

# Das Mindset von DevOps: Accelerate

24 Schlüsselkompetenzen, um leistungsstarke Technologieunternehmen zu entwickeln und zu skalieren

Bearbeitet von  
Von Nicole Forsgren, Jez Humble, und Gene Kim

1. Auflage 2019. Buch. 216 S. Hardcover  
ISBN 978 3 8006 5963 0  
Format (B x L): 16,0 x 24,0 cm

[Wirtschaft > Management > Unternehmensführung](#)

Zu [Inhalts-](#) und [Sachverzeichnis](#)

schnell und portofrei erhältlich bei

  
DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung beek-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

diese Items für ein *Team* und nicht für ein Unternehmen abgefragt haben. Wir haben uns, während wir die Umfrageitems entwickelt haben, dafür entschieden – eine Abweichung von Westrums Original-Framework – , weil Unternehmen sehr groß sein können und viele verschiedene Unternehmenskulturen haben können. Darüber hinaus können Menschen für ihr Team genauer antworten als für ihr Unternehmen. Dadurch können wir bessere Messergebnisse erhalten.

### Latente Konstrukte helfen, vor fehlerhaften Daten zu schützen

Das müssen wir etwas verdeutlichen. Latente Konstrukte, *die periodisch statistisch wieder getestet werden und gute psychometrische Eigenschaften aufweisen* schützen uns vor fehlerhaften Daten.

Wie bitte? Wir erklären es Ihnen.

Im vorherigen Abschnitt haben wir über Validität und Reliabilität gesprochen – statistische Tests, die wir durchführen können, um sicherzustellen, dass die Umfrageitems, die ein latentes Konstrukt messen, zusammengehören. Wenn unsere Konstrukte alle diese statistischen Tests bestehen, sagen wir, sie »weisen gute psychometrische Eigenschaften auf«. Es ist eine gute Idee, das periodisch erneut zu überdenken, um sicherzustellen, dass sich nichts geändert hat, besonders, wenn Sie eine Änderung im System oder der Umgebung vermuten.

In dem Beispiel der Unternehmenskultur sind alle diese Items gute Messgrößen des Konstrukts. Hier ist ein weiteres Beispiel eines Konstrukts, bei dem Tests die Möglichkeiten hervorhoben, unsere Messgröße zu verbessern. In diesem Fall interessierte uns die Information über Ausfälle. Die Items waren:

- Wir werden zuerst durch Berichte von Kunden über Ausfälle informiert.
- Wir werden zuerst durch das NOC über Ausfälle informiert.
- Wir bekommen Fehlermeldungen durch Logging und Monitoring Systeme.
- Wir überwachen die Systemgesundheit basierend auf einer Warnschwelle (außer die CPU überschreitet 90 %).
- Wir überwachen die Systemgesundheit basierend auf Warnungen der Änderungsrate (außer die CPU ist in den letzten 10 Minuten um 25 % angewachsen).

Im vorangegangenen Umfragedesign haben wir mit ungefähr 20 Technikern einen Pilottest für das Konstrukt durchgeführt und die Items zusammengeführt (das heißt, sie haben dasselbe Grundkonstrukt gemessen). Als wir jedoch unsere endgültige, größere Datenerhebung abgeschlossen hatten, haben wir Tests durchgeführt, um das Konstrukt zu bestätigen. In diesen abschließenden Tests stellten wir fest, dass diese Items tatsächlich zwei verschiedene Dinge gemessen hatten. Das heißt, als wir unsere statistischen Tests durchgeführt haben, haben sie nicht ein einziges Konstrukt bestätigt, sondern deckten stattdessen zwei Konstrukte auf. Die ersten

beiden Items messen ein Konstrukt, das die »Benachrichtigung, die von außerhalb des automatisierten Prozesses kommt«, erfasst:

- Wir werden zuerst durch Berichte von Kunden über Ausfälle informiert.
- Wir werden zuerst durch das NOC über Ausfälle informiert.

