



Dave van Herpen

(dave.van.herpen@sogeti.com)
ist Management Consultant Enterprise Agility & DevOps bei Sogeti Niederlande. Er unterstützt Organisationen in der Rolle des Beraters, Coaches, Trainers und Change-Management-Beraters, um wendiger zu werden.



Robert den Broeder

(r.den.broeder@trigono.nl)
ist Eigentümer der Trigono BV und spezialisiert auf Service-Management und Organizational Behavior Management (OBM). Er unterstützt Organisationen beim Verwirklichen von Verhaltensänderung als Trainer, Coach und Berater.



Alexander van Ewijk

(Alexander.van.Ewijk@sogeti.de)
ist Managing Consultant der Business Unit Testing bei Sogeti Deutschland.

Verhaltensänderung als Kraftstoff für DevOps

Eine effektive Einführung von DevOps erfordert Methoden zur Unterstützung von Verhaltensänderung

Weltweit interessieren sich Organisationen sehr für DevOps. DevOps ist die Verschmelzung von IT-Entwicklung und IT-Betrieb. Eine Transformation zu dieser weitreichenden Stufe der multidisziplinären Zusammenarbeit soll Agilität, höhere Lieferfähigkeit und mehr Kunden- und Nutzerzufriedenheit ergeben. Verhaltensänderung in der eigenen Organisation ist dafür eindeutig DER Katalysator. Aber welche Methode ist dabei wirksam?

Dass viele Organisationen, auch in West-Europa, eine agile Vorgehensweise angenommen haben, ist keine Neuigkeit. Das bedeutet jedoch nicht, dass diese Organisationen die damit verbundenen Erwartungen auch erfüllen können und somit schnell Softwareänderungen produktiv setzen können. Gerade in der letzten Phase („the last mile“) lahmst der Produktentstehungsprozess, da ein reibungsloser Übergang von der IT-Entwicklung zum IT-Betrieb fehlt.

DevOps strebt die kontinuierliche Verbesserung der digitalen Wertschöpfungskette durch multidisziplinäre Zusammenarbeit in der gesamten Organisation auf den folgenden Ebenen an: Prozess, Technologie, aber vor allem auch Unternehmenskultur. Die DevOps-Bewegung hat ihren Ursprung vor allem in Start-ups und Internetfirmen. In den vergangenen Jahren hat die Implementierung von DevOps allerdings in schnellem Tempo auf andere Sektoren übergegriffen.

Die enorme Aufmerksamkeit für DevOps hat einen direkten Zusammenhang zum explosiven Wachstum der Lean Startup-Bewegung. Lean Startup sieht u. a. kurze

„Build-Measure-Learn“-Zyklen vor. Dabei ist der Effekt (d. h. Änderung des Kunden- und Nutzerverhaltens) jeder Softwareverbesserung einzeln quantitativ zu messen und jede Änderung dafür unter Umständen einzeln produktiv zu setzen.

Traditionelle Organisationen wollen, nein, müssen sich auch ändern. Um den Kampf gegen wendige Start-ups nicht zu verlieren, setzen Großunternehmen signifikante Organisations- und Kulturveränderungen ein. Modernisierung der Unternehmenskultur und der Personalentwicklung stößt auf viel Beifall, aber auch auf Widerstand und Kritik.

Vor allem, wenn die Menschen schon langjährig in einem Silo (Entwurf, Entwicklung, Test, Betrieb) tätig waren, sehen wir entsprechenden Widerstand. Das Ändern der eingeschliffenen Muster und der Kommunikationswege ist eine der wichtigsten Herausforderungen der Organisationen bei der erfolgreichen Einführung von DevOps.

Das hängt mit dem zusammen, was wir **Conway's Law** nennen: Organisationen, die Informationssysteme entwerfen, sind beschränkt auf das Produzieren von Ent-

würfen, die Kopien von den Kommunikationsstrukturen der eigenen Organisation sind. Mit anderen Worten, wenn wir wollen, dass Informationssysteme, die wir machen, wendig und flexibel sind und ein konsistentes Kundenerlebnis (englisch „customer journey“) ermöglichen, dann werden sich vorher die eigene Unternehmenskultur, Verhaltensweisen und Kommunikationswege in diese Richtung bewegen müssen!



