungsstarken Beschäftigten, die ihre Zuverlässigkeitssziele erreichen, ist es:

**33 %**

wahrscheinlicher, dass sie die Versionsverwaltung einsetzen.

**39 %**

wahrscheinlicher, dass sie Continuous Integration verwenden.

**46 %**

wahrscheinlicher, dass sie auf Continuous Delivery setzen.

**40 %**

wahrscheinlicher, dass sie über Systeme verfügen, die auf einer lose gekoppelten Architektur basieren.

Tatsächlich weisen die Teilnehmenden, die die oben genannten Fähigkeiten überdurchschnittlich oft nutzen, eine **3,8-mal höhere Firmenleistung** auf, als diejenigen, die nicht auf diese technischen Fähigkeiten zurückgreifen.

## Continuous Integration

Continuous Integration, oft als CI bezeichnet, gehört zur äußeren Schleife des Entwicklungsprozesses, die nach jedem Code-Commit automatisch ein Artefakt erstellt und eine Reihe automatisierter Tests ausführt, um zu bewerten, ob der Code bereitgestellt werden kann. Dieser Prozess verschafft den Entwickelnden schnelles, automatisiertes Feedback, sodass sie ihrer Arbeit mit einem höheren Maß an Vertrauen nachgehen können. CI ist ein wesentlicher Bestandteil des Prozesses, Code vom Computer eines Entwickelnden in die Produktionsumgebung zu bringen. Wie schon in den Vorjahren hat sich auch diesmal gezeigt, dass CI die Bereitstellungsleistung ankurbelt. **Bei leistungsstarken Beschäftigten, die ihre Zuverlässigkeitssziele erreichen, ist es 1,4-mal wahrscheinlicher, dass sie CI einsetzen.**

Dieses Jahr haben wir den anderen Teil des Entwicklungsprozesses der äußeren Schleife ein bisschen genauer unter die Lupe genommen: Continuous Delivery. Dazu kommen wir in einem späteren Kapitel. Vorher sehen wir uns einen ergänzenden Bestandteil der Continuous Integration an: die Entwicklung nach Baumschema.

## Entwicklung nach Baumschema

Die Entwicklung nach Baumschema bedeutet, Code kontinuierlich in einem Hauptentwicklungszyklus zusammenzuführen und langlebige Feature-Zweige zu vermeiden. Diese Praxis wird als Ergänzung der Continuous Integration angesehen. Seit mehreren Jahren zeigt sich, dass sie die Geschwindigkeit der Softwarebereitstellung erhöht.

Dieses Jahr gab es eine veränderte demographische Zusammensetzung in Bezug auf die Erfahrung, die die Teilnehmenden aufwiesen. So hat sich gezeigt, dass die Erfahrung eine wichtige Rolle spielt, wenn es darum geht, die Entwicklung nach Baumschema zu implementieren. Letztes Jahr haben 40 % der Teilnehmenden angegeben, über mehr als 16 Jahre Erfahrung in ihrem Job zu verfügen. Dieses Jahr entfielen nur 13 % auf diese Kategorie. Im Rahmen der weiteren Bestätigung unserer These, dass die Bereitstellungsleistung von bestimmten Faktoren abhängt, haben wir herausgefunden, dass Personen, die insgesamt weniger Erfahrung haben, auch weniger positive Ergebnisse vorweisen können, wenn es um die Entwicklung nach Baumschema geht. Im Detail sehen wir folgendes Bild:

- ▼ **Reduzierte** Gesamtleistung in puncto Softwarebereitstellung
- ▲ **Erhöhtes** Maß an ungeplanter Arbeit
- ▲ **Erhöhte** Fehleranfälligkeit
- ▲ **Erhöhte** Änderungsfehlerquote

Fachkräfte mit mehr als 16 Jahren Erfahrung, die die Entwicklung nach Baumschema verwenden, ziehen aus der Praxis Vorteile. Hier stellen wir Folgendes fest:

- ▲ **Erhöhte** Gesamtleistung in puncto Softwarebereitstellung
- ▼ **Reduziertes** Maß an ungeplanter Arbeit
- ▼ **Reduzierte** Fehleranfälligkeit
- ▼ **Reduzierte** Änderungsfehlerquote

Dies liegt wahrscheinlich an den zusätzlichen Praktiken, die erforderlich sind, um die Entwicklung nach Baumschema erfolgreich zu implementieren. Gewissen Teams wird der Umstieg auf die Entwicklung nach Baumschema definitiv schwer fallen. Das gilt für Teams, die über keine rigoros durchgesetzten Regeln verfügen, die besagen, dass ein fehlerhafter Hauptentwicklungszyklus nie aufgegeben wird, oder die keine abgeschlossenen Codezweige verwenden und automatische Rollbacks für Code vornehmen, der dazu führt, dass der Code im Hauptentwicklungszyklus nicht mehr funktioniert.

Wenn allerdings mit der Entwicklung nach Baumschema gearbeitet wird, wirkt sich das positiv auf die Gesamtleistung des Unternehmens aus.





Frank Xu

## Continuous Delivery

Continuous Delivery (CD) ist eine Softwareentwicklungsmethode mit der:

1. ein Team befähigt werden kann, Software jederzeit in der Produktionsumgebung und für Endnutzerinnen und -nutzer bereitzustellen,
2. sichergestellt wird, dass die Software über ihren gesamten Lebenszyklus hinweg in einem bereitstellbaren Zustand ist – auch dann, wenn gerade an neuen Features gearbeitet wird,
3. ein schneller Feedback Loop erstellt wird, der es dem Team ermöglicht, die Qualität und Bereitstellungstauglichkeit des Systems zu prüfen und vorrangig die Probleme zu beheben, die den Bereitstellungsprozess behindern.

Beachten Sie, dass die Continuous Delivery nicht zwingend auch die kontinuierliche Bereitstellung beinhaltet, die Praxis, bei der jeder Software-Build automatisch bereitgestellt wird. Die Continuous Delivery erfordert nur, dass ein Software-Build jederzeit bereitgestellt werden kann.

Letztes Jahr haben wir die technischen DevOps-Ressourcen untersucht, die Vorhersagen darüber erlauben, wie wahrscheinlich es ist, dass ein Team CD einsetzt. Dabei haben wir herausgefunden, dass Faktoren wie eine lose gekoppelte Architektur und kontinuierliche Tests/eine kontinuierliche Integration zu den wichtigsten Indikatoren für den Einsatz von CD zählen. Dieses Jahr haben wir nicht nur die treibenden Faktoren für den Einsatz von CD beleuchtet, sondern auch analysiert und ermittelt, welche Effekte die CD für sich genommen auf die Bereitstellungsergebnisse hat und welche Effekte sie auf diese hat, wenn sie mit anderen DevOps-Funktionen zusammen eingesetzt wird.

### CD verbessert die Softwarebereitstellungsleistung

Ähnlich wie in den vorigen Jahren kamen wir auch diesmal zu dem Ergebnis, dass das Verwenden von CD ein Indikator für eine höhere Softwarebereitstellungsleistung ist. Dies gilt sowohl, wenn nur CD eingesetzt wird, als auch, wenn sie zusammen mit anderen DevOps-Funktionen verwendet wird. Bei Teams mit einer besseren CD-Bewertung ist es wahrscheinlicher, dass sie ihren Code häufiger in der Produktionsumgebung bereitstellen und dass die Vorlaufzeit für Änderungen und zum Wiederherstellen von Diensten kürzer ist.

**Teams, die die Versionsverwaltung mit Continuous Delivery kombinieren, haben eine 2,5-mal höhere Wahrscheinlichkeit für eine hohe Softwarebereitstellungsleistung als Teams, die sich nur auf eines davon fokussieren.**

Hinzu kommt, dass bei den befragten Personen die Wahrscheinlichkeit 2,5-mal höher ist, dass sie eine bessere Softwarebereitstellungsleistung melden, wenn ihr Team auch Praktiken der Versionsverwaltung einsetzt.

## **CD kann zu mehr ungeplanter Arbeit führen**

Die Daten weisen darauf hin, dass der Einsatz von Continuous Delivery dazu führt, dass Entwicklerinnen und Entwickler mehr Zeit mit Revisionen und ungeplanter Arbeit verbringen. Eine Hypothese für diese Entdeckung ist, dass Entwicklerinnen und Entwickler Anwendungen eher dann iterativ erstellen, wenn engere Feedback Loops vorhanden sind. Das hat zur Folge, dass sie manche iterativen Änderungen möglicherweise als ungeplante Arbeit am selben Teil des Systems betrachten. Diese Arbeit kann gleichzeitig ungeplant und durch Feedback aus einer vorherigen Bereitstellung entstanden sein.

## **Technische Best Practices und CD**

Unsere Forschung hat immer wieder ergeben, dass eine Vielzahl technischer Fähigkeiten CD unterstützt. Dieses Jahr haben wir untersucht, was passiert, wenn einige dieser Fähigkeiten zusammen mit CD eingesetzt werden. Wir haben dabei herausgefunden, dass es sich negativ auf die Leistung eines Teams auswirken kann, wenn die Entwicklung nach Baumschema und eine lose gekoppelte Architektur zusammen mit CD verwendet werden. Beispielsweise sehen wir Beweise, dass Teams, die lose gekoppelte Architekturen und CD gemeinsam verwenden, eine 43 % höhere Wahrscheinlichkeit für eine überdurchschnittliche Fehleranfälligkeit aufweisen (z. B. Produktausfälle, Sicherheitslücken oder eine deutliche Leistungsverschlechterung bei ihren Diensten), wenn sie mit Teams verglichen werden, die nur CD eingeführt haben. Diese Effekte müssen noch näher untersucht werden; sie deuten aber darauf hin, dass sich Teams, die daran arbeiten, besser zu machen, möglicherweise auf ein paar Stolpersteine einstellen müssen. Diese Stolpersteine können mit der J-Kurve der Transformation im Zusammenhang stehen, gemäß der Teams frühzeitig Verbesserungen erreichen, aber dann nachlassen, sobald sie die „niedrig hängenden Früchte gepflückt haben“. Es ist nötig, dass Sie sich ganz der Verbesserung verpflichten, wenn Sie Ihr Potenzial voll ausschöpfen möchten. Wenn Sie versuchen, eine Fähigkeit zu verbessern, z. B. CD, sollten Sie unbedingt ein Auge auf die Effekte haben, die dies auf das Team und die Gesamtleistung hat.



David Farley

## Lose gekoppelte Architektur

Lose gekoppelte Systeme sind für die Effektivität von Teams und Organisationen wichtig. Das gilt nicht nur für cloud- und mikrodienstbasierte Systeme – es geht um die Fähigkeit einer Organisation, Änderungen auf den Weg zu bringen. Die Mühelosigkeit, mit der eine Organisation sicher und selbstbewusst ihre Software verändern kann, ist ein Merkmal der Qualität dieser Software.

Mit einer lose gekoppelten Architektur können Teams:

- umfangreiche Änderungen am Design ihres Systems vornehmen, ohne darauf angewiesen zu sein, dass andere Teams Änderungen an ihren Systemen vornehmen,
- durch unabhängige On-demand-Tests schneller Feedback erhalten – bei geringeren Kosten für die interne Abstimmung,
- Code ohne nennenswerte Ausfallzeiten bereitstellen.

Im diesjährigen Bericht haben wir die Teilnehmenden gebeten, zu beschreiben, ob die Software, die sie erstellen, auf einer lose gekoppelten Architektur basiert oder nicht. Die Ergebnisse waren faszinierend. Sie spiegelten eine Vielfalt überwiegend positiver Assoziationen zwischen dem Vorhandensein einer lose gekoppelten Architektur und der Leistung der Teams über mehrere Größen hinweg wider.



## Die Vorteile einer lose gekoppelten Architektur

Teams, die sich darauf konzentrieren, Software mit lose gekoppelten Architekturen zu erstellen, sind besser aufgestellt, eine starke Leistung in puncto Stabilität, Zuverlässigkeit und Durchsatz zu erbringen. Bei diesen Teams ist auch die Wahrscheinlichkeit höher, dass deren Mitglieder ihren Arbeitsplatz befreundeten Personen, Kolleginnen und Kollegen empfehlen.

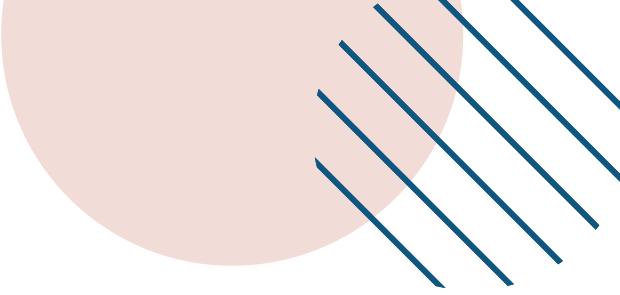
Software, die eine lose gekoppelte Architektur verwendet, wird oft von Teams entwickelt, die in der Cloud Bereitstellungen vornehmen, auf ein Mikrodienst-Architekturmuster setzen und Hunderte

von Diensten verwalten. Jedoch geht es bei der losen Kopplung um mehr als nur ein einfaches Maß der Anzahl der Dienste in einem System. Komponenten in einer lose gekoppelten Architektur können unabhängig voneinander bereitgestellt werden. Diese Unabhängigkeit ermöglicht es den Teams, ihre Dienste zu entwickeln, zu testen und bereitzustellen, ohne dass eine aufwändige und teure Koordination zwischen einzelnen Teams erforderlich ist.

In der realen Welt ist die lose Koppelung nicht auf einen Architekturstil beschränkt. Im Grunde geht es dabei nämlich um die Fähigkeit, eine Änderungen in einem Teil des Systems vorzunehmen, ohne dass sich diese auf andere Teile auswirkt. Dadurch können Organisationen ihre Arbeit aufteilen, sodass einzelne Teams Fortschritte machen können, ohne dass sie sich mit anderen Teams abstimmen müssen.

Teams, die sich darauf konzentrieren, Software mit lose gekoppelten Architekturen zu erstellen, sind besser aufgestellt, in puncto Stabilität, Zuverlässigkeit und Durchsatz gut abzuschneiden.

Unserer Erfahrung nach haben Teams noch keine lose Kopplung erreicht, wenn es weiterhin tiefgreifender Integrationstests mit anderen Diensten bedarf, bevor sie genug Vertrauen in ihre Software haben, um sie bereitzustellen. Um eine lose Kopplung zu erreichen, sollten diese Teams die Schnittstellen und die Isolation zwischen ihren Systemen verbessern. Eine effiziente Methode, die Schnittstellen und die Isolation zu verbessern, ist es, die „Testbarkeit“ von Diensten und Komponenten zu verbessern. Wenn es Ihr Design erlaubt, dass Sie einen Dienst isoliert testen, ist seine Schnittstelle definitionsgemäß lose gekoppelt.



Wir haben auch festgestellt, dass eingespielte, stabile Teams, die eine lose gekoppelte Architektur verwenden, eher solche Praktiken zum Entwickeln von Software einsetzen, durch die man zu kontinuierlichen Verbesserungen ermutigt und dahingehend unterstützt wird. Beispielsweise unterstützen SRE-Praktiken wie das Festlegen von Zuverlässigkeitszielen, um gewisse Arbeiten zu priorisieren, oder das Durchführen regelmäßiger Überprüfungen, um diese Ziele anhand der Belege zu überarbeiten, eine lose gekoppelte Architektur.

Bei lose gekoppelten Architekturen kann die Organisation auch leichter neue Mitarbeitende einstellen, da unabhängige Teams, die sich nicht mit anderen Teams abstimmen müssen, es einfacher haben, ihre Mitgliederzahl unabhängig zu erhöhen.

Kurz gesagt wirkt sich die lose Kopplung von Softwarediensten nicht nur auf technische Weise aus. Sie hat auch Auswirkungen auf die soziotechnischen Aspekte der Softwareentwicklung. Die Kopplung bildet den Grundstein des Gesetzes von Conway – der Idee, dass die Systemgestaltung einer Organisation deren Kommunikationsstruktur widerspiegelt. Mehr lose gekoppelte Systeme bedeuten mehr lose gekoppelte Organisationen mit einem verteilten, skalierbareren Entwicklungsansatz.

## Überraschende Entdeckungen

Unsere diesjährige Forschung brachte zutage, dass eine lose gekoppelte Architektur möglicherweise dazu beiträgt, dass Beschäftigte an Burnout erkranken. Das ist eine überraschende Erkenntnis, die im Widerspruch zu Ergebnissen aus vorigen Jahren steht. Unsere Analyse zeigt, dass stabile Teams, bei denen die Informationen ungehindert fließen, weniger Burnout-Fälle verzeichnen. Die generative Kultur und Teamstabilität nach Westrum unterstützen beide eine lose gekoppelte Architektur und führen zu weniger Burnout-Erkrankungen, also gibt es hier eindeutig Widersprüche. Es ist folglich noch weitere Forschung nötig, bevor wir diesbezüglich endgültige Schlüsse ziehen können.