Das zweite Item-Set erfasst ein anderes Konstrukt – »Benachrichtigungen, die von Systemen kommen oder proaktive Fehlerbenachrichtigung:

- Wir bekommen Fehlermeldungen durch Logging und Monitoring Systeme.
- Wir überwachen die Systemgesundheit basierend auf Warnschwellen (außer die CPU überschreitet 90 %).
- Wir überwachen die Systemgesundheit basierend auf Warnungen der Änderungsrate (außer die CPU ist in den letzten 10 Minuten um 25 % angewachsen).

Wenn wir unsere Teilnehmer nur mit einer einzigen Frage gefragt hätten, ob sie Fehler überwachen, hätten wir die Bedeutung übersehen, zu erfassen, *wo* diese Benachrichtigungen herkommen. Wenn eine der Informationsquellen darüber hinaus, ihr Verhalten ändert, werden unsere statistischen Tests das merken und uns alarmieren. Dasselbe Konzept kann auch bei Systemdaten angewandt werden. Wir können verschiedene Messgrößen aus unseren Systemen nehmen, um Systemverhalten zu erfassen und diese Messgrößen können unsere Validitätstests bestehen. Wir sollten jedoch weiterhin periodische Tests bei diesen Messgrößen durchführen, da sie sich ändern können. Unsere Forschung hat ergeben, dass dieses zweite Konstrukt, proaktive Fehlerbenachrichtigung, eine technische Kompetenz ist, die ein Indikator für die Performance der Softwarebereitstellung ist.

## DIE FACHBUCHHANDLUNG

### Wie man latente Konstrukte auf Systemdaten anwendet

Einige dieser Vorstellungen über latente Konstrukte gelten ebenfalls für Systemdaten: Sie helfen uns, ungültige Daten zu vermeiden, indem wir mehrere Messgrößen anwenden, um nach ähnlichen Mustern oder Verhaltensweisen zu suchen. Außerdem helfen sie uns dabei, zu durchdenken, was wir wirklich versuchen abzubilden. Nehmen wir beispielsweise an, wir möchten die Systemperformance messen. Wir können einfach Reaktionszeiten einiger Aspekte des Systems sammeln. Um nach ähnlichen Mustern in den Daten zu suchen, können wir mehrere Datenteile aus unserem System sammeln, die uns helfen, die Reaktionszeit zu verstehen. Um darüber nachzudenken, was wir wirklich versuchen zu messen – die Performance – können wir über verschiedene Aspekte der Performance nachdenken und wie diese noch in den System-Kennzahlen reflektiert werden könnte. Wir stellen vielleicht fest, dass uns eine konzeptuelle Messgröße der Systemperformance interessiert, die sehr schwer direkt zu messen ist und besser mit mehreren verbundenen Messgrößen erfasst wird.

Es gibt eine wichtige Anmerkung: Alle Messgrößen sind Stellvertreter. Das heißt, sie repräsentieren für uns eine Vorstellung, selbst wenn wir sie nicht bewusst erkennen. Das gilt sowohl für Systemdaten als auch für Umfragedaten. Wir können beispielsweise die Reaktionszeit als Vertreter für die Performance unseres Systems nehmen.

Wenn nur einer der Datenpunkte als Barometer genutzt wird und dieser dann auch noch ungültig ist – oder ungültig wird –, werden wir es nicht erfahren. Eine Änderung des Quellcodes, der Kennzahlen sammelt, kann beispielsweise eine Messgröße beeinträchtigen. Wenn nur diese einzige Messgröße erfasst wird, ist die Wahrscheinlichkeit, dass wir diese Änderung bemerken, niedrig. Wenn wir jedoch mehrere Kennzahlen sammeln, können wir die Änderung eher wahrnehmen. Latente Konstrukte bieten uns einen Mechanismus, um uns gegen schlechte Messgrößen oder Schwachstellen zu schützen. Das gilt sowohl für Umfrage- als auch Systemdaten.