Abb. 1: Die CALMS-Pfeiler

Das Akronym CALMS

DevOps ist viel mehr als nur kontinuierliche Integration und automatische Auslieferungswerkzeuge. Das geht aus den innerhalb der DevOps-Bewegung allseits bekannten CALMS-Pfeilern hervor. Die richtige Balance zwischen den Pfeilern ist für DevOps von entscheidender Bedeutung.

Die CALMS-Pfeiler umfassen:

- Kultur (Culture): Solange die Professionals und Manager nicht die erforderlichen Kompetenzen besitzen und nicht das erwünschte Verhalten zeigen, werden die Silos nie wirklich verschwinden. Somit ist eine integrale Veränderungsmethode erforderlich, die inkrementell (also schrittweise) an einer integrierten und ständig lernenden (d. h. fehlerverzeihenden) Organisation baut.
- Automatisierung (Automation): Das Ziel ist, so viele manuelle Aktivitäten wie möglich zu standardisieren und zu automatisieren.
- Lean (deutsch „schlank“): Die Arbeitsweise in DevOps-Organisationen ist stark mit Lean und mit der Engpasstheorie (englisch „theory of constraints“) verknüpft. Die Lean-Bewegung hat u. a. das Vermeiden von Verschwendung (englisch „waste“) als Ziel, d. h. von allen unnötigen, nicht wertbeitragenden Aktivitäten. Die Engpasstheorie beschäftigt sich mit der Optimierung von Gesamtsystemen, als Gegenentwurf zu der traditionellen Optimierung einzelner Silos.
- Metriken (Metrics): Um nachweislich besser zu werden, ist quantitativ auswertbare Information über Leistungen unentbehrlich, wie die Anzahl der Auslieferungen pro Zeiteinheit, oder der Zuwachs an Kunden- oder Nutzerinteraktionen.
- Teilen (Sharing): DevOps-Teams decken eine Verschiedenheit an Kompetenzen und Rollen ab, die vorzugsweise am selben Ort und mit einem adäquaten Prozess zum Wissensmanagement arbeitet.

Fokus auf Kultur & Metriken

In diesem DevOps-Artikel liegt der Fokus bewusst auf zwei entscheidenden Elementen aus den CALMS-Pfeilern: Kultur und Metriken.

Kultur

Kultur ist zu definieren als die Art, wie Personen in einer Einheit (Team, Grup-

pe, Abteilung, Organisation, Land usw.) miteinander umgehen. Geschriebene und ungeschriebene „Spielregeln“ – branchen-, unternehmens- oder standortspezifisch – werden in Wort und Tat umgesetzt und gleichzeitig von anderen erwartet: „So machen wir das hier!“

Weil die Kultur sich in Folge von DevOps ändert, kommt dem Management der Organisation eine ganz wichtige Rolle zu. Sie muss angeben, was DevOps in Bezug auf Kultur und Kulturveränderung bedeutet. Verhaltensregeln werden eingeführt, angepasst oder aufgehoben. Regeln werden von den Mitgliedern der Organisation ständig auf ihre Wirkung und ihre Konsequenzen getestet. Langsam, aber sicher entsteht eine Mischung aus funktionalen und weniger funktionalen Verhaltensmustern, gebündelt in der neuen Kultur der Organisation.

Metriken

Das Management ist gesamtverantwortlich für die Organisationsresultate. Metriken, die von (Projekt-)Managern verwendet werden, sollten eine ganz klare Beziehung zum Verhalten und damit zu der Kultur der Organisation haben. Die Beziehung steckt in einer der Gesetzmäßigkeiten des Leistungsmanagements (englisch „performance management“): „Man bekommt, was man verstärkt“. Das gilt für erwünschtes wie für unerwünschtes Verhalten. Bei unerwünschtem Verhalten spielen oft auch unbeabsichtigte Belohnungsmechanismen mit.