Gleichwohl kann es für Teams schwieriger sein, ihre Software von anderen Teams zu entkoppeln, wenn die Sicherheitsanforderungen durch eine zusammengeführte Sicherheitsorganisation definiert und kontrolliert werden. Das liefert einen weiteren Beweis dafür, welche Vorteile es bringt, wenn jenes Team mit Sicherheitsbedenken betraut wird, das für die Anwendung hauptsächlich verantwortlich ist (siehe auch: [Warum die Sicherheit der Lieferkette zählt](#)). Das ist eine der subtileren Formen der Kopplung in Organisationen und obwohl wir die Daten bezüglich der Sicherheit erhoben haben, trifft das wahrscheinlich genau so auch für andere zentralisierte Funktionen zu. Wenn Sie Ihren Teams erlauben, hinsichtlich Sicherheit und anderer oft zentralisierter Funktionen eigene Entscheidungen zu treffen, ist das eine Möglichkeit, wie Sie sich dem Ziel nähern können, dass Ihre Firma von einer lose gekoppelten Architektur profitiert.





Daniella Villalba

## Firmenkultur

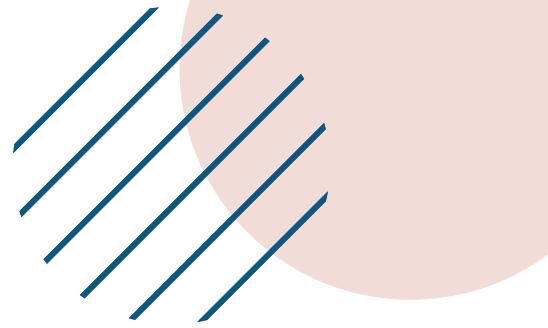
„Das haben wir immer schon so gemacht.“

Dieser Satz fiel bestimmt unzählige Male, als Mitarbeitende aus allen möglichen Branchen beschreiben sollten, wie ihre Organisation mit Herausforderungen und Chancen umgeht.

Jede Organisation hat ihre eigene Kultur und unsere Forschung hat immer wieder gezeigt, dass die Organisationskultur für den Erfolg einer Organisation und das Wohlergehen ihrer Belegschaft grundlegend ist.

Auch die Unternehmenskultur ist ein notwendiger Aspekt von DevOps, da es bei DevOps grundsätzlich um Folgendes geht: Tools, Best Practices und darum, **wie Menschen zusammenarbeiten**, um Software schnell, zuverlässig und sicher zu entwickeln und bereitzustellen. Wenn die Unternehmensführung die Faktoren versteht, die sich auf die Kultur des Unternehmens auswirken, kann das ihr dabei helfen, kulturbezogene Herausforderungen direkt anzugehen. Daher sollten es Firmen priorisieren, eine gesunde Unternehmenskultur zu schaffen. Wenn diese Herausforderungen nicht angegangen werden, kann das dazu führen, dass sich die DevOps-Praktiken nur schwer durchsetzen.





Dieses Jahr haben wir wiederum die Typologie der Unternehmenskultur von Westrum verwendet, um sie zu messen und einzuordnen. Zusätzlich haben wir unser Verständnis von Firmenkultur erweitert, indem wir auch Teamabwanderung, flexible Arbeits(zeit)modelle, wahrgenommene betriebliche Beteiligung und Burnout gemessen haben.

Das Datenmaterial aus diesem Jahr unterstützt frühere Erkenntnisse, die besagen, dass es sich auf die **Unternehmensleistung** auswirkt, welche Art von Kultur in einer Organisation herrscht. Insbesondere gibt es einen Zusammenhang zwischen einer **generativen Kultur** und einer höheren Unternehmensleistung – ganz im Gegensatz zu Firmen, die von einer bürokratischen oder pathologischen Kultur geprägt sind. Bei Beschäftigten

in Organisationen mit einer generativen Kultur ist es wahrscheinlicher, dass sie stabilen Teams angehören, eine qualitativ hochwertigere Dokumentation erstellen und den größten Teil ihrer Arbeitszeit einer sinnvollen Beschäftigung nachgehen.

## Teamabwanderung

Wir haben diesmal die Teamabwanderung untersucht. Dabei haben wir herausgefunden, dass sich **stabile Teams** – Teams, deren Zusammensetzung sich in den letzten 12 Monaten kaum verändert hat – eher in leistungsstarken Organisationen befinden. Ein ständiger Wechsel kann sich auf die Produktivität und Arbeitsmoral auswirken, da neue Teammitglieder Zeit brauchen, um eingearbeitet zu werden. Und die verbliebenen Beschäftigten müssen sich möglicherweise an eine

## Unternehmenskultur nach Westrum

Pathologisch Machtorientiert	Bürokratisch Regelorientiert	Generativ Leistungsorientiert
Wenig Zusammenarbeit	Mäßige Zusammenarbeit	Intensive Zusammenarbeit
Personen, die eine Nachricht überbringen, werden „mundtot“ gemacht	Personen, die eine Nachricht überbringen, werden ignoriert	Personen, die eine Nachricht überbringen, werden gefördert
Verantwortung wird weggeschoben	Geringer Verantwortungsspielraum	Geteilte Verantwortung
Abteilungsübergreifendes Handeln ist unerwünscht	Abteilungsübergreifendes Handeln wird toleriert	Abteilungsübergreifendes Handeln wird gefördert
Fehler führen zur Schuldfrage	Fehler führen zu Verurteilung	Fehler führen zu Untersuchungen
Neues wird unterdrückt	Neues führt zu Problemen	Neues wird umgesetzt



veränderte Arbeitslast und Teamdynamik anpassen. Außerdem ergab unsere Forschung, dass stabile Teams eher angeben, dass sie eine hochwertige Dokumentation erstellen, als Teams, bei denen es zu mehr Fluktuation kommt. Einem Team, das einen ständigen Wandel bewältigen muss, kann es schwerer fallen, die Praktiken, die die Basis einer hochwertigen Dokumentation bilden, vollständig umzusetzen.

**Leistungsstarke Organisationen haben eher flexible Arbeits(zeit)modelle.**

## Flexible Arbeits(zeit)modelle

Da seit Beginn der Coronapandemie viele Organisationen flexible Arbeits(zeit)modelle eingeführt haben, haben wir untersucht, ob die Firmenleistung höher ist, wenn die Mitarbeitenden zwischen Telearbeit,

Arbeit vor Ort und einer Kombination von beidem wählen können. Aus unserem Datenmaterial geht hervor, dass Organisationen mit größerer **Mitarbeiterflexibilität** eine höhere Unternehmensleistung als Organisationen mit starreren Arbeitsmodellen aufweisen. Diese Ergebnisse belegen, dass es einer Organisation spürbare und unmittelbare Vorteile bringt, wenn die Mitarbeitenden ihre Arbeits(zeit)modelle nach Bedarf anpassen dürfen.

## Burnout

Burnout ist ein arbeitsbezogenes Gefühl von Furcht, Gleichgültigkeit und Zynismus. Wenn Menschen an Burnout erkranken, sind sie nicht nur unmotiviert und erschöpft, es ist auch wahrscheinlicher, dass sie mit ihrer Arbeitsstelle unzufriedener sind, was zu einer höheren Fluktuation führen kann. Burnout-Erkrankungen wurden bereits mit einer Vielzahl von negativen Auswirkungen auf die psychische und körperliche Gesundheit in Zusammenhang gebracht, wie einem erhöhten Risiko für Depressionen und Angsterkrankungen sowie für Herzkrankheiten und das Auftreten von Suizidgefährdungen<sup>1</sup>.

Letztes Jahr haben wir Burnout vor dem Hintergrund der Coronapandemie gemessen und herausgefunden, dass eine generative Kultur mit weniger Burnout-Erkrankungen der Mitarbeitenden in Verbindung steht.

<sup>1</sup> Maslach C, Leiter MP. Understanding the burnout experience: recent research and its implications for psychiatry. World Psychiatry. 2016 Jun;15(2):103–11. doi: 10.1002/wps.20311. PMID: 27265691; PMCID: PMC4911781.

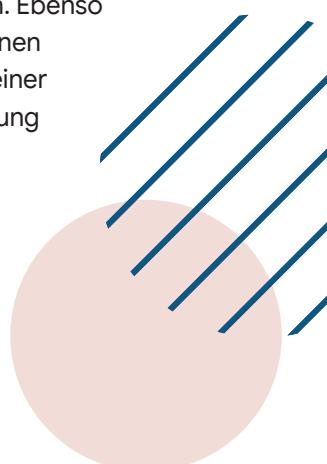
**Flexible Arbeitsmodelle stehen im Zusammenhang mit weniger Burnout-Erkrankungen in der Belegschaft und einer höheren Wahrscheinlichkeit, dass Mitarbeitende ihr Team weiterempfehlen.**

Dieses Jahr haben wir diese Feststellung reproduziert und unser Verständnis von Burnout erweitert, indem wir gezeigt haben, dass stabile Teams und flexible Arbeits(zeit)modelle ebenfalls mit einer geringeren Anzahl an Burnout-Erkrankungen im Zusammenhang stehen. Außerdem haben wir dieses Jahr den Net Promoter Score (NPS) für Teams gemessen, der angibt, ob angestellte Personen ihr Team befreundeten oder mitarbeitenden Personen empfehlen würden. Dabei kam heraus, dass der Team-NPS mit der wahrgenommenen Beteiligung der Führungsebene korreliert. Ähnlich wie beim Thema Burnout haben wir herausgefunden, dass bei einer generativen Kultur, einem stabilen Team und einem flexiblen Arbeits(zeit)modell Beschäftigte ihr Team eher an andere weiterempfehlen.

## Wie Mitarbeitende ihre Organisation wahrnehmen

Außerdem haben wir die wahrgenommene Beteiligung der Führungsebene untersucht, indem wir die Teilnehmenden gebeten haben, zu prognostizieren, wie viel Unterstützung sie für ihr Team in den nächsten 12 Monaten erwarten. Die Ergebnisse zeigten, dass eine höhere wahrgenommene Beteiligung der Führungsebene (zum Beispiel größere finanzielle Unterstützung, verstärkte Zuteilung von Ressourcen, Führungskräftepatenschaften) im Zusammenhang mit leistungsstarken Organisationen steht.

Wir haben die Teilnehmenden auch gebeten, vorherzusagen, wie wahrscheinlich es ist, dass es in den nächsten 12 Monaten zu einem Sicherheitsverstoß oder Kompletttausfall kommt. Die Ergebnisse haben gezeigt, dass Beschäftigte, die für leistungsstarke Organisationen arbeiten, weniger damit rechnen, dass in den nächsten Monaten ein schwerer Fehler auftritt – sie blicken positiver in die Zukunft ihrer Organisation. Ebenso haben wir festgestellt, dass Mitarbeiterinnen und Mitarbeiter von Organisationen mit einer hohen Software- und Bereitstellungsleistung **seltener** der Meinung waren, dass ihre aktuellen Praktiken geändert werden müssen, um bessere Geschäftsergebnisse zu erzielen.



## Ein paar Worte zur Vertretung verschiedener Gruppen

Aus unserer Untersuchung ging hervor, dass Mitarbeitende aus unterrepräsentierten Gruppen eher berichten, dass sie mehr Zeit mit ungeplanter Arbeit verbringen – egal ob sie für leistungsstarke oder leistungsschwache Organisationen arbeiten. Außerdem haben wir festgestellt, dass es bei Mitarbeitenden aus unterrepräsentierten Gruppen zu mehr Burnout-Fällen kommt, als bei Beschäftigten, die keiner unterrepräsentierten Gruppe angehören. Teamleiterinnen und -leiter sollten sich des Risikos bewusst sein, dass die Arbeitslast ungleich verteilt ist, und sicherstellen, dass dies nicht so ist.

Zusammengekommen unterstreichen diese Erkenntnisse, wie wichtig es ist, für die Belegschaft sowohl auf Organisations- wie auch auf Teamebene eine gesunde und inklusive Umgebung zu schaffen.

Wir heben zwar weiterhin hervor, dass die Kultur von Unternehmen wichtig ist, uns ist aber auch klar, dass es keine leichte Aufgabe darstellt, die Kultur einer Firma zu ändern oder sogar zu verbessern. Unsere Empfehlung lautet, dass sich Organisationen zuerst bemühen sollten, zu verstehen, welche Erfahrungen ihre Mitarbeitenden machen. Erst im zweiten Schritt sollten sie in entsprechende Ressourcen investieren, um kulturbezogene Probleme im Rahmen von DevOps-Transformationen anzugehen.



# 04

## Warum die Sicherheit der Lieferkette zählt



John Speed Meyers



Todd Kulesza

Im November 2020 hatten nur relativ wenig Technologiefachleute die Befürchtung, dass sich in Bezug auf die Softwarelieferkette eine Sicherheitskrise anbahnte. Damals war die [Open Source Security Foundation](#) gegründet worden, die an die Stelle bisheriger Bemühungen trat, um die Sicherheit von Open-Source-Software in den Fokus zu rücken. Zwar gab es damals ein paar [Schimmer der Hoffnung](#), dass dieses Problem angegangen wird, dennoch landete das Thema nicht auf den Titelseiten der großen Zeitungen. Mit einem Großangriff auf [SolarWinds](#) hat sich dann das Blatt

gewendet. Aber die Zeiten ändern sich schnell, wenn es angreifenden Personen gelingt, mit Trojanern verseuchte Softwareupdates einzusetzen, um unbemerkt tausende große Unternehmen und Regierungsnetzwerke auszuspionieren.

Heutzutage herrscht breiter Konsens, dass die Sicherheit der Softwarelieferkette ein dringliches Thema ist – vielleicht nicht unter Privatpersonen, aber definitiv bei den Mitarbeitenden der Geschäftsführung.



Diesbezüglich gibt es zahlreiche Initiativen und weite Teile der Softwarebranche haben sich verpflichtet, die Sicherheitspraktiken, die sie bei ihrer Softwarelieferkette anwenden, zu überarbeiten und die Sicherheit des Open-Source-Gemeinguts zu verbessern.

In diesem Kapitel konzentrieren wir uns auf zwei Initiativen: Die Supply Chain Levels for Software Artifacts ([SLSA](#), Aussprache: „Salsa“) und das Secure Software Development Framework ([SSDF](#)) des NIST. Beide bieten eine Reihe von Schutzmaßnahmen an. Mit diesen lässt sich sicherstellen, dass angreifende Personen keine Softwarereproduktionsprozesse manipulieren und auch keine Netzwerkschutzmaßnahmen überlisten können, indem sie schädliche Softwareupdates einsetzen.

Aber wie weit verbreitet sind die Sicherheitspraktiken aus SLSA und SSDF für die Softwarelieferkette? Welche Praktiken sind bereits weit verbreitet und bei welchen brauchen Firmen Unterstützung, damit diese mehr eingesetzt werden? Bis dato konnten wir keine systematischen Antworten auf diese Fragen finden. Die Befragung hunderter Softwarefachkräfte zu ihrer Verwendung von Praktiken hinsichtlich der Sicherheit der Lieferkette hat uns aber einige erste Antworten geliefert. Dabei sind vor allem vier Ergebnisse hervorzuheben:

**01 Die Einführung hat bereits begonnen:**  
Die Sicherheitspraktiken für die Softwarelieferkette, die in SLSA und SSDF zum Ausdruck gebracht wurden, werden bereits in gewissem Maße eingesetzt – aber da ist noch viel Luft nach oben.

**02 Gesündere Firmenkulturen verleihen einen Vorsprung:** Die Firmenkultur ist eine der wichtigsten Antriebskräfte für Sicherheitspraktiken in der Softwareentwicklung. In Firmen mit einer von hohem Vertrauen geprägten Kultur mit wenig Schuldzuweisungen werden eher SLSA- und SSDF-Praktiken eingeführt als in Firmen, deren Kultur von geringerem Vertrauen geprägt ist.

**03 Es gibt einen Hauptintegrationsbereich:**  
Die Anwendung der technischen Aspekte der Softwarelieferkettensicherheit scheint davon abzuhängen, ob CI/CD zum Einsatz kommt, da über die zugehörigen Plattformen oft viele Sicherheitspraktiken der Lieferkette eingebunden werden können.

**04 Sie bieten unerwartete Vorteile:**  
Abgesehen von geringeren Sicherheitsrisiken bringen bessere Sicherheitspraktiken noch weitere Vorteile wie weniger Burnout-Fälle mit sich.