**beck-shop.de**  
DIE FACHBUCHHANDLUNG

## KAPITEL 14

# Warum eine Umfrage?

Jetzt wissen wir, dass unsere Umfragedaten vertrauenswürdig sind. Das heißt, wir haben eine angemessene Sicherheit, dass die Daten aus unseren gut konzipierten und vielfach getesteten psychometrischen Umfrage-Konstrukten uns das sagen, was wir glauben, dass sie uns sagen. Warum nutzen wir also eine Umfrage? Warum nutzt irgendjemand sonst eine Umfrage? Teams, die die Performance ihrer Softwarebereitstellung verstehen möchten, beginnen oft damit, ihren Bereitstellungsprozess mit Werkzeugen auszustatten und eine Toolchain zu nutzen, um Daten zu erhalten (wir nennen die auf diese Weise erhobenen Daten in diesem Buch »Systemdaten«). Tatsächlich bieten einige Werkzeuge auf dem Markt die Analyse von Aspekten wie Durchlaufzeit. Warum würde also jemand Daten aus Umfragen erheben wollen und nicht einfach aus der Toolchain?

Es gibt einige Gründe, die für Umfragedaten sprechen. In diesem Kapitel stellen wir kurz einige von ihnen vor:

1. Umfragen ermöglichen Ihnen, Daten schnell zu erheben und zu analysieren.
2. Das Gesamtpaket mit Systemdaten zu messen, ist schwierig.
3. Ausschließlich mit Systemdaten zu messen, ist schwierig.
4. Sie können Umfragedaten vertrauen.
5. Einige Dinge können nur mithilfe von Umfragen gemessen werden.

## Umfragen ermöglichen Ihnen, Daten schnell zu erheben und zu analysieren

Häufig ist der ausschlaggebende Grund für Umfragen, dass die Daten schnell und einfach erhoben werden können. Das gilt besonders für neue oder einmalige Datenerhebungen, oder für Datenerhebungen, die über Unternehmensgrenzen hinweg gehen. Die Forschung in diesem Buch wurde zu vier verschiedenen Zeiten durchgeführt.

Jedes Mal haben wir über einen Zeitraum von vier bis sechs Wochen Daten aus der ganzen Welt, von Tausenden Teilnehmern aus Tausenden von Unternehmen, erhoben.

Stellen Sie sich die Schwierigkeit vor (in Wirklichkeit die Unmöglichkeit), Systemdaten von so vielen Teams im gleichen Zeitraum zu erhalten. Allein die rechtlichen Freigaben würden unmöglich sein, ganz zu schweigen von den Datenspezifikationen und dem Transfer.

Aber nehmen wir einmal an, dass wir in der Lage waren, Systemdaten von einigen tausend Teilnehmern aus aller Welt in einem Zeitfenster von vier Wochen zu erheben. Der nächste Schritt ist die Datenbereinigung und -analyse. Die Datenanalyse für den State of DevOps Report dauert drei bis vier Wochen. Viele von Ihnen haben wahrscheinlich bereits mit Systemdaten gearbeitet und noch mehr von Ihnen hatten die besondere Freude (wahrscheinlich eher Mühe) Excel-Tabellen zu verknüpfen und abzugleichen. Stellen Sie sich vor, von einigen tausend Teams aus aller Welt Roh-Systemdaten (oder vielleicht Tabellen zur Kapitalplanung) zu bekommen. Stellen Sie sich die Herausforderung vor, diese Daten zu bereinigen, zu organisieren und dann zu analysieren und Ergebnisse zur Veröffentlichung innerhalb von drei Wochen zu liefern.

Zusätzlich zu der grundsätzlichen Herausforderung, die Daten zu bereinigen und die Analysen durchzuführen, besteht noch die spezielle Herausforderung, die Ihre gesamte Arbeit infrage stellen kann und wahrscheinlich die größte Beschränkung darstellt: die Daten selbst. Genauer gesagt, die zugrunde liegende *Bedeutung* der Daten selbst.