Da steckt die Krux für das erfolgreiche Implementieren von DevOps: Infolge der Transformation zu DevOps ändern sich die Spielregeln und wahrscheinlich auch die Metriken und Belohnungsmechanismen, die verwendet werden, um die Organisation zu steuern. Deshalb ist die Empfehlung, die Metriken sorgfältig zu wählen, die zu den erwünschten Leistungen und dem erwünschten Verhalten führen sollen.

Menschen wollen einen konkreten Beitrag leisten und dafür anerkannt werden. Von Natur aus suchen sie also die Verhaltensweisen, die zu Aufmerksamkeit und Anerkennung führen. In vielen Organisationen entsteht jedoch eine besondere „nicht-optimale“ Situation. Mitarbeiter, die tun, was man von ihnen erwartet, bekommen keine Aufmerksamkeit. Das Management richtet sich auf die Leute, die *nicht richtig arbeiten*. Der Grund ist einfach: *Dort ist nämlich Verbesserung möglich*.

Von Kultur zu Verhalten

Organisationen, die die DevOps-Ansätze

anwenden, bekommen es mit einer Kulturveränderung zu tun. Die Frage ist: Wie packt man das so an, dass eine „erwünschte Kultur“ entsteht, wobei DevOps sichtbar wird?

Eine klassische Methode für Kulturveränderung ist es, seitens des Managements vor allem *viele Signale* in Richtung der Organisation zu *senden*. Die Botschaft scheint klar zu sein: Wir müssen „den Kunden mehr in den Mittelpunkt rücken“, „proaktiv vorgehen“ und „transparent kommunizieren“. Diese Botschaft wird dann vielleicht zusätzlich noch durch das Nennen einiger Kernwerte wie „offen“, „ehrlich“ und „gemeinsam“ unterstützt.

Hierin versteckt sich eine wesentliche Gefahr: solche Botschaften und Kernwerte sind für viele Menschen zu generisch und haben kaum Effekt auf ihr Alltagsverhalten. Es wird nämlich nicht klar, welches Verhalten genau gezeigt oder geändert werden soll, um z. B. „den Kunden in den Mittelpunkt zu rücken“. Sollten Sie das bezweifeln: Machen Sie es mal vor! Wahrscheinlich müssen Sie eine ganze Menge verschiedener Handlungen verrichten. Daraus schlussfolgern Sie dann, dass Sie versuchen, den Kunden in den Mittelpunkt zu rücken.

Von Metriken zu Verbesserung

Nach unserer Vorstellung ist die Kultur ein Ergebnis aller Spielregeln und Verhaltensweisen in der Organisation. Direkt auf ein Endergebnis zu steuern, ist nicht praktisch. Es ist viel praktischer, zu untersuchen, welche Faktoren die Kultur beeinflussen, diese zu messen und zu verbessern, wo nötig und möglich. Diese Verbesserungen führen dann indirekt zu einer besseren Kultur. Klar spezifizierte Leistungen und relevante Metriken sind der Startpunkt der DevOps-Veränderung.

Die Kernfrage ist: Welche Variablen müssen wir messen, um DevOps-unterstützendes Verhalten (und damit eine DevOps-Kultur) aufzubauen?

Beispielspezifizierung DevOps-Verhalten

Im Folgenden werden einige Verhaltensweisen aufgezählt, die im Kontext von DevOps als erwünschtes Verhalten genannt werden:

- *Eigentümerschaft*: Der Produkteigner (englisch „product owner“) leitet die Produktvorführung und informiert alle Geschäftsinteressenten.

- **Multidisziplinär:** Teammitglieder übernehmen aktiv die Aufgaben eines Anderen, wenn sie sehen, dass ein Kollege zu beschäftigt oder nicht verfügbar ist.
- **Teilen:** Alle Teammitglieder teilen ihre Erfahrungen und ihr Wissen aktiv und persönlich mit ihren Teamkollegen und aktualisieren direkt einen Wiki.
- **Empathie:** Alle Teammitglieder (inklusive Produkteigner und Scrum Master) sprechen offen über ihre Sorgen und Bitten – wo nötig – um Hilfe.
- **Verbessern:** Auf jeden ernsthaften Zwischenfall folgt ein „blameless postmortem“, bei dem jeder

Beteiligte offen und ehrlich Ideen und Verbesserungsmöglichkeiten mitteilt.