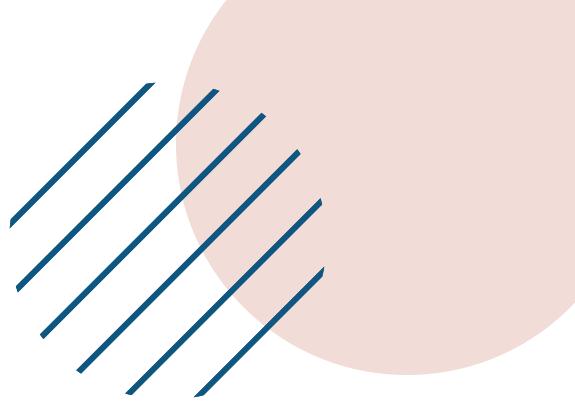
# Was Firmen heutzutage unternehmen, um Sicherheitslücken zu vermeiden

Damit wir besser verstehen, was Organisationen heutzutage tun, um bei der von ihnen erstellten Software Sicherheitslücken zu erkennen und diese zu beheben, haben wir dem diesjährigen Fragebogen mehr als zwei Dutzend neue Fragen hinzugefügt. Diese Fragen ließen sich weitgehend zwei Kategorien zuordnen:

- Fragen, bei denen die Teilnehmenden gebeten wurden, eine Aussage zu bejahen oder zu verneinen (zum Beispiel: „Meine Organisation verfügt über eine wirkungsvolle Methode, um Sicherheitsbedrohungen abzuwehren“ oder „Ich habe Zugriff auf die notwendigen Tools, um Sicherheitstests auszuführen“).
- Fragen dazu, ob und inwieweit Sicherheitspraktiken in der Organisation der befragten Person bestehen oder nicht (zum Beispiel: „Builds werden ausschließlich durch Build-Skripts definiert“ oder „Produktionsfreigaben werden mithilfe eines zentralisierten CI/CD-Systems erstellt und nie auf dem Computer einer entwickelnden Person“). Wir verwendeten hier den Maßstab „besteht/besteht nicht“, da wir in frühen Tests festgestellt hatten, dass Teilnehmende einige sicherheitsbezogene Fragen bereits voreingenommen bejahten. Bei den Fragen zu SSDF haben wir jedoch wieder die Antwortskala „zustimmen/nicht zustimmen“ verwendet.

Das **SLSA-Framework**, bei Redaktionsschluss in Version 0.1, bietet eine Reihe von Praktiken, die dazu dienen, die Integrität der Softwarelieferkette sicherzustellen. Ihnen werden SLSA-„Stufen“ zugeordnet, wobei eine höhere Stufe einem höheren Maß an Softwarelieferkettensicherheit entspricht. Wir haben die Teilnehmenden zu vielen der SLSA-Praktiken befragt. Insbesondere haben wir gefragt: „Wie etabliert sind die folgenden Praktiken für die Anwendung bzw. den Dienst, an der bzw. dem Sie hauptsächlich arbeiten?“ In Tabelle 1 sind die SLSA-bezogenen Praktiken aufgeführt, die in der Befragung berücksichtigt wurden.

Das **SSDF** (aktuelle Version: 1.1) legt den Fokus auf Praktiken, die es Firmen ermöglichen, Software mit weniger Sicherheitslücken auszuliefern und die möglichen Auswirkungen eventuell verbliebener Sicherheitslücken zu minimieren. Statt mit „Stufen“ wird hier mit vier Kategorien gearbeitet, in denen die SSDF-Praktiken organisiert werden: Vorbereiten der Organisation, Schützen der Software während der Entwicklung, Erstellen von gut gesicherter Software und effektives Reagieren bei erkannten Sicherheitslücken. Bei der Befragung wurden die Teilnehmenden gebeten, anzugeben, wie sehr sie gewissen Aussagen zustimmen bzw. nicht zustimmen, die verschiedene SSDF-Praktiken beschreiben. Diese Fragen sind in Tabelle 2 aufgeführt.



SLSA-Praktik	Definition in der Befragung
Zentrales CI/CD-System	Produktionsfreigaben werden mithilfe eines zentralisierten CI/CD-Systems erstellt und nie auf dem PC einer entwickelnden Person.
Verlauf wird gespeichert	Überarbeitungen und ihr Änderungsverlauf werden unbefristet gespeichert.
Build-Skript	Builds werden ausschließlich durch das jeweilige Build-Skript vollständig definiert.
Isolierung	Builds sind voneinander isoliert; sie können keine gleichzeitigen und nachfolgenden Builds beeinträchtigen.
Build-Textdateien	Build-Definitionen und -Konfigurationen sind in Textdateien definiert, die in einem Versionsverwaltungssystem gespeichert werden.
Parameter-Metadaten	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) zu einem Artefakt enthalten alle Build-Parameter.
Abhängigkeiten-Metadaten	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) zu einem Artefakt dokumentieren alle Build-Abhängigkeiten.
Generierte Metadaten	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) werden entweder vom Build-Dienst oder von einem Build-Metadatengenerator erstellt, der den Build-Dienst liest.
Eingaben werden nicht geladen	Wenn Builds ausgeführt werden, wird verhindert, dass Build-Schritte Build-Eingaben dynamisch laden (d. h. alle nötigen Quellen und Abhängigkeiten werden vorab abgerufen).
Keine Bearbeitung durch Nutzende	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) zu einem Artefakt können nicht von Nutzenden der Build-Dienste bearbeitet werden.
Metadaten sind verfügbar	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) stehen den Personen zur Verfügung, die sie brauchen, z. B. in einer zentralen Datenbank, und werden in einem von diesen akzeptierten Format bereitgestellt.
Zwei-Personen-Überprüfung	Jede Änderung aus dem Verlauf einer Überarbeitung muss von zwei vertrauenswürdigen Personen individuell geprüft und genehmigt werden, bevor sie eingereicht wird.
Metadaten werden signiert	Die Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) dazu, wie ein Artefakt erstellt wurde, werden von meinem Build-Dienst signiert.

**Tabelle 1. SLSA-bezogene Fragen**

Hinweis: Die Teilnehmenden hatten bei jeder Frage fünf mögliche Antworten zur Auswahl: besteht überhaupt nicht, besteht in geringem Umfang, besteht in moderatem Umfang, besteht in hohem Umfang und besteht in vollem Umfang.

SSDF-Praktik	Definition in der Befragung
Sicherheitsüberprüfungen	Für alle wichtigen Features der Anwendungen, an denen ich arbeite, wird eine Sicherheitsüberprüfung durchgeführt.
Kontinuierliche Codeanalyse/Tests	Wir führen für alle unterstützten Freigaben kontinuierlich automatisierte oder manuelle Codeanalysen und Tests durch, um zuvor nicht erkannte Sicherheitslücken zu erkennen oder ihr Vorhandensein zu bestätigen.
Frühe Sicherheitstests	Sicherheitstests werden frühzeitig im Softwareentwicklungsprozess entweder durch mich oder durch ein anderes Team durchgeführt.
Effektives Reagieren auf Bedrohungen	Meine Organisation nutzt eine effektive Methode, um auf Sicherheitsbedrohungen zu reagieren.
Einbindung ins Entwicklungsteam	Es sind Sicherheitsrollen in unser Softwareentwicklungsteam eingebunden.
Anforderungen an die Dokumentation	Unsere Organisation verfügt über Verfahren, um alle Sicherheitsanforderungen zu ermitteln und zu dokumentieren, die die Software betreffen, die die Organisation entwickelt oder erwirbt, einschließlich Drittanbieter- und Open-Source-Software.
Regelmäßige Prüfung der Anforderungen	Die Sicherheitsanforderungen werden in regelmäßigen Abständen geprüft (jährlich oder häufiger, falls nötig).
Generierte Metadaten	Build-Metadaten (z. B. Abhängigkeiten, Build-Prozess, Build-Umgebung) werden entweder vom Build-Dienst oder von einem Build-Metadatengenerator erstellt, der den Build-Dienst liest.
Einbindung in Entwicklungszyklus	In meinem Unternehmen ist das Softwaresicherheitsprotokoll nahtlos in den Entwicklungsprozess eingebunden.
Projektübergreifender Standardprozess	In meinem Unternehmen gibt es einen standardisierten Prozess, um die Softwaresicherheit über Projekte hinweg sicherzustellen.
Monitoring von Sicherheitsberichten	Wir sichten fortlaufend Informationen aus öffentlichen Quellen zu möglichen Sicherheitslücken in der Software, die wir verwenden, sowie ihren Drittanbieterkomponenten.
Zugriff auf notwendige Tools	Ich habe Zugriff auf die nötigen Tools, um Sicherheitstests auszuführen.

**Tabelle 2: SSDF-bezogene Fragen**

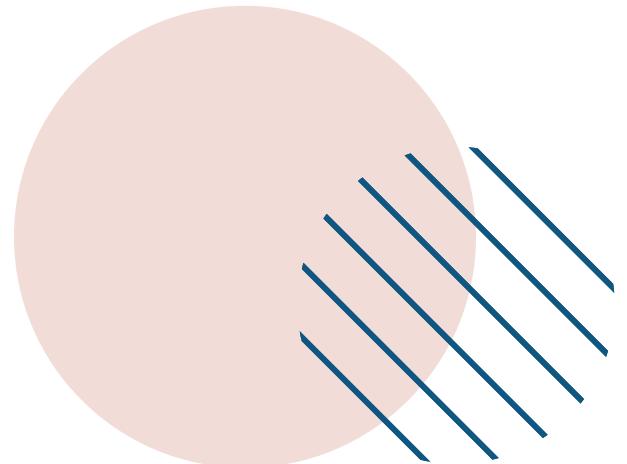
Hinweis: Die Teilnehmenden hatten bei jeder Frage sieben mögliche Antworten zur Auswahl: stimme überhaupt nicht zu, stimme nicht zu, stimme eher nicht zu, neutral, stimme eher zu, stimme zu und stimme uneingeschränkt zu.

Wir haben herausgefunden, dass aufkommende Branchenverfahren von recht vielen Unternehmen angenommen werden, aber sie könnten noch deutlich besser implementiert werden. Beispielsweise haben 66 % der befragten Personen der Aussage „In meinem Unternehmen ist das Softwaresicherheitsprotokoll nahtlos in den Entwicklungsprozess eingebunden“ zugestimmt, während nur 18 % uneingeschränkt zustimmten. In Abbildung 1 und 2 sind die Antworten der Teilnehmenden auf unsere Sicherheitsfragen zusammengefasst.

Unsere Untersuchung hat ergeben, dass die Verwendung von CI/CD-Systemen (Continuous Integration/Continuous Delivery) für Produktionsfreigaben die am häufigsten eingeführte Praktik ist – 63 % der Teilnehmenden gaben an, dass diese „in hohem Umfang“ bzw. „in vollem Umfang“ bestehe. Dass CI/CD diese Liste anführt, knüpft an vorherige Forschungsergebnisse im Bereich Sicherheit an, die ergaben, dass [die meisten Organisationen Sicherheitsscans auf Anwendungsebene im Rahmen ihres CI/CD-Prozesses implementieren](#). Außerdem ging aus gesonderten qualitativen Interviews mit einem Sicherheitsschwerpunkt hervor, dass es den meisten Entwicklerinnen und Entwicklern nicht möglich ist, solche Tools während des Entwicklungsprozesses lokal auszuführen. Beim SLSA-Framework wird ebenso auf CI-Systeme gesetzt, um sie als zentrale Integrationsbereiche für die Lieferkettensicherheit zu verwenden. Unsere Modellanalyse, auf die wir im nächsten Abschnitt näher eingehen, ergab, dass das Vorhandensein von CI in einer Organisation ein guter

Anhaltspunkt für den Reifegrad ihrer Sicherheitspraktiken ist. Daher sind wir der Überzeugung, dass es einer Organisation ohne diesen kritischen Infrastrukturteil schwerfällt, sicherzustellen, dass für die erstellten Softwareartefakte ein konsistentes Bündel an Scannern, Lint-Tools und Tests ausgeführt wird.

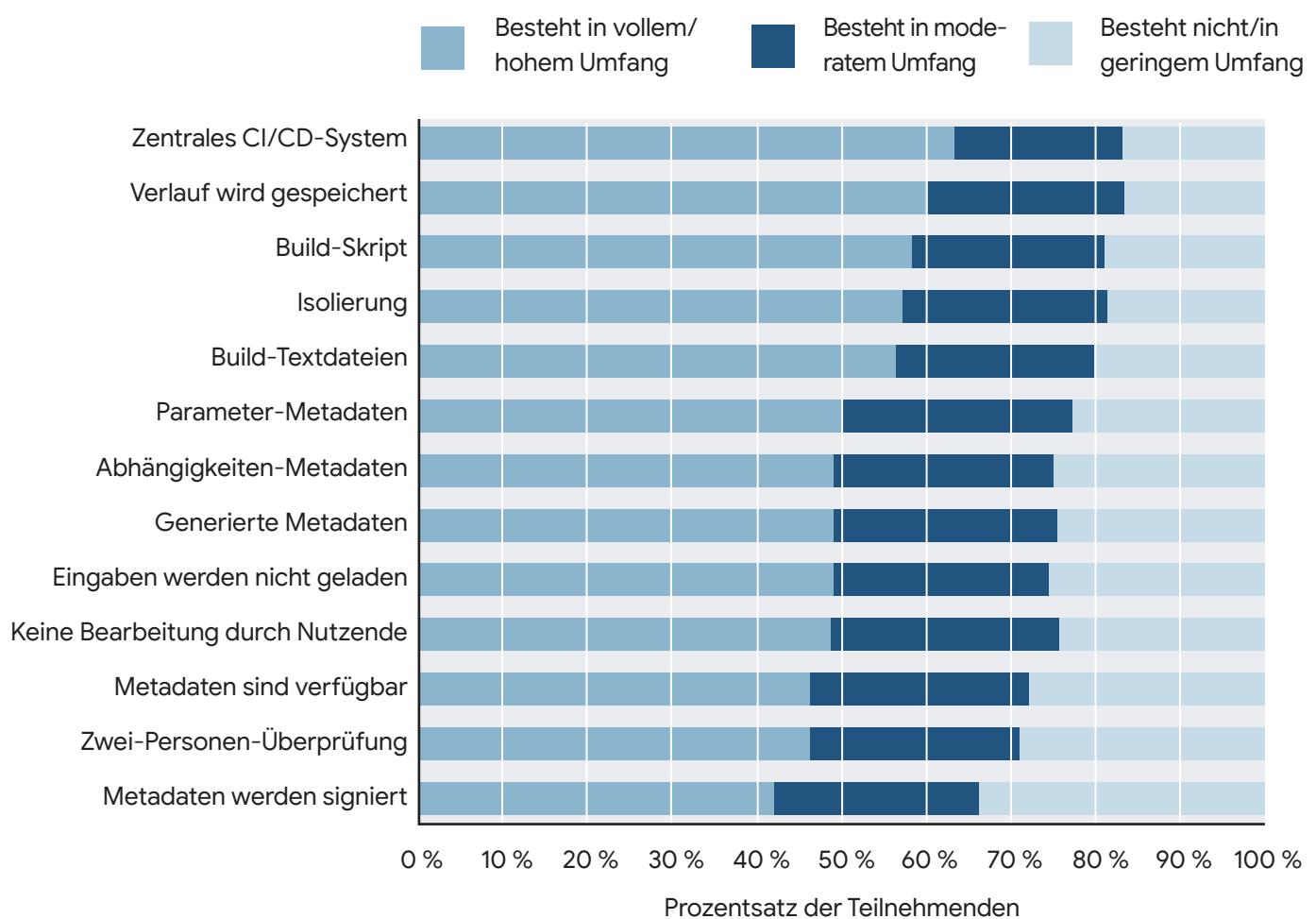
Abseits von CI/CD wurden unter anderem bereits folgende Praktiken allgemein implementiert: der Codeverlauf wird unbegrenzt aufbewahrt (60 %), Builds werden ausschließlich über Skripts definiert (58 %), Builds werden voneinander isoliert (57 %) und Build-Definitionen werden in Versionsverwaltungssystemen gespeichert (56 %). Die zwei am seltensten etablierten Praktiken waren dagegen, dass zwei oder mehr prüfende Personen jede Codeänderung genehmigen müssen (45 %) und dass Build-Metadaten signiert werden, um Manipulation zu verhindern/zu entdecken (41 %).