Sie haben es wahrscheinlich in Ihren eigenen Unternehmen erlebt. Verschiedene Teams können sich zu sehr unterschiedlichen (oder auch nur etwas unterschiedlichen) Messgrößen unter Verwendung derselben Bezeichnung äußern. Zwei Beispiele sind »Durchlaufzeit« (wir definieren sie als Zeit vom Code Commit bis zum Deployment) und »Zykluszeit« (einige definieren sie als Zeit, ab der die Entwicklung am Code arbeitet bis hin zu einem Code, der zum Deployment bereit ist). Diese beiden Begriffe werden jedoch häufig synonym benutzt und ziemlich oft verwechselt, obwohl sie zwei verschiedene Dinge messen.

Also was passiert, wenn das eine Team den Begriff Zykluszeit verwendet und das andere Durchlaufzeit – sie aber beide das gleiche messen? Oder wenn sie beide den Begriff Durchlaufzeit verwenden, aber zwei unterschiedliche Dinge messen? Dann haben wir die Daten erhoben und versuchen die Analyse durchzuführen, wissen aber nicht sicher, welche Variablen welche sind. Dies birgt erhebliche Mess- und Analyseprobleme.

Sorgfältig formulierte und gestaltete Umfragen, die geprüft wurden, helfen bei der Lösung des Problems. Alle Teilnehmer arbeiten nun auf Grundlage derselben Items, Wörter und Definitionen. Es spielt keine Rolle, wie sie es in ihren Unter-

nehmen nennen – es ist nur relevant, wonach sie in der Umfrage gefragt werden. Es spielt eine Rolle, was sie gefragt werden und die Qualität und Eindeutigkeit der Umfrageitems wird so viel wichtiger. Aber sobald die Umfrage formuliert wurde, ist es schneller und direkter möglich, die Daten für die Analyse zu bereinigen und vorzubereiten.

In der exakten Forschung werden zusätzliche Analysen (z. B. Varianzprüfung der gängigen Methode) durchgeführt, um sicherzustellen, dass die Umfrage selbst keine Verzerrung in die Ergebnisse gebracht hat. Die Antworten werden zwischen den frühen und späten Antworten daraufhin geprüft (siehe Anhang C).

## Das Gesamtpaket mit Systemdaten zu messen, ist schwierig

Selbst wenn Ihr System gute und nützliche Daten hergibt (wie wir aus Erfahrung wissen, ist diese Annahme ziemlich oft falsch und muss im Allgemeinen durch Ausprobieren ermittelt werden), sind die Daten selten erschöpfend. Mit anderen Worten, können Sie wirklich sicher sein, dass sie 100 Prozent des Systemverhaltens messen, an dem Sie interessiert sind?

Nehmen wir zur Veranschaulichung folgendes Beispiel. Die Autorin hat einen Teil ihrer Karriere als Performance Engineer bei IBM mit der Arbeit an Plattenspeichersystemen verbracht. Ihr Team hatte die Aufgabe, die Performance dieser Maschinen unter verschiedenen Arbeitsbelastungen festzustellen und zu optimieren, darunter Disk Read, Write, Cache und RAID-Neuaufbau. Nachdem sie einige Aktionen durchlaufen hatte, funktionierte »die Kiste« gut, und das Team besaß die Kennzahlen aller Systemlevel, um das zu beweisen. Gelegentlich hörte das Team von Kunden, dass das Gerät etwas langsam sei. Die Mitarbeiter untersuchten das dann immer – aber der erste Report oder auch zwei wurden verworfen, da sie die Bestätigung hatten, dass die Performance des Geräts gut sei – das bewiesen alle Systemlogs.

Als das Team jedoch immer weitere Berichte über eine langsame Performance bekam, wurden weitere Untersuchungen nötig. Natürlich konnten Kunden und das Umfeld auch lügen, beispielsweise, um Nachlässe aufgrund von nicht eingehaltenen Service Level Agreements (SLA) zu erhalten. Aber die Kunden- und Erfahrungsberichte wiesen ein Muster auf – sie zeigten alle eine ähnliche Langsamkeit. Obwohl die Daten, die das Team von Menschen bekam, nicht dieselbe Präzision hatten wie Systemlogs (z. B. die auf die Minute genaue Präzision bei den berichteten Reaktionszeiten gegenüber den auf die Millisekunde genauen Logdateien), hatte das Team genug Daten, um zu erkennen, wo es nach dem Problem suchen musste. Sie wiesen Muster auf und zeigten ein Signal, dem das Team folgen musste.