Organizational Behavior Management

Seit einigen Jahren ist eine zunehmende Zahl von Organisationen mit „Organizational Behavior Management“, kurz OBM, beschäftigt. Dies ist eine konkrete, auf Leistungserhöhung orientierte Methode, deren Wurzeln im Behaviorismus (Skinner, Thorndike, Sulzer-Azaroff usw.) liegt.

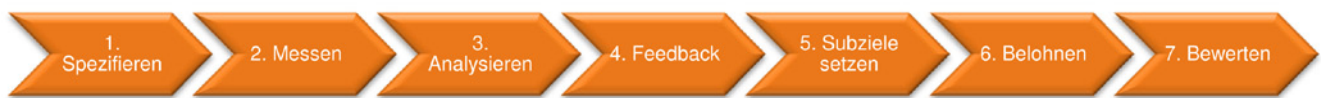
Die Stärke von OBM liegt in einem Protokoll, das erwiesenermaßen zu Leistungserhöhung und dauerhafter Verhaltensänderung führt. Der Kern des Protokolls besteht aus

Spezifizieren, Messen und danach Verstärken von Verhalten, indem man das erwünschte Verhalten belohnt.

Dieses Protokoll ist auch bei der Einführung von DevOps anwendbar. Denn die DevOps-Arbeitsweise funktioniert nur dann gut, wenn alle Beteiligten konsequent „das richtige Verhalten“ zeigen. Wenn alle mitmachen, entstehen neue Verhaltensmuster und eine neue Kultur. Die neuen Verhaltensmuster müssen sich einschleifen und OBM ist eine sehr effektive Methode, das zustande zu bringen.

Die 7 Schritte von OBM

Das Protokoll aus OBM, um zu Leistungserhöhung und Verhaltensänderung zu kommen, besteht aus den folgenden 7 Schritten:



Schritte:	Beispiel:
<p>1. Spezifizieren: Das Protokoll beginnt mit dem Spezifizieren der erwünschten Leistungen. Leistungen bestehen aus zwei Komponenten: einem konkreten Ergebnis und dem Verhalten, das zu diesem Ergebnis führt.</p>	<p>Spezifizieren einer Leistung:</p> <ul style="list-style-type: none"> ■ In Bezug auf das Ergebnis: Dadurch, dass Fehler möglichst an der Quelle behoben werden („no defects downstream“), ergeben sich niedrigere Fehlerfolgekosten in der Softwareentwicklung. ■ Dafür erforderliches Verhalten: Wachsamere, breiter Blick sowie direktes Melden von möglichen Qualitätsproblemen und gemeinsames Beheben dieser (nach Analogie der „Andon-Reißleine“).
<p>2. Messen: Wir messen, wie oft das erwünschte Verhalten schon gezeigt wird. Wenn dieses Verhalten zum ersten Mal, zum Beispiel während eines Zeitraums von einigen Wochen, gemessen wird, entsteht eine Baseline. Ein Fehler wäre, wenn man Messdaten dann als Argument verwendet, um „abzurechnen“.</p>	<ul style="list-style-type: none"> ■ Messung: Die Messung beinhaltet sowohl die Anzahl frühzeitig festgestellter Qualitätsprobleme als auch die geschätzten vermiedenen Fehlerfolgekosten in Euro = Basis für die Belohnung (in Schritt 6).
<p>3. Analysieren: Wir wollen das Verhalten, das zum aktuellen Leistungsniveau führt, analysieren. Wenn man Verhalten ändern will, dann muss man die Konsequenzen für dieses Verhalten ändern.</p>	<ul style="list-style-type: none"> ■ Transparenz über eine Ursachenanalyse der Qualitätsprobleme: z. B. mit der 5-Why-Methode aus Lean-Startup. Mögliche Fehlerursachen sind z. B. eine unzureichend durchgeführte Einarbeitung oder ungeprüfte bzw. fehlerhafte automatisierte Bereitstellung von Umgebungen.
<p>4. Feedback: Das Team braucht konkrete Leistungsrückmeldungen (englisch „performance feedback“). Leistungsrückmeldungen zeigen deutlich, wo das Team hinsichtlich des Zieles steht. Die beste Art und Weise, das zu tun, ist, den Fortschritt in einer Grafik wiederzugeben. Z. B. einfach mithilfe eines Flipcharts, das deutlich sichtbar in der Abteilung steht.</p>	<ul style="list-style-type: none"> ■ Feedback nach jedem Sprint: Trendlinien mit der Entwicklung der Anzahl der frühzeitig festgestellten Qualitätsprobleme je Ursache.
<p>5. Subziele setzen: Wir wollen Ziele vor allem konkret, erreichbar und herausfordernd machen. Das Endziel wird in SMART-Subziele (Akronym: „smart, measurable, achievable, realistic, timed“) verteilt. Es gilt die Regel: Erfolg erzielen, wie klein er auch sein mag, schmeckt nach mehr!</p>	<ul style="list-style-type: none"> ■ Ziel: sinkende Trendlinie innerhalb von 3 Monaten bei der Anzahl der frühzeitig festgestellten Qualitätsprobleme je Ursache. Die Ursachen sollten nach-und-nach strukturell abgestellt werden!