Wir haben die Teilnehmenden nicht nur zu etablierten Praktiken befragt, sondern auch gebeten, einer Reihe von Aussagen zur Sicherheit in ihrer Organisation zuzustimmen bzw. diesen nicht zuzustimmen. Am meisten Zustimmung verzeichnete die Aussage „Wir sichten fortlaufend Informationen aus öffentlichen Quellen zu möglichen Sicherheitslücken in der Software, die wir verwenden, sowie ihren Drittanbieterkomponenten“ – 81 % der Teilnehmenden äußerten diesbezüglich ihre Zustimmung. Am anderen Ende des Spektrums wurde dagegen der Aussage hinsichtlich negativer Auswirkungen von Sicherheitspraktiken auf die Softwareentwicklung am wenigsten zugestimmt – 56 % der befragten Personen stimmten folgender Aussage zu: „Die Softwaresicherheitsprozesse, die in meiner Firma vorhanden sind, verlangsamen den Entwicklungsprozess der Anwendungen, an denen ich arbeite.“

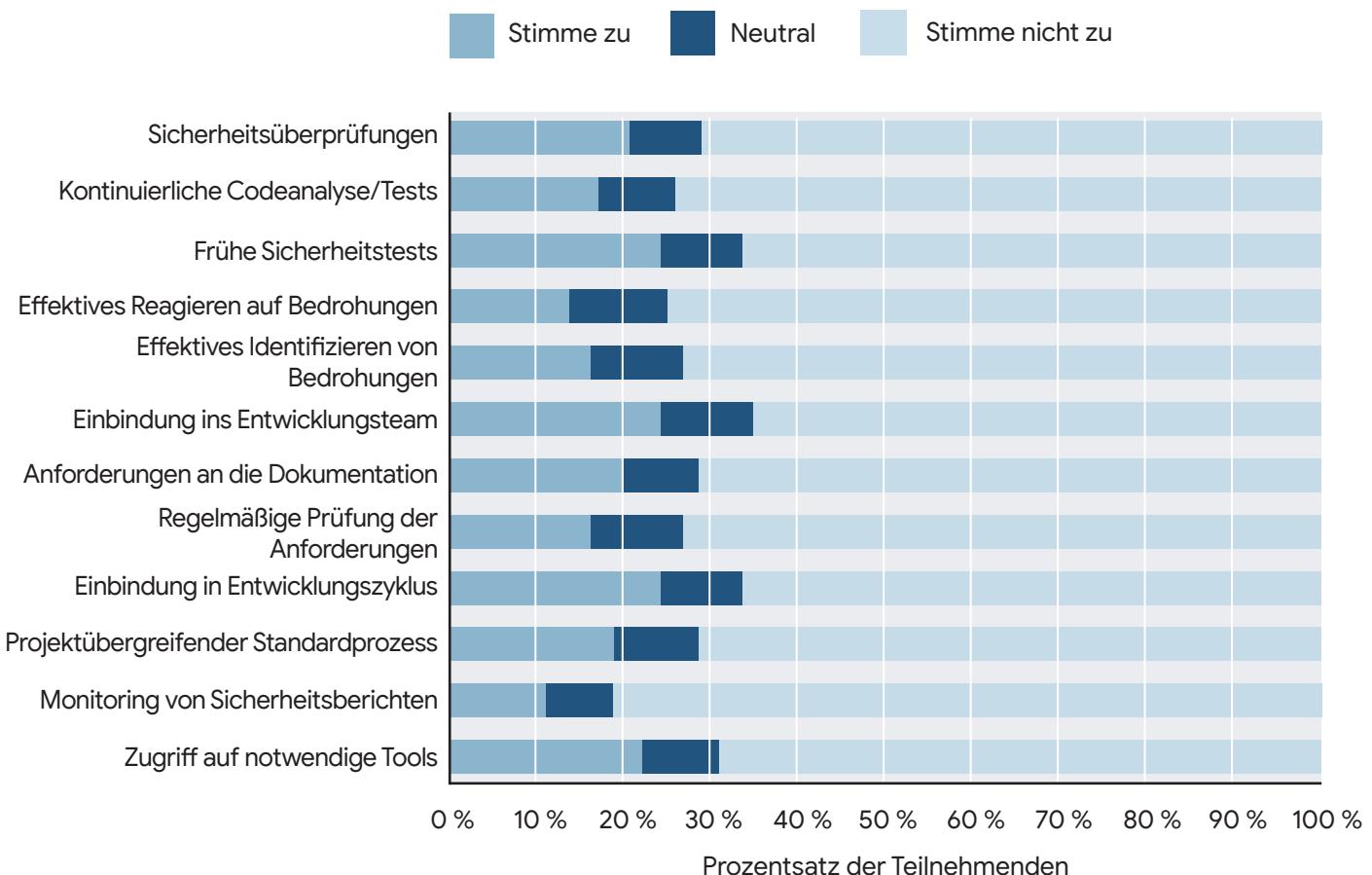
Es ist zwar ermutigend, dass dieser Aussage am wenigsten zugestimmt wurde. Dennoch lässt die Tatsache, dass die meisten befragten Personen angaben, die aktuellen Sicherheitsprozesse würden das Entwicklungstempo bremsen, erahnen, dass in puncto Verbesserung der Sicherheitstools und -ansätze noch eine Menge Luft nach oben ist. Auch unsere Modellanalyse unterstützt diese Interpretation und zeigt gemischte, aber dennoch geringfügige Auswirkungen der Sicherheitsprozesse auf die Softwarebereitstellungsleistung.





**Abbildung 1: Verwurzelung der SLSA-Praktiken**

Umfrageantworten zur Verwurzelung der SLSA-Praktiken. Die meisten Teilnehmenden geben an, dass diese Praktiken in gewissem Maße etabliert seien, aber nur relativ wenige sagten, dass sie „in vollem Umfang“ bestünden.



**Abbildung 2: Verwurzelung der SSDF-Praktiken**

Umfrageantworten zur Verwurzelung der SSDF-Praktiken. Ähnlich wie bei den SLSA-Praktiken gaben die meisten Teilnehmenden an, dass ihre Organisation all diese Praktiken befolge.

# Was hilft Firmen dabei, gute Sicherheitspraktiken zu befolgen?

Die Anwendungssicherheit stellt nur einen Aspekt der Softwareentwicklung dar. Daher ist sie nur eine unter vielen konkurrierenden Anforderungen, denen Entwickelnde Zeit und Aufmerksamkeit widmen müssen. Sicherheitsansätze, die das Potenzial für viel Reibung bergen, können für Entwicklerinnen und Entwickler frustrierend und im Großen und Ganzen ineffektiv sein, da die Beschäftigten versuchen werden, die Reibungspunkte zu umgehen. Beispielsweise hat eine Reihe von Umfrageinterviews mit Softwareentwicklerinnen und -entwicklern ergeben, dass ihre Berührungspunkte mit den Sicherheitsteams entweder auf den Beginn oder das Ende eines Projektes beschränkt waren und es schwierig sein konnte, mit den Teams zu interagieren. Eine Person drückte das so aus: „Wir haben zwar ein Team für die Anwendungssicherheit, aber ich habe meinen Code noch nie von dessen Mitgliedern prüfen lassen... Ich bin da wie die meisten Entwicklerinnen und Entwickler; ich meide sie in der Regel.“

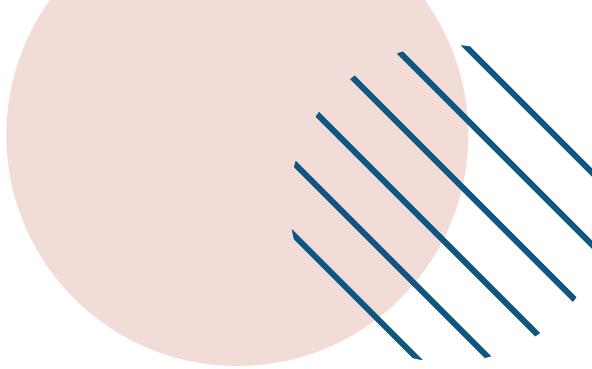
Ein Ansatz, um die Softwaresicherheit zu verbessern, besteht darin, die Hemmnisse abzubauen, die verhindern, dass die Sicherheitspraktiken befolgt werden können. Die Mitglieder der Entwicklungsabteilungen, mit denen wir gesprochen haben, wollten eigentlich das Richtige tun, und viele von ihnen haben ihren Frust darüber geäußert, dass das Liefern von Features und Fehlerkorrekturen immer Vorrang gegenüber potenziellen Sicherheitsproblemen hatte. Beispielsweise hat eine Person, die für eine andere sicherheitsfokussierte Umfrage interviewt wurde, die größte Sicherheitsherausforderung ihrer Firma so beschrieben: „Wir müssen das erst einmal zum vorrangigen Ziel erklären. Das ist nicht aufregend, dadurch erhöhen sich unsere Verkaufszahlen nicht [und] es ist kein Problem, bis es zu einem wird.“

Unseren Umfragedaten weisen darauf hin, dass es einige Faktoren gibt, die es den Entwickelnden leichter machen, „die sichere Route zu wählen.“

Der wichtigste Faktor, auf den wir gestoßen sind, ist gar nicht technischer, sondern kultureller Natur: Bei **Organisationen, die der „generativen“ Kulturgruppe nach Westrum am nächsten kamen, war es deutlich wahrscheinlicher, dass ihre Beschäftigten angaben, dass Sicherheitspraktiken weitgehend im Unternehmen etabliert sind**, so wie sie im SLSA-Framework definiert sind.<sup>1</sup> Zu den Merkmalen einer generativen Kultur zählt, dass die Beschäftigten intensiv zusammenarbeiten, Risiken und Verantwortlichkeiten untereinander aufteilen und aus den Fehlern der Vergangenheit lernen. Wir nehmen an, dass sich diese Merkmale auf verschiedene Arten in besseren Sicherheitspraktiken zeigen, z. B. dass Softwareentwicklerinnen und -entwickler ermutigt werden, die Sicherheit der Lieferkette proaktiver anzugehen, dass Mitarbeitende unabhängig von ihrer Position für ihre Sicherheitsbemühungen belohnt werden oder dass die wahrgenommenen Risiken minimiert werden, die Beschäftigte damit verbinden, potenzielle Sicherheitsprobleme zu melden.

Technologisch gesehen haben drei der wichtigsten treibenden Faktoren der Sicherheit mit der Infrastruktur zu tun. Das ergibt Sinn: Wenn es mit Ihrer Infrastruktur leicht möglich ist, auf Sicherheitslücken zu scannen oder manuelle Code Reviews durchzuführen, führen die Entwickelnden diese Aufgaben auch eher durch. Insbesondere haben wir festgestellt, dass Firmen, **die Systeme für die Versionsverwaltung, Continuous Integration und Continuous Delivery haben, auch über fester verankerte SLSA-Praktiken verfügen.**

<sup>1</sup> Interessanterweise war bei diesen Teilnehmenden die Wahrscheinlichkeit nicht höher, dass sie auch den Fragen zum SSDF des NIST zustimmen. Obwohl SLSA und SSDF unterschiedliche Aspekte der Sicherheit in puncto Anwendungsentwicklung behandeln, hatten wir erwartet, dass es zwischen diesen zwei Reihen von Fragen zu Überlappungen kommt. Wie bereits erwähnt, ist es möglich, dass bei der SSDF-Antwortskala Voreingenommenheit zum Tragen kam, bestimmten Fragen „zuzustimmen“. Das würde die Unterschiede erklären.



Dabei entfällt der Löwenanteil wahrscheinlich auf den Zeitpunkt, zu dem eventuelle Sicherheitsprobleme die Aufmerksamkeit der Entwickelnden erhalten. Aus einer anderen Untersuchung wissen wir, dass das primär während der CI der Fall ist. In der Regel wird die CI direkt vor den Code Reviews durchgeführt und zwar dann, wenn die Scanner für Sicherheitslücken und andere Codeanalysetools ausgeführt werden, da so dafür gesorgt wird, dass alle Code-Commits denselben Sicherheitsanforderungen unterliegen. Ist kein zentralisiertes Build-System vorhanden, ist es viel schwieriger, solche ständigen Scans durchzuführen, und ist keine Versionsverwaltung vorhanden, ist es schwieriger, überhaupt ein zentralisiertes Build-System zu nutzen.

Wenn die Sicherheitsscans im Rahmen von CI/CD durchgeführt werden, ist das jedoch möglicherweise den Softwareentwicklerinnen und -entwicklern nicht früh genug. In einer Reihe sicherheitsbezogener Interviews mit Personen, die Anwendungen entwickeln, hörten wir immer wieder, dass es ihnen Zeit und Mühe ersparen würde, wenn sie bereits auf ihren Entwickler-Workstations Sicherheitsscans ausführen würden. Zwei Situationen wurden oft erwähnt: 1) Sie wollen vorab wissen, ob sie auf einer Abhängigkeit mit bekannten Schwachstellen aufbauen, damit sie noch einmal prüfen können, ob sie diese Abhängigkeit wirklich verwenden möchten, bevor sie auf ihr aufbauen. 2) Sie möchten vermeiden, dass sie lange CI-Wartezeiten auf sich nehmen müssen, die manchmal Stunden betragen, um lediglich herauszufinden, ob ihre letzten

Änderungen ein Sicherheitsproblem behoben haben. In beiden Fällen haben die Entwickelnden gesagt, dass eine CI-Absicherung zwar nötig sei, ihnen aber die Möglichkeit, dieselben Sicherheitstools auch lokal auszuführen, helfen würde, schneller und effizienter zu arbeiten.

Die oben beschriebenen kulturellen und technologischen Faktoren haben den größten Einfluss auf die Sicherheit, sind aber nicht die einzigen. Weitere nennenswerte Faktoren umfassen:

- Flexible Arbeitsmodelle (zum Beispiel: Unterstützt die Organisation das Arbeiten im Homeoffice?)
- Cloud-Nutzung (öffentliche oder private Cloud)
- Arbeit an „cloudnativen“ Anwendungen oder Diensten
- Vorhandensein des Gefühls, dass das Unternehmen das eigene Team schätzt und in dieses investiert
- Geringe Fluktuation im Team
- Organisationsgröße, wobei größere Organisationen höhere Sicherheitspunktzahlen melden

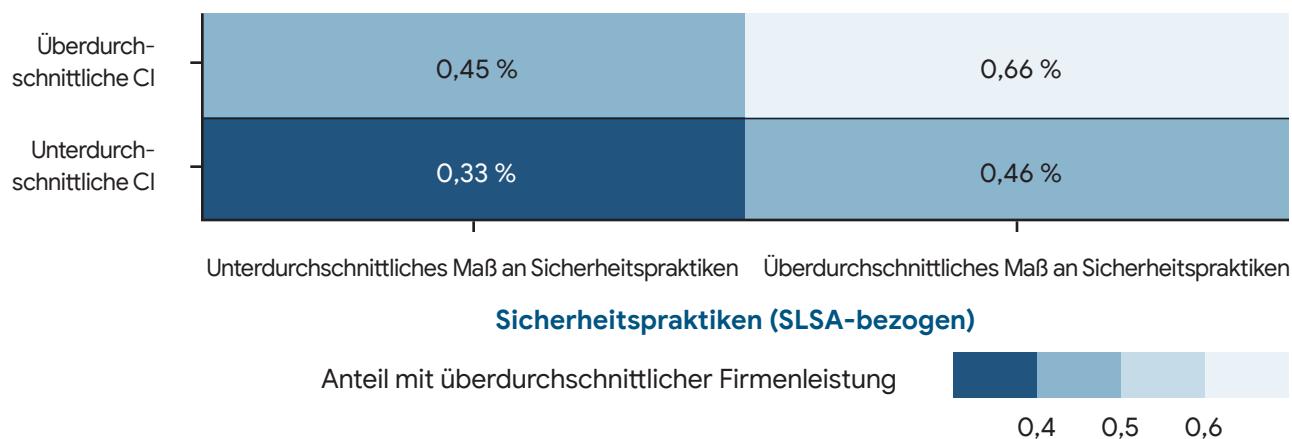
Diese Faktoren scheinen jedoch im Wesentlichen mit der generativen Kultur nach Westrum (z. B. flexible Arbeitsmodelle, Gefühl der Wertschätzung durch das Unternehmen oder geringe Teamfluktuation) oder dem Einsatz von CI/CD (z. B. Arbeiten an einer cloudnativen Anwendung oder für eine große Organisation) zu korrelieren. Diese Daten begründen unsere Meinung, dass die Organisationskultur sowie Prozesse der modernen Entwicklung wie Continuous Integration die Anwendungsentwicklungssicherheit einer Organisation am stärksten antreiben und Firmen am besten dort ansetzen sollten, wenn sie ihren Sicherheitsstatus verbessern möchten.

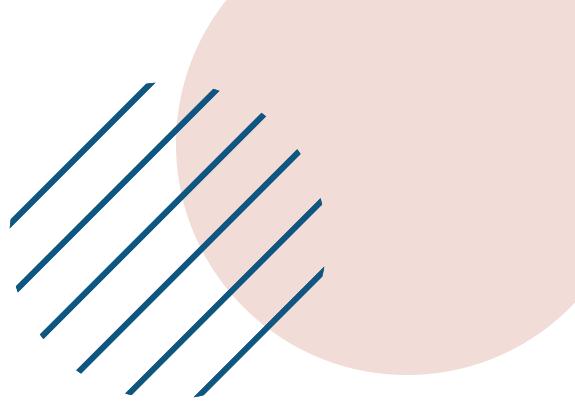
# Zu welchen Ergebnissen führen gute Sicherheitsgrundsätze?