Also, was war es dann? Es stellte sich heraus, dass das Gerät an sich außerordentlich gut funktionierte. Das Team *hatte* alle Level des Stacks instrumentiert und erfasste

alles, was es zu erfassen gab ... *in dem Gerät*. Was das Team nicht erfasst hatte, war die Schnittstelle. Die Art und Weise, wie die Kunden mit dem Gerät interagierten, wies signifikante Abstriche bei der Leistung auf. Das Team bildete daraufhin eine kleine Gruppe, um diesen neuen Bereich zu bearbeiten, und schon bald darauf bot das gesamte System eine Spitzenleistung.

Das Team hätte nicht verstanden, was los war, wenn es nicht *Menschen* zu der Systemleistung befragt hätte. Wenn man sich die Zeit nimmt, periodische Beurteilungen durchzuführen, die die Auffassungen von Technologen einbeziehen, die Ihre Technologie entwickeln und bereitstellen, bekommt man wesentliche Erkenntnisse über die Engpässe und Beschränkungen Ihres Systems. Indem Sie jeden im Team befragen, können Sie helfen, Probleme zu vermeiden, die damit verbunden sind, nur wenige übermäßig positive oder übermäßig negative Antworten zu haben.<sup>1</sup>

## Ausschließlich mit Systemdaten zu messen, ist schwierig

Ein weiterer Grund für Umfragen, ist, dass es nicht möglich ist, alles was passiert, mithilfe von Systemdaten zu erfassen, da Ihr System diese nur innerhalb seiner Grenzen erkennt. Im Gegensatz dazu können Menschen alles sehen, was in und um das System herum passiert und darüber berichten. Nehmen wir ein Beispiel, um das zu veranschaulichen.

Unsere Forschung hat ergeben, dass die Nutzung einer Versionsverwaltung eine wesentliche Kompetenz der Performance der Softwarebereitstellung ist. Wenn wir also wissen möchten, in welchem Umfang ein Team die Versionsverwaltung für alle Produktionsartefakte nutzt, können wir das Team fragen. Die Teammitglieder können uns darüber berichten, weil sie Einblick in die ganze Arbeit haben. Wenn wir das jedoch mithilfe eines Systems messen wollen, gibt es erhebliche Einschränkungen. Das System kann uns nur sagen, was es sieht – wie viele Dateien oder Repositories in die Versionsverwaltung kommen. Aber die bloße Zahl ist ohne einen Kontext bedeutungslos.

Idealerweise würden wir gerne den Prozentsatz der Dateien und Repos in der Versionsverwaltung kennen – aber den kann uns das System nicht sagen. Man müsste die enthaltenen Dateien und die *nicht* enthaltenen Dateien zählen. Das System weiß aber gar nicht, wie viele Dateien nicht in der Versionsverwaltung sind. Ein System sieht nur die in ihm enthaltenen Dinge – in diesem Fall ist die Verwendung einer Versionsverwaltung etwas, das mithilfe von Logdateien und Instrumentieren nicht genau gemessen werden kann.

<sup>1</sup> Ausgehend davon, dass Sie Daten im Hinblick auf Verbesserung erheben – ohne jedem zu sagen, dass sie positiv antworten müssen, *sonst*... Das entspräche dem Witz »Die Schläge gehen weiter, bis sich die Moral verbessert.« Sie würden die Daten bekommen, die Sie möchten – positive Antworten – aber sie wären bedeutungslos. Eine Möglichkeit, ehrliche Antworten zu unterstützen, ist eine anonyme Datenerhebung zu gewährleisten.