Mit welchen möglichen Vorteilen können Organisationen rechnen, wenn sie ihre Sicherheitspraktiken rund um die Softwareentwicklung verbessern? Unsere Untersuchungsdaten bestätigen, dass die befragten Personen mit einer **geringeren Wahrscheinlichkeit von Sicherheitsverstößen, Dienstausfällen und Leistungsverschlechterungen rechnen, wenn Firmen die Verankerung von Sicherheitspraktiken für die Lieferkette vorantreiben**. Gleichermassen haben wir in einer anderen Untersuchung in der ersten Hälfte des Jahres 2022 herausgefunden, dass das Ausführen von Tools wie Sicherheitslückenscannern während der CI die Wahrscheinlichkeit deutlich erhöht, dass vorhandene Sicherheitslücken in Softwareabhängigkeiten identifiziert werden: Bei Teilnehmenden, die solche Tools verwendeten, war die Wahrscheinlichkeit fast doppelt so hoch, dass sie angaben, eine Sicherheitslücke im eigenen Code oder in einer der Abhängigkeiten gefunden zu haben. Kurzum: Die SLSA- und SSDF-Praktiken scheinen wie vorgesehen zu funktionieren.

Wir behaupten nicht, Sicherheitsbedrohungen beseitigen lassen, aber unsere Daten lassen vermuten, dass damit das Sicherheitsrisiko von Organisationen gesenkt werden kann.

Sicherheitspraktiken können sich auch positiv auf leistungsbasierte Ergebnisse auswirken, aber da gibt es einen Haken: CI spielt dabei eine zentrale Rolle. Wenn CI nicht implementiert wurde, wirken sich die Sicherheitspraktiken nicht auf die Softwarebereitstellungsleistung aus. Wenn CI jedoch implementiert wurde, haben die Sicherheitspraktiken einen starken positiven Effekt auf die Softwarebereitstellungsleistung. Das bedeutet im Klartext, dass CI notwendig ist, damit sich Sicherheitspraktiken positiv auf die Softwarebereitstellungsleistung auswirken. Ferner haben Sicherheitspraktiken im Allgemeinen einen positiven Effekt auf die Firmenleistung und wenn die CI dann noch fest im Unternehmen verankert ist, verstärkt sich dieser Effekt noch. In der folgenden Grafik wurde versucht, diesen Effekt zu visualisieren.





Zusammen mit einer Verringerung der wahrgenommenen Sicherheitsrisiken meldeten die Teilnehmenden auch **weniger Burnout-Erkrankungen** unter ihren Teammitgliedern und waren eher bereit, ihre Organisation weiterzuempfehlen, da sie diese als **tollen Ort zum Arbeiten** empfanden. Diese beiden Erkenntnisse sprechen dafür, dass das Thema Sicherheit dem sowieso schon mit Arbeit ausgelasteten Softwareentwicklungsteam noch eine weitere Last aufbürdet. Es gibt einen Weg, die Sicherheitsrisiken zu reduzieren und gleichzeitig den Entwickelnden mehr Freude an der Arbeit zu bescheren – mit Tools und Prozessen, die ihnen helfen, in den vorhandenen Entwicklungs-Workflow gewisse sichere Praktiken einzubinden, statt dass es zu ungeplanter Arbeit oder „Feueralarmübungen“ kommt, wenn eine Bedrohung entdeckt wird.

Zusammengenommen deuten unsere Erkenntnisse darauf hin, dass **gesunde, leistungsstarke Teams dazu tendieren, auch über weitgehend etablierte gute Sicherheitspraktiken zu verfügen**. Wie bereits angemerkt, gibt es hier jedoch nach wie vor noch Verbesserungspotenzial. Wenn Sie Ansätze wie das SLSA- oder SSDF-Framework verfolgen, reicht das möglicherweise nicht aus, um alle Kultur- und Leistungsmesswerte, die wir erfassen, zu verbessern. Fest steht jedoch, dass die Sicherheit keinesfalls zulasten anderer Entwicklungsrioritäten gehen muss.

Es gibt einen Weg, die Sicherheitsrisiken zu reduzieren **und** gleichzeitig den Entwickelnden mehr Freude an der Arbeit zu bescheren – mit Tools und Prozessen, die ihnen helfen, in den vorhandenen Entwicklungs-Workflow gewisse sichere Praktiken einzubinden, statt dass es zu ungeplanter Arbeit oder „Feueralarmübungen“ kommt, wenn eine Bedrohung entdeckt wird.

# 05

## Überraschende Erkenntnisse



Derek DeBellis

Zwar fokussieren wir uns in den Berichten der einzelnen Jahre auf die Antworten aus der zugehörigen Befragung, aber wir bemühen uns dennoch nach Kräften, diese Erkenntnisse auch im Kontext aller bisherigen „State of DevOps“-Berichte und angrenzenden Forschungsergebnisse (zum Beispiel Forschungsergebnisse zu Burnout und Firmenkultur) zu verstehen. Es stellt eine Grundkomponente dieses Forschungsprogramms dar, die Zuverlässigkeit dieser Effekte zu testen, indem wir versuchen, sie zu replizieren. Das gibt uns die Möglichkeit, unsere Überzeugungen anzupassen, sodass sie zu den Daten passen, und sich entwickelnde und neue Trends zu verstehen.

Dieses Jahr haben wir einige überraschende Entdeckungen gemacht. Dafür gibt es unzählige mögliche Gründe. Zum einen hat sich dieses Jahr der Querschnitt verändert und enthielt mehr Personen, die noch nicht so lange in ihrem Job arbeiten, als dies bei den vorherigen Berichten der Fall war. Eine Interpretation dafür ist, dass wir vermehrt von den Leuten hören, die direkt dafür verantwortlich sind, die technischen Praktiken und Fähigkeiten zu implementieren, statt vielleicht von den Personen, die dafür verantwortlich sind, die Implementierung dieser Praktiken zu überwachen oder diesbezüglich

Mitarbeitende anzuleiten. Eine andere Möglichkeit ist, dass es in der Branche oder der Welt zu einer Veränderung kam; was gestern noch funktioniert hat, funktioniert nicht zwingenderweise auch morgen noch. Beispielsweise könnten gesamtwirtschaftliche Kräfte und ein weiteres Jahr, das größtenteils von der Pandemie überschattet wurde, dafür gesorgt haben, dass sich die „Physik“ von DevOps geändert hat. Zuletzt könnten subtile Änderungen an den Dingen, die wir in unserem Modell berücksichtigt haben, die Beziehung zwischen den einzelnen Variablen verändert haben.<sup>1</sup>



<sup>1</sup> Das „Book of Why“ von Judea Pearl und „Statistical Rethinking“ von Robert McElreath enthalten tolle Beispiele dafür, wie sich die Auswahl dessen, was man in ein statistisches Modell aufnimmt bzw. nicht aufnimmt, auf die Ausgabe eines Modells auswirken kann.

Ein unerwartetes Ergebnis oder eines, für das keine Hypothese aufgestellt wurde, bringt uns Forschende in eine schwierige Lage, wenn es an der Zeit ist, den Bericht zu schreiben. Angesichts des Risikos, dass das Ergebnis unecht ist oder zumindest der in mehreren Studien empirisch erbrachte Beweis fehlt oder es gar im Widerspruch zu den Ergebnissen solcher Studien steht, besteht der verantwortungsvolle Schritt darin, weitere Nachforschungen anzustellen, um zu versuchen, die Ergebnisse zu replizieren und ihre Ursache zu verstehen.<sup>2</sup> Wenn sich eine Forscherin oder ein Forscher auf die überraschenden Erkenntnisse konzentriert, besteht außerdem die Gefahr, die vielen Effekte, die in jahrelanger Forschung zuverlässig aufgetreten sind, herunterzuspielen. Für jeden „State of DevOps“-Bericht untersuchen wir statistisch über hundert Pfade. Das birgt das Risiko, dass es einfach durch Zufall zu unechten Erkenntnissen kommt. Wir versuchen, dem entgegenzuwirken, indem wir uns jährlich um die Replikation der aufgetretenen Effekte bemühen.

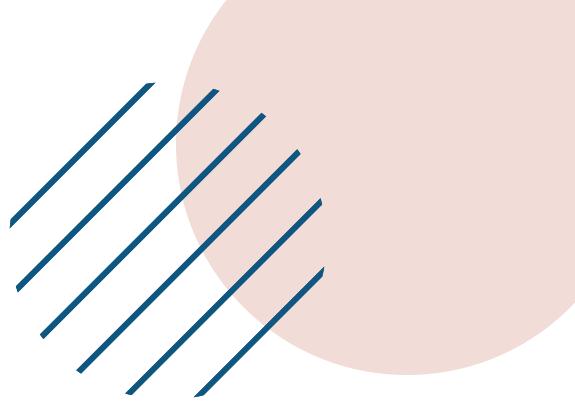
Wenn wir dagegen nicht von der Erkenntnis berichten, birgt dies das Risiko eines Publikationsbias<sup>3</sup>, was effektiv dazu führt, dass das bekannt wird, was erwartet wird oder akzeptabel ist, und das, was unerwartet oder „schwer bekömmlich“ ist, in einer Schublade verschwindet. Wir bemühen uns um eine Balance: Wir wollen aus neu aufkommenden Erkenntnissen keine Sensationsmeldungen machen, aber wir halten es dennoch für äußerst wichtig, darüber zu berichten. Hier nennen wir die Dinge, die uns am meisten überrascht haben, und versuchen uns an einer Interpretation:

**01** Wir haben immer wieder festgestellt, dass sich die Praktiken der Entwicklung nach Baumschema positiv auf die Softwarebereitstellungsleistung auswirken. Tatsächlich haben wir dies seit 2014 in jedem Jahr der Studie festgestellt. Dieses Jahr kam es bei den Kompetenzen der Entwicklung nach Baumschema zu Ergebnissen, die aus der Reihe tanzen. Sie wirkten sich negativ auf die Softwarebereitstellungsleistung aus. In früherer Berichten war das Gegenteil der Fall. Da diese Erkenntnis so abnorm ist, sind wir schon gespannt, ob sie in zukünftigen Untersuchungen bestätigt wird, und auch neugierig, ob die Community dafür Erklärungen hat.

**02** Wir haben herausgefunden, dass sich die Softwarebereitstellungsleistung nur dann vorteilhaft auf die Firmenleistung auswirkt, wenn auch die operative Leistung hoch ist, und bei vielen der Teilnehmenden war die operative Leistung nicht hoch. Dies steht im Widerspruch zu früheren Iterationen unserer Forschung, bei denen der Zusammenhang zwischen der Softwarebereitstellungsleistung und der Firmenleistung viel deutlicher war.

<sup>2</sup> Kerr, N. L. (1998). HARKing: Hypothesizing after the results are known. *Personality and social psychology review*, 2(3), 196–217.

<sup>3</sup> Rosenthal, R. (1979). The file drawer problem and tolerance for null results. *Psychological bulletin*, 86(3), 638.



## 03 Dokumentationsverfahren haben sich negativ auf die Softwarebereitstellungsleistung ausgewirkt.

Auch dies steht im Widerspruch zu früheren Berichten. Eine Hypothese dafür ist, dass besonders bei leistungsstarken Teams die Dokumentation zunehmend zu einer automatisierten Tätigkeit wird. Solange wir diesbezüglich noch keine weiteren Daten erhoben haben, haben wir nur wenig Beweise, um diese Annahme zu stützen oder zu widerlegen.

## 04 Einige technische Funktionen (z. B. die Entwicklung nach Baumschema, lose gekoppelte Architektur, CI, CD) schienen das Maß an Burnout-Erkrankungen vorauszusagen. Wie oben bereits erwähnt, umfasste der diesjährige Querschnitt viele Teilnehmende, die sich in ihrer Laufbahn an einem viel früheren Punkt befanden, als die Teilnehmenden der Querschnitte der letzten Jahre. Daher kann es sein, dass wir mit den Leuten gesprochen haben, die dafür verantwortlich sind, die Funktion zu implementieren, und nicht mit den Personen, deren Aufgabe es ist, die Initiative zu konzipieren oder deren Umsetzung zu beaufsichtigen. Der Implementierungsprozess kann deutlich schwieriger sein als dessen Überwachung. Wir möchten weitere Nachforschungen anstellen, um diese Erkenntnis besser zu verstehen.

## 05 Reliability-Engineering-Praktiken hatten negative Auswirkungen auf die Softwarebereitstellungsleistung. Eine Erklärung dafür ist, dass diese zwei Dinge nicht zwingend kausal miteinander verknüpft sind. Dieses Jahr haben wir durch eine neue Clusteranalyse festgestellt (siehe „Wie schneiden Sie ab?“), dass bei einer Teilmenge von Clustern der Fokus anscheinend auf der Zuverlässigkeit lag, während die Softwarebereitstellungsleistung ignoriert wurde. Wir sind der Meinung, dass diese zwei Dinge voneinander entkoppelt sind, also dass Sie eines davon tun können, ohne auch das andere tun zu müssen. Letzten Endes muss aber die Zuverlässigkeit etabliert sein, wenn die Softwarebereitstellungsleistung zur Firmenleistung beitragen soll.

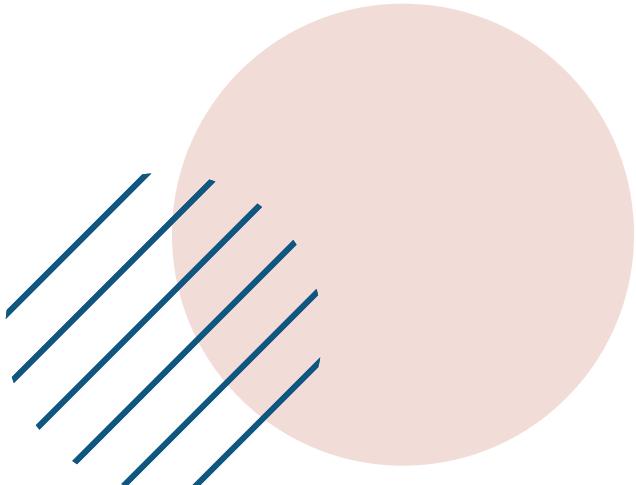
## 06 Wir haben die SLSA-bezogenen Praktiken ergänzt, um herauszufinden, ob Teams auf diese Ansätze setzen, um dauerhaft für eine sichere Softwarelieferkette zu sorgen. Wir haben erwartet, dass die Implementierung von Sicherheitspraktiken und die Leistung miteinander verknüpft sind (z. B. die Verwendung technischer Fähigkeiten, bessere Softwarebereitstellungsleistung und bessere Firmenleistung). Aber wir waren überrascht, dass es tatsächlich die Sicherheitspraktiken waren, durch die sich die technischen Fähigkeiten auf die Softwarebereitstellungsleistung und Firmenleistung ausgewirkt haben.



Das Einbeziehen der SLSA-bezogenen Praktiken scheint einen Großteil des Effektes von Continuous Integration, Versionsverwaltung und Continuous Delivery auf die Softwarebereitstellungsleistung und die Firmenleistung zu erklären. Anders gesagt konnten wir in den Daten eine Kausalkette ausmachen: Viele technische Fähigkeiten wirken sich positiv auf SLSA-bezogene Praktiken aus und durch diesen positiven Einfluss auf die SLSA-bezogenen Praktiken wirken sie sich dann positiv auf die Softwarebereitstellungsleistung und die Firmenleistung aus. Wir haben Mediationsanalysen verwendet, um dieses Ergebnis zu ermitteln.<sup>4,5</sup> All dies gibt uns einen Anlass, zu untersuchen, ob unser Maß der SLSA-bezogenen Praktiken auch andere Eigenschaften von Teams verfolgt (z. B. allgemeine Leistung) und in welcher Weise Sicherheitspraktiken zu einer besseren Softwarebereitstellungsleistung und Firmenleistung führen.

Wir freuen uns darauf, diese Effekte nächstes Jahr wieder zu untersuchen, um herauszufinden, ob wir diese neuen Muster reproduzieren und erklären können oder sie eher als Ausreißer abtun sollten, in welchem Fall wir uns aber auch bemühen sollten, dafür eine Erklärung zu finden. Wie immer freuen wir uns auch hier auf Feedback von der Community.

Treten Sie doch der [DORA-Community](http://dora.community) (<http://dora.community>) bei und beteiligen Sie sich an einer Diskussion über diese überraschenden Ergebnisse und die weiteren Erkenntnisse des diesjährigen Berichts.



<sup>4</sup> Jung, Sun Jae. „Introduction to Mediation Analysis and Examples of Its Application to Real-world Data.“ Journal of Preventive Medicine and Public Health = Yebang Uihakhoe chi Vol. 54,3 (2021): 166-172. doi:10.3961/jpmph.21.069.

<sup>5</sup> Carrión, Gabriel Cepeda, Christian Nitzl und José L. Roldán. „Mediation Analyses in Partial Least Squares Structural Equation Modeling: Guidelines and Empirical Examples.“ Partial Least Squares Path Modeling. Springer, Cham, 2017. 173-195.

# 06

## Demografische und firmenbezogene Merkmale



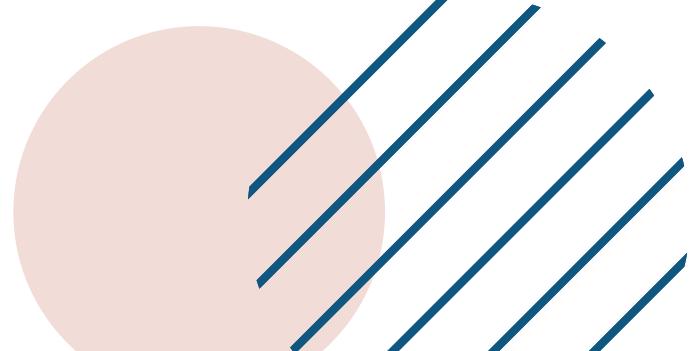
Derek DeBellis

Danke für Ihre Beiträge zu unserer Forschung und Branche!

Ähnlich wie in den vergangenen Jahren haben wir für alle Teilnehmenden der Befragung demografische Informationen erfasst. Die Kategorien umfassen Geschlecht, Behinderungen und unterrepräsentierte Gruppen.

### Wer hat an der Umfrage teilgenommen?

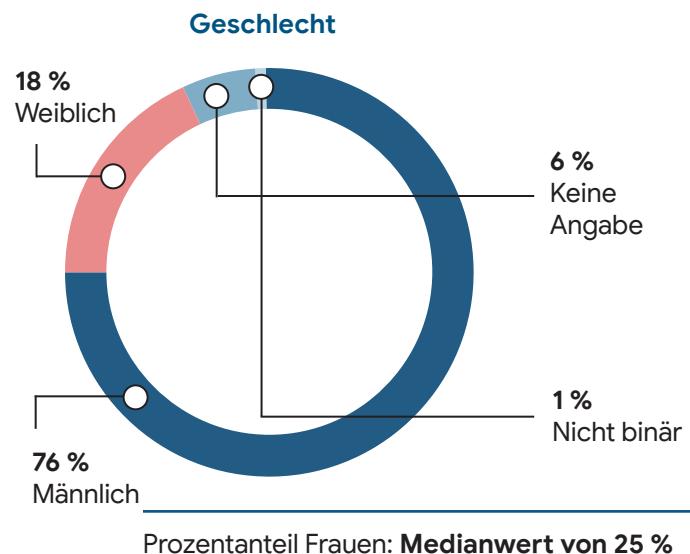
Da wir acht Jahre lang geforscht und dabei über 33.000 Umfrageergebnisse von Fachkräften der Branche erhalten haben, zeigt der Bericht „State of DevOps“ die Softwareentwicklungs- und DevOps-Praktiken auf, die die Teams und Organisationen von heute am erfolgreichsten machen. Dieses Jahr haben über 1.350 Berufstätige aus einer Vielzahl von Branchen rund um den Globus ihre Erfahrungen mit uns geteilt, damit wir die Faktoren, die zu einer höheren Leistung führen, besser verstehen. Danke für Ihre Beiträge zu unserer Forschung und Branche! Zusammenfassend lässt sich sagen, dass die Repräsentation über alle demografischen und firmenbezogenen Größenwerte hinweg bemerkenswert stabil geblieben ist.



# Demografie

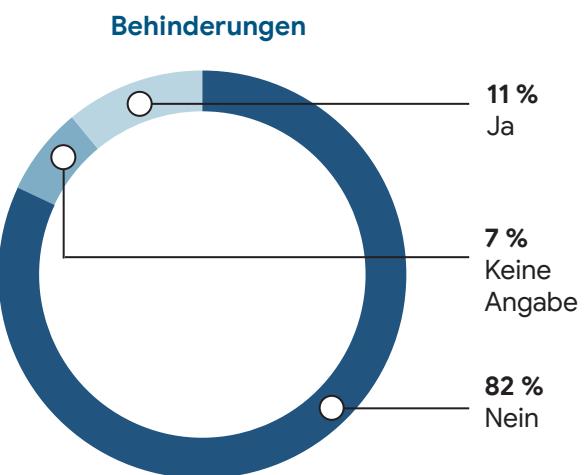
## Geschlecht

Relativ zum Jahr 2021 wies der diesjährige Querschnitt einen höheren Anteil von Teilnehmerinnen auf (18 % im Vergleich zu 12 %). Der Anteil von Teilnehmern (76 %) war dagegen niedriger als 2021 (83 %). Die befragten Personen gaben an, dass Frauen 25 % ihrer Teams ausmachen, was dem Wert aus 2021 entspricht (25 %).



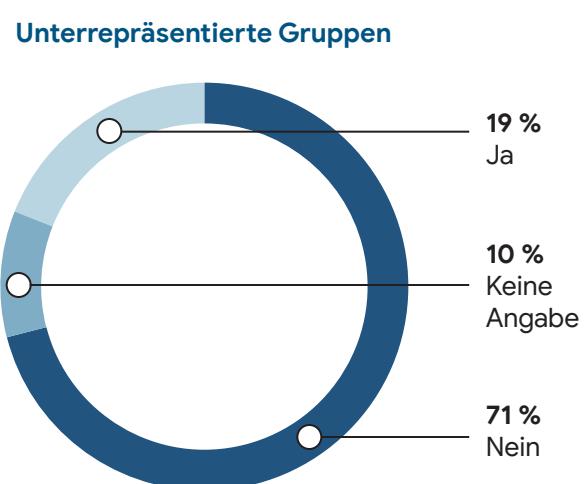
## Behinderungen

Um Behinderungen zu erfassen, haben wir sechs Aspekte verwendet, die sich am [Washington Group-Short Set](#) orientieren. Dies ist das vierte Jahr, in dem wir nach Behinderungen gefragt haben. Der Prozentsatz der Menschen mit Behinderungen entspricht dem Prozentsatz unseres Berichtes aus dem Jahr 2021 (11 %).



## Unterrepräsentierte Gruppen

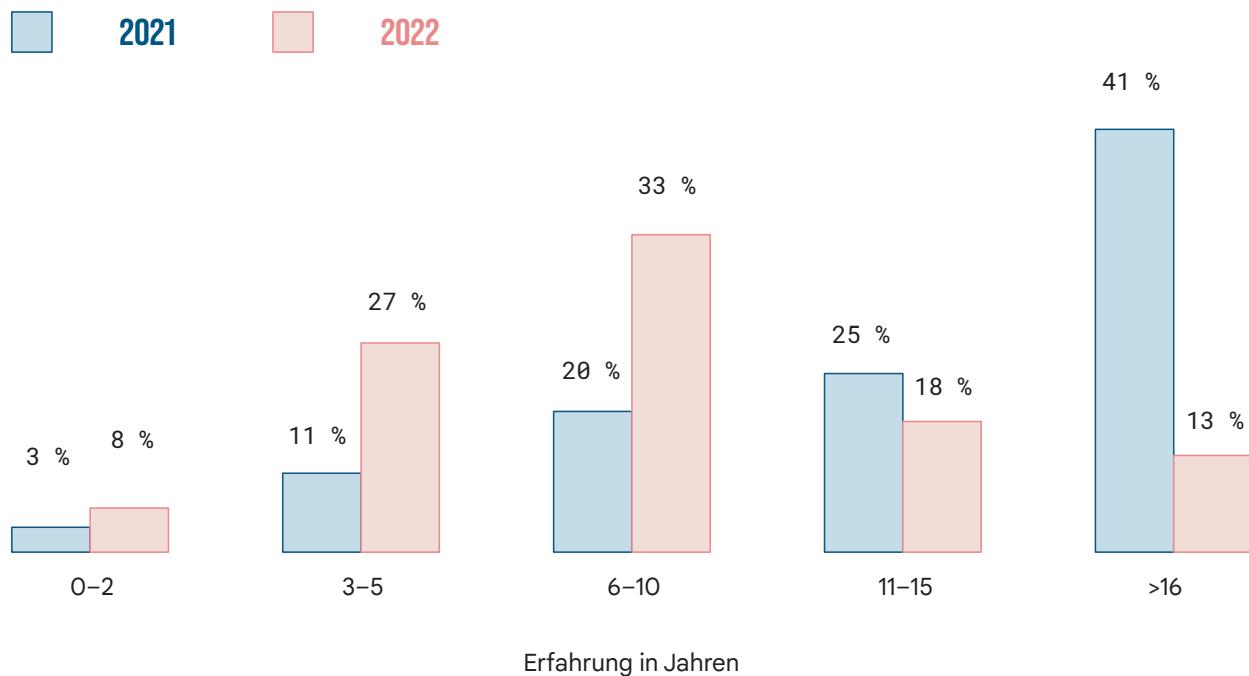
Eine Person kann aufgrund ihrer ethnischen Herkunft, ihres Geschlechts oder einer anderen Eigenschaft einer unterrepräsentierten Gruppe angehören. Dies ist das fünfte Jahr, in dem wir nach Unterrepräsentation gefragt haben. Der Prozentsatz der Personen, die sich als Teil einer unterrepräsentierten Gruppe einstufen, ist leicht gestiegen: von 17 % im Jahr 2021 auf 19 % im Jahr 2022.

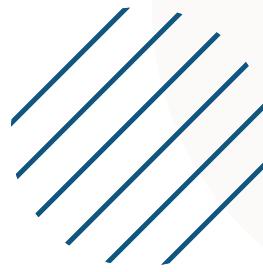


## Erfahrung in Jahren

Auffallend ist, dass dieses Jahr mehr Teilnehmende nur fünf Jahre oder weniger Erfahrung hatten (35 %) als 2021 (14 %). Da ist es sicherlich nicht überraschend, dass der Anteil der Teilnehmenden mit über 16 Jahren Erfahrung (13 %) nur einen Bruchteil des Wertes von 2021 darstellt (41 %).

Anhand dieser Verschiebung lassen sich einige der Muster erklären, die sich aus den diesjährigen Daten herauskristallisiert haben. Wir halten es daher für wichtig, dies beim Interpretieren der Ergebnisse im Hinterkopf zu behalten, besonders, wenn man sie mit denen des Vorjahres vergleicht.





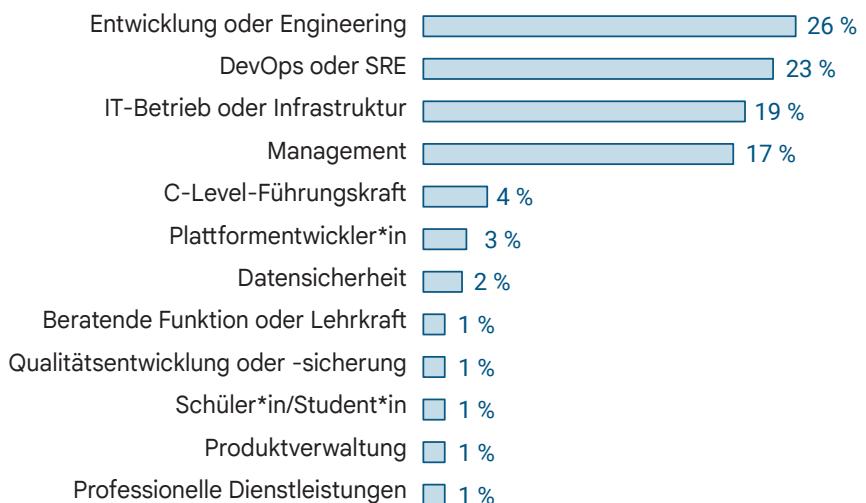
# Firmenbezogene Merkmale

## Position

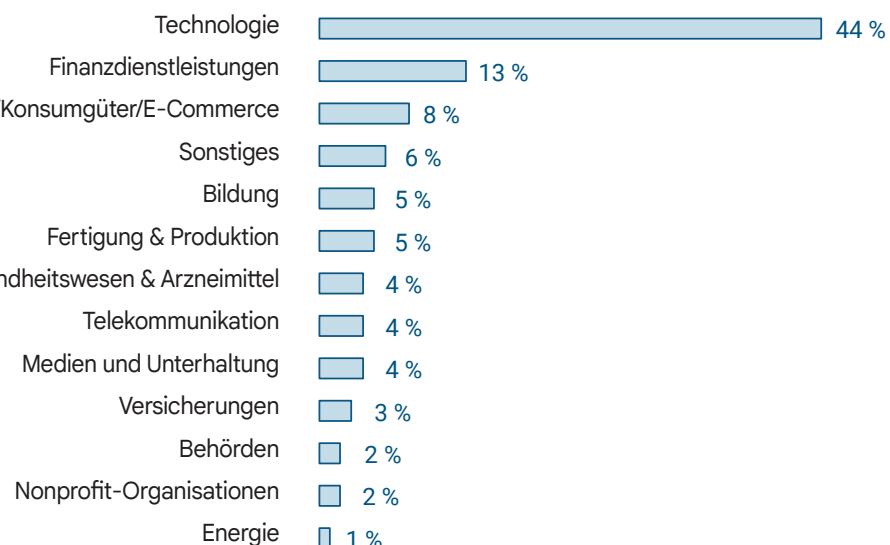
Bei 85 % der Teilnehmenden handelt es sich um Personen, die entweder zu Entwicklungs- oder Engineering-Teams (26 %), DevOps- oder SRE-Teams (23 %), operativen IT-Teams oder Infrastrukturteams (19 %) gehören oder im Managementbereich arbeiten (17 %). Der Anteil der Teilnehmenden, die in operativen IT-Teams oder Infrastrukturteams arbeiten (19 %), beträgt mehr als das Doppelte des Anteils im letzten Jahr (9 %). Bei den C-Level-Führungskräften (9 % in 2021 gegenüber 4 % 2022) und den Mitarbeitenden im Bereich professionelle Dienstleistungen (4 % in 2021 gegenüber 1 % 2022) kam es dagegen relativ zum Vorjahr zu einem ausgeprägten Rückgang.

## Branche

Wie schon frühere „State of DevOps“-Berichte gezeigt haben, arbeiten auch dieses Jahr die meisten Teilnehmenden in der Technologiebranche, gefolgt von der Finanzdienstleistungsbranche, sonstigen Branchen und dem Einzelhandel.

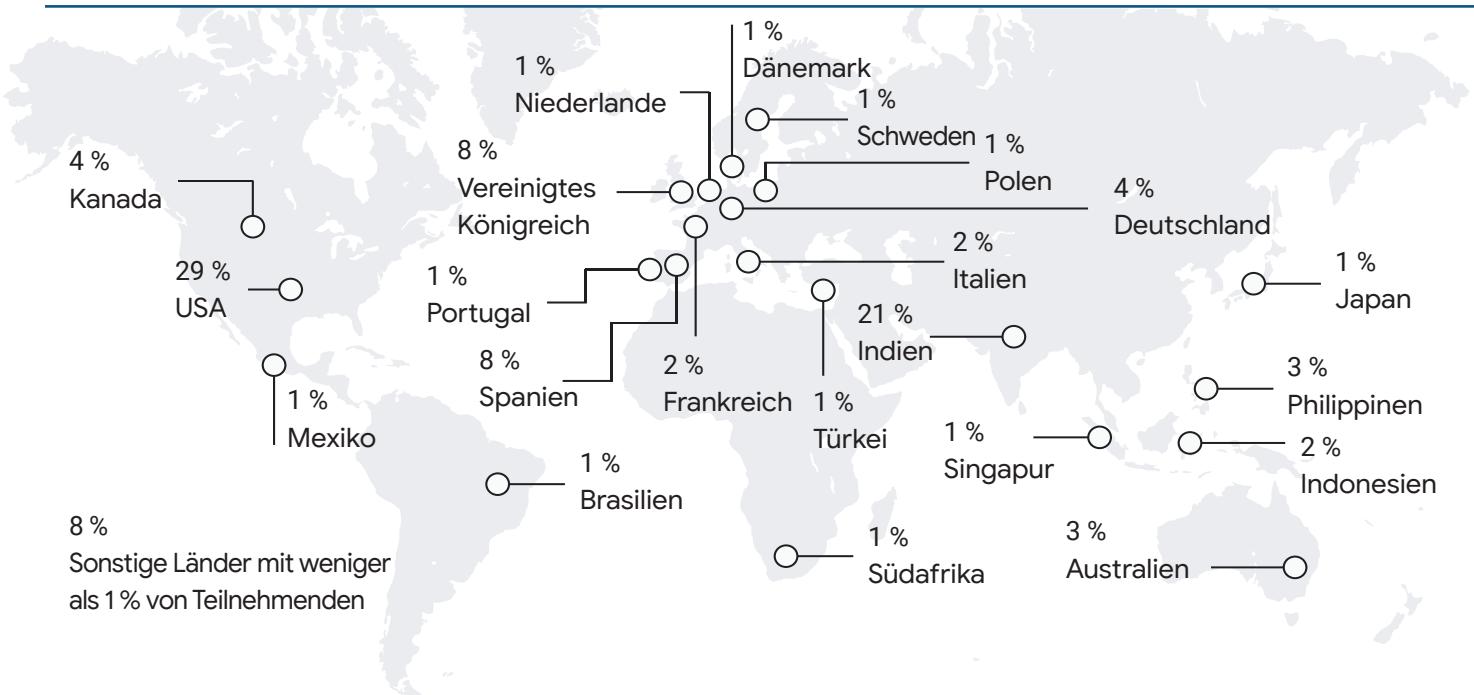


## 2022



## Region

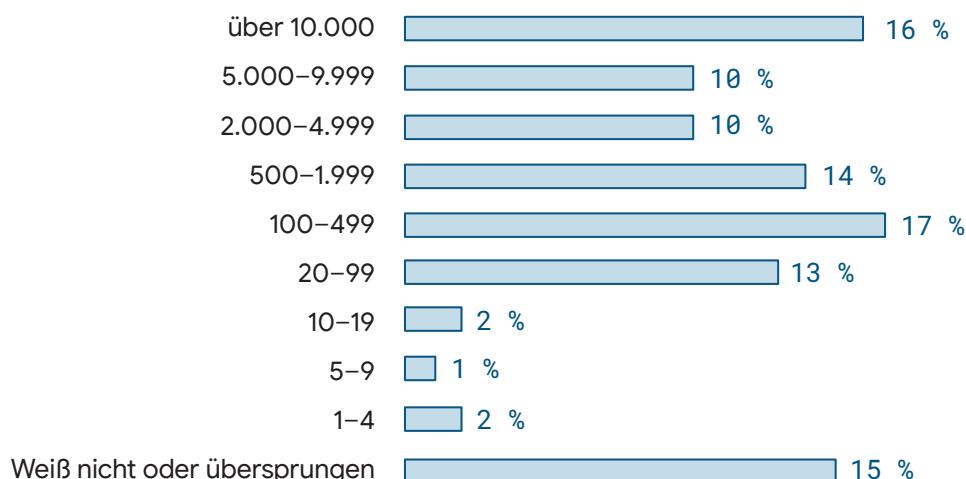
Dieses Jahr haben wir die Teilnehmenden gebeten, anzugeben, aus welchem Land (statt der Region) sie kommen. Die Regionen, die oft einem Kontinent entsprachen, erschienen uns ein wenig zu unpräzise, um die Zusammensetzung der Teilnehmenden wirklich zu verstehen. Wir erhielten Antworten von Teilnehmenden aus über 70 Ländern; 89 % der Teilnehmerinnen und Teilnehmer stammen aber aus 22 Ländern.

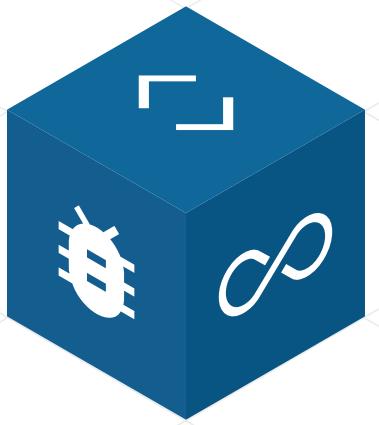


## Anzahl der Mitarbeiter

Im Einklang mit früheren „State of DevOps“-Befragungen stammen die Teilnehmenden auch diesmal aus Organisationen verschiedener Größen. 22 % der Teilnehmenden stammen aus Unternehmen mit mehr als 10.000 Mitarbeitenden, 7 % aus Unternehmen mit 5.000 bis 9.999 Mitarbeitenden. Weitere 15 % der Teilnehmenden arbeiten für

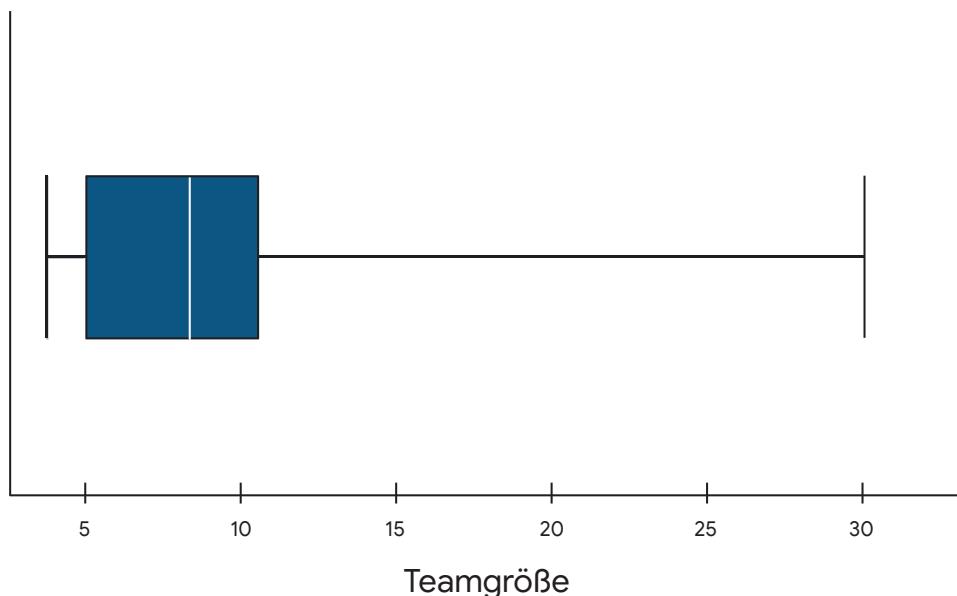
Organisationen mit 2.000–4.999 Beschäftigten. Außerdem lag die Zahl der Teilnehmenden aus Organisationen mit 500 bis 1.999 Mitarbeitenden bei 13 %, mit 100 bis 499 Mitarbeitenden bei 15 % und mit 20 bis 99 Mitarbeitenden bei 15 %. Dieses Jahr haben wir den befragten Personen auch angeboten, die Frage nach der Größe ihrer Organisation mit „Ich weiß nicht“ zu beantworten; 15 % der Teilnehmenden gaben entweder an, es nicht zu wissen, oder übersprangen die Frage.





## Teamgröße

Dieses Jahr haben wir die Teilnehmerinnen und Teilnehmer angeben lassen, wie viel Mitglieder ihr Team ungefähr hat. 25 % der Teilnehmenden arbeiteten in Teams mit 5 oder weniger Mitgliedern. 50 % der Teilnehmenden arbeiteten in Teams mit 8 oder weniger Mitgliedern. 75 % der Teilnehmenden arbeiteten in Teams mit 12 oder weniger Mitgliedern.

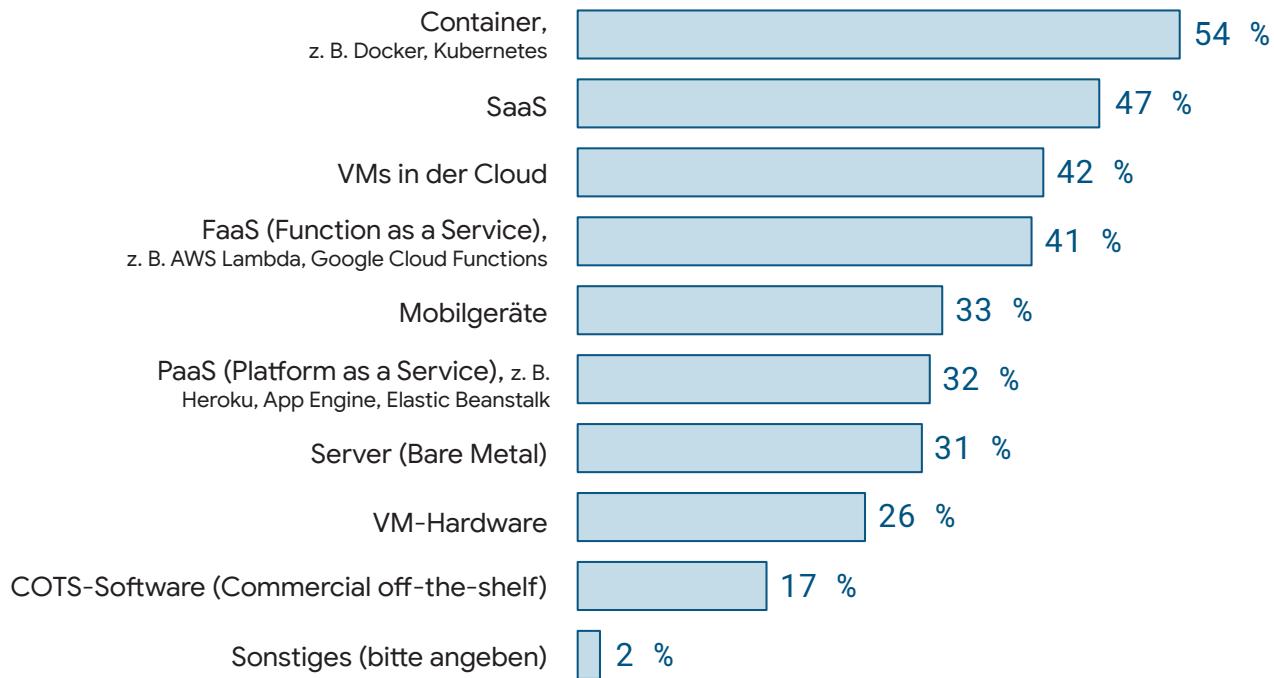


## Bereitstellungsziel

2021 war das erste Jahr, in dem wir danach gefragt haben, wo die befragten Personen den Dienst oder die Anwendung bereitstellen, an dem bzw. der sie primär arbeiten. Zu unserer Überraschung hieß das Bereitstellungsziel Nummer 1 Container.

Dieses Jahr war das nicht anders, aber der Anteil (54 %) war geringer als letztes Jahr (64 %).

Außerdem haben wir weitere Antwortmöglichkeiten ergänzt, um es den Teilnehmenden zu ermöglichen, genau zu benennen, wo sie Bereitstellungen vornehmen.



# 07

## Schlussgedanken



Derek DeBellis

In jedem Jahr, in dem wir diesen Bericht erstellen, bemühen wir uns, präzise darzulegen, wie bestimmte Praktiken und Fähigkeiten zu geschäftskritischen Ergebnissen wie der Firmenleistung führen. Wir bewerten die Replizierbarkeit vieler Effekte aus vorherigen Berichten und erweitern den Umfang unserer Arbeit dahingehend, dass auch neue Prioritäten in puncto DevOps berücksichtigt werden. Dieses Jahr haben wir die Befragung und die Analysen so modifiziert, dass wir einen tieferen Einblick in das Thema Sicherheitspraktiken gewinnen konnten. Außerdem haben wir unseren statistischen Modellierungsansatz geändert, um die Konditionalität bzw. die Abhängigkeiten bestimmter Effekte zu untersuchen. Wir haben auch nach neuen Möglichkeiten gesucht, um die Welt der Softwarebereitstellung und der operativen Leistung zu beschreiben.

In vielerlei Hinsicht sind die Erkenntnisse, die sich dieses Jahr herauskristallisiert haben, ein Echo der vorherigen Jahre: Technische Fähigkeiten bauen aufeinander auf und sorgen so für eine bessere Leistung, die Verwendung der Cloud birgt viele Vorteile, eine gesunde Kultur und Flexibilität am Arbeitsplatz führen zu einer besseren Firmenleistung und Burnout-Erkrankungen der Mitarbeitenden halten Organisationen davon ab, ihre Ziele zu erreichen. Das Analysieren von Interaktionen, die wir ausdrücklich in das Modell aufgenommen haben, hat uns geholfen, die Bedingungen zu verstehen, unter denen bestimmte

Effekte auftreten können. Beispielsweise scheint sich die Softwarebereitstellungsleistung nur dann positiv auf die Firmenleistung auszuwirken, wenn die operative Leistung (die Zuverlässigkeit) hoch ist, was die Schlussfolgerung zulässt, dass beides erforderlich ist, um als Organisation erfolgreich zu sein. Wir haben auch ein paar überraschende Entdeckungen gemacht, auf die wir in einem eigenen Abschnitt eingegangen sind.

Wir möchten allen danken, die an der diesjährigen Umfrage teilgenommen haben. Wir hoffen, dass unsere Forschung Ihnen und Ihrer Organisation dabei hilft, bessere Teams zu schaffen, bessere Software zu entwickeln und dennoch eine gesunde Work-Life-Balance zu wahren.

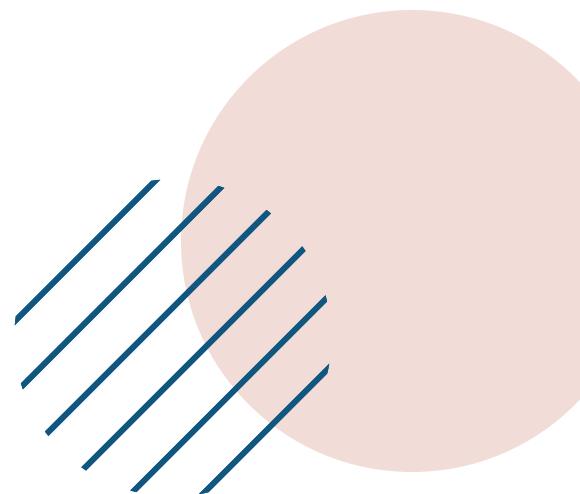


# 08

## Danksagungen

Eine Unzahl von leidenschaftlichen Mitwirkenden hat den diesjährigen Bericht erst möglich gemacht. Das Entwerfen der Fragen, das Durchführen von Analysen, das Schreiben und Bearbeiten von Text und das Strukturieren des Berichts sind nur einige der Arten, auf die unsere Kolleginnen und Kollegen uns bei der Umsetzung dieses großen Projekts geholfen haben. Die Autoren möchten all diesen Personen für ihre Beiträge und ihre Hilfe für den diesjährigen Bericht danken. Unsere Danksagungen sind alphabetisch aufgeführt.

Scott Aucoin	Eric Maxwell
Alex Barrett	John Speed Meyers
James Brookbank	Steve McGhee
Kim Castillo	Jacinda Mein
Lolly Chessie	Alison Milligan
Jenna Dailey	Pablo Pérez Villanueva
Derek DeBellis	Claire Peters
Rob Edwards	Connor Poske
Dave Farley	Dave Stanke
Christopher Grant	Dustin Smith
Mahshad Haeri	Seth Vargo
Nathen Harvey	Daniella Villalba
Damith Karunaratne	Brenna Washington
Todd Kulesza	Kaiyuan „Frank“ Xu
Amanda Lewis	Nicola Yap
Ian Lewis	



# 09

## Autoren



### Claire Peters

Claire Peters arbeitet bei Google als User Experience Researcher. Ihre Forschung umfasst verschiedene Aspekte und Ausprägungen von DORA in angewandten Situationen. Sie forscht in Cloud Application Modernization Program (CAMP) von Google und befasst sich mit DORA-gestützter Kundeninteraktionen sowie Four-Keys-bezogenen Tools. Sie verfolgt dabei das Ziel, Teams und Einzelpersonen dabei zu unterstützen, die DORA-Prinzipien effektiver bei ihrer täglichen Arbeit anzuwenden. Claire Peters ist außerdem Mitglied des DORA-Forschungsteams und erstellt jedes Jahr die DORA-Umfrage und den State of DevOps-Bericht. Sie hat einen MA in Applied Cultural Analysis an der University of Copenhagen erworben.



### Dave Farley

Dave Farley ist der Managing Director und Gründer von Continuous Delivery Ltd und Creator des YouTube-Kanals „Continuous Delivery“. Dave Farley ist Mitautor des Bestsellers „Continuous Delivery“. Er ist außerdem Autor des Bestsellers „Modern Software Engineering: Doing What Works to Build Better Software Faster“. Er ist Mitautor des „Reactive Manifesto“ und Gewinner des Duke Award für das Open-Source-Projekt „LMAX Disruptor“. Dave Farley ist ein Pioneer der Continuous Delivery sowie Vordenker und praktizierender Experte in Sachen CD, DevOps, testgetriebener Entwicklung und Softwaredesign. Seit Langem hilft er Teams, leistungsstärker zu werden, bringt Organisationen auf Erfolgskurs und entwickelt herausragende Software. Er ist auch auf [Twitter](#) und [YouTube](#) und hat einen [Blog](#) sowie eine [Website](#).



## Daniella Villalba

Daniella Villalba ist Forscherin im Bereich Nutzererfahrung für das DORA-Projekt. Sie konzentriert sich darauf, die Faktoren zu ermitteln, die Entwickelnde glücklich und produktiv machen. Vor ihrer Zeit bei Google untersuchte Daniella Villalba die Vorteile von Meditationstraining, die psychosozialen Faktoren, die sich darauf auswirken, welche Erfahrungen Studierende machen, Erinnerungen von Augenzeuginnen und -zeugen und falsche Geständnisse. Sie hat einen Doktortitel in Experimental Psychology von der Florida International University.



## Dave Stanke

Dave Stanke ist Developer Relations Engineer bei Google und berät Kundinnen und Kunden, welche Best Practices für sie am besten geeignet sind, um DevOps und SRE einzuführen. Während seiner beruflichen Laufbahn hat er in allen möglichen Bereichen gearbeitet, darunter als CTO eines Start-ups, als Produktmanager, im Kundensupport, als Softwareentwickler, Systemadministrator und Grafikdesigner. Er hat einen Master of Science-Abschluss in Technology Management von der Columbia University.



## Derek DeBellis

Derek DeBellis arbeitet bei Google als Quantitative User Experience Researcher. Dort konzentriert er sich auf die Umfrageforschung und Loganalyse und versucht herauszufinden, wie sich Konzepte messen lassen, die für die Produktentwicklung elementar sind. Derek DeBellis hat vor Kurzem Veröffentlichungen zu folgenden Themen herausgebracht: Interaktionen zwischen Menschen und KI, Auswirkungen des Beginns einer Covid-Erkrankung auf die Raucherentwöhnung, Berücksichtigung von NLP-Fehlern bei der Entwicklung und die Rolle der Nutzererfahrung in Diskussionen zum Datenschutz.



## Eric Maxwell

Eric Maxwell leitet den Bereich DevOps Digital Transformation von Google und berät die erfolgreichsten Unternehmen der Welt, wie sie noch besser werden können – fortlaufend Stück für Stück. Er hat die erste Hälfte seines Arbeitslebens damit verbracht, als Entwickler zu arbeiten und alles mögliche zu automatisieren sowie Empathie für andere Fachkräfte zu entwickeln. Eric Maxwell hat zusammen mit anderen das Cloud Application Modernization Program (CAMP) von Google entwickelt, ist Mitglied des DORA-Kernteams und Autor des DevOps Enterprise Guidebook. Vor seiner Zeit bei Google hat er zusammen mit seinen tollen Kollegen bei Chef Software dort vieles bewirkt.



## John Speed Meyers

John Speed Meyers arbeitet als Security Data Scientist bei Chainguard, einem Start-up, das sich mit der Sicherheit der Softwarelieferkette befasst. Er hat bereits unter anderem zu folgenden Themen geforscht: Sicherheit der Softwarelieferkette, Sicherheit von Open-Source-Software und Reaktionen der Welt auf die zunehmende militärische Macht Chinas. Vorher hat John Speed Meyers bei In-Q-Tel, der RAND Corporation und dem Center for Strategic and Budgetary Assessments gearbeitet. Er hat einen Doktortitel in Policy Analysis von der Pardee RAND Graduate School, einen Master in Public Affairs (MPA) von der School of Public and International Affairs der Princeton University und einen BA in International Relations von der Tufts University.



## Kaiyuan „Frank“ Xu

Kaiyuan „Frank“ Xu arbeitet bei Google als Quantitative User Experience Researcher. Er analysiert Log- und Umfragedaten, um die Nutzungsmuster und das Feedback von Nutzenden in Bezug auf Google Cloud-Produkte zu verstehen und so unsere Produkte für Entwickelnde noch besser zu machen. Vor seiner Zeit bei Google hat Kaiyuan „Frank“ Xu jahrelang bei Microsoft qualitative und quantitative Nutzerstudien für Azure- und Power Platform-Produkte durchgeführt. Er hat einen Master of Science-Abschluss in Human Centered Design and Engineering von der University of Washington.



## Nathen Harvey

Nathen Harvey ist Developer Relations Engineer bei Google und verbringt sein Arbeitsleben damit, Teams dabei zu helfen, ihr Potenzial voll auszuschöpfen und gleichzeitig ihre Technologien so auszuwählen, dass sie auf die Geschäftsergebnisse abgestimmt sind. Er hatte das Privileg, mit einigen der besten Teams und Open-Source-Communities zusammenzuarbeiten und sie beim Anwenden der Prinzipien und Best Practices von DevOps und SRE zu unterstützen. Nathen Harvey hat außerdem gemeinsam mit einer Kollegin *97 Things Every Cloud Engineer Should Know*, O'Reilly 2020 redigiert und war auch am Schreiben der Texte beteiligt.



## Todd Kulesza

Todd Kulesza arbeitet bei Google als User Experience Researcher und untersucht dort, wie Entwicklerinnen und Entwickler von Software heutzutage arbeiten und wie sie in Zukunft vielleicht noch bessere Arbeit leisten könnten. Er hat einen Doktortitel in Computer Science von der Oregon State University.

# 10

# Methodik

## Forschungsdesign

In dieser Studie haben wir ein theoriebasiertes Querschnittsdesign eingesetzt, dass als **inferentiell-prognostisch** bekannt ist – eine der am weitesten verbreiteten Studiendesignarten, was die heutige Forschung in der Geschäfts- und Technologiewelt angeht. Mit einem inferentiell-prognostischen Design wird gearbeitet, wenn ein rein experimentelles Design nicht praktisch oder nicht möglich ist.

## Zielpopulation und Befragung eines repräsentativen Querschnitts

Die Zielpopulation dieser Umfrage bestand aus Fachkräften und Führungskräften, die in den Bereichen Technologie und Transformation oder in verwandten Bereichen arbeiten, und insbesondere aus Personen, die mit DevOps vertraut sind. Wir haben die Umfrage über E-Mail-Listen und in den sozialen Medien beworben, haben dafür Onlinewerbung geschaltet, sie in einem Online-Panel bereitgestellt und die Teilnehmenden der Umfrage gebeten, sie mit Personen aus ihrem Netzwerk zu teilen (Schneeballauswahl).

## Latente Konstrukte erstellen

Wir haben unsere Hypothesen und Konstrukte so oft wie möglich mit zuvor validierten Konstrukten formuliert. Wir haben auf Grundlage von Theorien, Definitionen und Fachwissen neue Konstrukte entwickelt. Anschließend haben wir zusätzliche Maßnahmen ergriffen, um die Absicht der Befragung klarzustellen, damit die darin erhobenen Daten eine hohe Wahrscheinlichkeit aufweisen, dass sie verlässlich und gültig sind.<sup>1</sup>

## Unterschiede zwischen den leistungsschwachen und leistungsstarken Mitarbeitenden berechnen

Im Abschnitt „Wie schneiden Sie ab?“ haben wir die leistungsschwachen und leistungsstarken Mitarbeitenden anhand der vier Messwerte der Bereitstellungsleistung verglichen. Dazu haben wir eine einfache Methode verwendet. Nehmen wir die Bereitstellungshäufigkeit als Beispiel. Im Cluster mit hoher Leistung wird on demand bereitgestellt (d. h. mehrmals pro Tag). Wenn diese Beschäftigten durchschnittlich vier Bereitstellungen pro Tag durchführen, sind das 1460 Bereitstellungen pro Jahr ( $4 * 365$ ). Die leistungsschwachen Mitarbeitenden nehmen dagegen einmal im Monat bis einmal alle sechs Monate eine Bereitstellung vor, sodass sich eine durchschnittliche Bereitstellungshäufigkeit von einmal alle 3,5 Monate und von etwa 3,4 Bereitstellungen pro Jahr ( $12/3,5$ ) ergibt. Wir errechnen das Verhältnis (1.460 Bereitstellungen der Leistungsstarken geteilt durch 3,4 Bereitstellungen der Leistungsschwachen) und erhalten einen Faktor von 417. Dieser Ansatz wird dann für die anderen Messwerte der Bereitstellungsleistung verallgemeinert.

<sup>1</sup> Churchill Jr, G. A. „A paradigm for developing better measures of marketing constructs“, Journal of Marketing Research 16:1, (1979), 64–73.

# Methoden der statistischen Analyse

## Clusteranalyse

Für die beiden Clusteranalyselösungen, die wir im Abschnitt „Wie schneiden Sie ab?“ dargestellt haben, haben wir die hierarchische Clusteranalyse mit einer Version des agglomerativen Verfahrens nach Ward<sup>2</sup> verwendet, um einzuschätzen, wie gut verschiedene Clusteranalyselösungen für die Daten geeignet sind.

Bei den Ergebnissen der ersten Clusteranalyse haben wir nach Clustern für die Antworten gesucht, die sich um die Bereitstellungshäufigkeit, die Vorlaufzeit für Änderungen, die Zeit bis zur Dienstwiederherstellung und die Änderungsfehlerquote drehten. Dieses Jahr haben wir drei Cluster identifiziert, nachdem wir 14 verschiedene hierarchische Clusteranalyselösungen ausgewertet und 30 verschiedene Indizes verwendet hatten, um die Anzahl der Cluster zu bestimmen.<sup>3</sup>

Bei der zweiten Clusteranalyse haben wir dieselbe Methodik verwendet, aber sie für andere Größen in den Daten verwendet. Wir wollten nach häufigen Antwortmustern suchen (d. h. Clustern),

was den Durchsatz (eine Kombination aus Bereitstellungshäufigkeit und Vorlaufzeit), die operative Leistung (Zuverlässigkeit) und die Stabilität (eine Kombination aus Zeit bis zur Dienstwiederherstellung und Änderungsfehlerquote) angeht. Wir haben außerdem verschiedene Clusteranalyse-Algorithmen ausprobiert, um herauszufinden, wie empfindlich unsere Ergebnisse auf unseren Ansatz reagierten. Uns ist zwar keine etablierte Methode bekannt, um diese Empfindlichkeit zu quantifizieren, aber die Cluster, die sich ergeben haben, tendierten dazu, ähnliche Merkmale aufzuweisen:

## Messmodell

Vor der Analyse haben wir Konstrukte ermittelt, indem wir die explorative Faktorenanalyse mit der Hauptkomponentenanalyse und dem Varimax-Rotationsverfahren kombiniert haben.<sup>4</sup> Wir bestätigten statistische Tests zur konvergenten und divergenten Validität und Zuverlässigkeit unter Verwendung der durchschnittlich erfassten Varianz (Average Variance Extracted, AVE), der Korrelation, des cronbachschen Alpha<sup>5</sup>, von rhoA<sup>6</sup>, dem Heterotrait-Monotrait-Verhältnis<sup>5,7</sup> und der zusammengesetzten Zuverlässigkeit.

<sup>2</sup> Murtagh, Fionn und Pierre Legendre. „Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion?“ Journal of classification 31.3 (2014): 274–295.

<sup>3</sup> Charrad M., Ghazzali N., Boiteau V., Niknafs A. (2014). „NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set.“, „Journal of Statistical Software, 61(6), 1–36.“, „URL <http://www.jstatsoft.org/v61/i06/>“.

<sup>4</sup> Straub, D., Boudreau, M. C. & Gefen, D. (2004). Validation guidelines for IS positivist research. Communications of the Association for Information systems, 13(1), 24.

<sup>5</sup> Nunnally, J.C. Psychometric Theory. New York: McGraw-Hill, 1978.

<sup>6</sup> Hair Jr, Joseph F., et al. „Partial least squares structural equation modeling (PLS-SEM) using R: A workbook.“ (2021): 197.

<sup>7</sup> Brown, Timothy A., and Michael T. Moore. „Confirmatory factor analysis.“ Handbook of structural equation modeling 361 (2012): 379.

## Strukturelle Gleichungsmodellierung

Wir haben die Strukturgleichungsmodelle (Structural Equation Models, SEM) mit der PLS-Analyse (Partial-Least-Squares) getestet<sup>8</sup>, wobei es sich um eine korrelationsbasierte Strukturgleichungsmodellierung handelt.

## Analyse des zweiten Clustermodells

Um zu verstehen, wie sich die Mitgliedschaft in einem Cluster prognostizieren lässt, haben wir eine multinomiale logistische Regression verwendet.<sup>9</sup>

Wir haben diesen Ansatz gewählt, da wir versucht haben, die Clustermitgliedschaft von verschiedenen Personen vorherzusagen, was in diesem Fall die Arbeit mit ungeordneten kategorischen Daten mit mehr als zwei Kategorien bedeutete. Um die Ergebnisse nachzuvollziehen, die die jeweilige Clustermitgliedschaft prognostiziert hatte, haben wir für jedes Ergebnis eine lineare Regression verwendet (Burnout, ungeplante Arbeit und Firmenleistung).



<sup>8</sup> Hair Jr, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2021). „A primer on partial least squares structural equation modeling (PLS-SEM).“ Sage publications.

<sup>9</sup> Ripley, Brian, William Venables und Maintainer Brian Ripley. „Package ‘nnet’.“ R package version 7.3–12 (2016): 700.

# 11

## Weitere Informationen

Treten Sie der DORA-Community bei, um die Softwarebereitstellung und die Betriebsleistung zu verbessern und gemeinsam mit anderen zu lernen.

<http://dora.community>

Weitere Informationen zu den „Four Keys“-Messwerten

<https://goo.gle/four-keys>

Weitere Information zu DevOps-Ressourcen

<https://goo.gle/devops-capabilities>

Weitere Information zum Einsatz von DORA-Praktiken in Ihrem Unternehmen

<https://goo.gle/enterprise-guidebook>

Ressourcen zum Site Reliability Engineering (SRE)

<https://sre.google>

<https://goo.gle/enterprise-roadmap-sre>

DevOps-Schnelltest

<https://goo.gle/devops-quickcheck>

DevOps-Forschungsprogramm

<https://goo.gle/devops-research>

E-Book über die Gewinnerinnen und Gewinner der Google Cloud DevOps Awards: Lernen Sie von anderen Firmen, die die DORA-Praktiken bereits implementiert haben.

<https://goo.gle/devops-awards>

Weitere Informationen zum Google Cloud Application Modernization Program (CAMP)

<https://goo.gle/3daLa9s>

Daten und DORA-Messwerte nutzen, um Technologieprozesse zu transformieren

<https://goo.gle/3Doh8Km>

Whitepaper lesen: „The ROI of DevOps Transformation: How to quantify the impact of your modernization initiatives“ by Forsgren, N., Humble, J., & Kim, G. (2018).  
<https://goo.gle/3qECIIh>

Buch: Accelerate: *The science behind devops: Building and scaling high performing technology organizations.* IT Revolution.  
<https://itrevolution.com/book/accelerate>

Weitere Informationen zum SLSA-Framework (Supply Chain Levels for Software Artifacts)  
<https://slsa.dev>

Weitere Informationen zum Secure Software Development Framework (NIST SSDF)  
<https://goo.gle/3qBXLWk>

Weitere Informationen zur DevOps-Kultur:  
Unternehmenskultur nach Westrum  
<https://goo.gle/3xq7KBV>

Weitere Informationen zur Open Source Security Foundation  
<https://openSSF.org/>

Weitere Informationen zu in-toto  
<https://in-toto.io/>

Weitere Informationen zur Software Bill of Materials von NTIA.gov  
<https://www.ntia.gov/SBOM>

Cybersecurity: Federal Response to SolarWinds and Microsoft Exchange Incidents  
<https://www.gao.gov/products/gao-22-104746>

Weitere Informationen zu Sicherheitsscans auf Anwendungsebene als Teil von CI/CD  
<https://go.dev/blog/survey2022-q2-results#security>

Und zu guter Letzt: „State of DevOps“-Berichte der Vorjahre Liste unter:

[https://goo.gle/dora-sodrs:](https://goo.gle/dora-sodrs)  
[Bericht „Accelerate State of DevOps 2014“](https://www.acceleratestateofdevops.com/2014-report.html)  
[Bericht „Accelerate State of DevOps 2015“](https://www.acceleratestateofdevops.com/2015-report.html)  
[Bericht „Accelerate State of DevOps 2016“](https://www.acceleratestateofdevops.com/2016-report.html)  
[Bericht „Accelerate State of DevOps 2017“](https://www.acceleratestateofdevops.com/2017-report.html)  
[Bericht „Accelerate State of DevOps 2018“](https://www.acceleratestateofdevops.com/2018-report.html)  
[Bericht „Accelerate State of DevOps 2019“](https://www.acceleratestateofdevops.com/2019-report.html)  
[Bericht „Accelerate State of DevOps 2021“](https://www.acceleratestateofdevops.com/2021-report.html)