Schritte:	Beispiel:
<p>6. Belohnen: Jedes Subziel, das erreicht wird, führt zu einer bestimmten Form der Belohnung. Ein konsistenter Belohnungsmechanismus für umgesetzte Leistungen sorgt für eine Änderung im Denkmuster aller Beteiligten.</p> <p>7. Bewerten: Was ging gut, was könnte besser sein? In der Praxis entpuppt sich vor allem das Spezifizieren der erwünschten Leistungen als ausgesprochen herausfordernd.</p>	<p>■ Belohnung: z. B. ein Teamevent für das DevOps-Team mit einem Budget, das im Umfang von der Höhe der geschätzten Einsparung – z. B. 2% – abhängt.</p> <p>■ Nach einer gewissen Zeit sind alle Ursachen – hoffentlich – strukturell behoben (vgl. Abbildung 3) und es ist ein komplett neues Ziel erforderlich (im Beispiel: nach Sprint5).</p>

Abb. 2: Die 7 Schritte von OBM

OBM & DevOps

OBM ist ein sehr wertvolles Instrument, um Verhaltensänderung im Kontext von DevOps zu erreichen. DevOps ist vor allem eins: ein denkart- und kulturgesteuertes Organisationsthema. Natürlich sind technologische Weichenstellungen relevant, wie kontinuierliche Integration einschließlich automatisierter Tests sowie automatische Auslieferung. Prozesse sollten sich entwickeln in Richtung „Lean“ und „Agile“, aber die absolute Krux in Transformationen zu DevOps bleibt der Übergang zu einer neuen Art des Denkens UND des Verhaltens. Nur so bekommt die Transformation einen strukturellen und zukunftssicheren Charakter.

Insgesamt gesehen ist DevOps nicht das endgültige Endziel, sondern nichts mehr als ein logischer nächster Schritt für viele Organisationen, das ist sicher. ■

Fallstricke bei Verhaltensänderung

Bei Verhaltensänderung ist es gut, die folgenden drei Punkte zu beachten:

1. Eine Veränderungsbotschaft soll mit dem Ziel einer Verhaltensänderung gesendet werden. Verhalten ändert sich nur, wenn das Verhalten auch mit Konsequenzen verbunden ist.
2. Mit Konsequenzen arbeiten heißt nicht, dass man sich vor allem auf negative Konsequenzen wie Strafen konzentrieren soll, um Veränderungen zustande zu bringen. Belohnen von guten Leistungen funktioniert deutlich besser, ist schöner, nachhaltiger und weniger arbeitsintensiv.
3. Belohnen wird oft mit „mehr Geld“ übersetzt. Obwohl Geld für Viele schon wichtig ist, ist es kein nachhaltiger „Antrieb“. Viel wichtiger sind die sogenannten sozialen Belohnungen. Denken Sie dabei an Komplimente, Aufmerksamkeit und Anerkennung. Die einzige Bedingung für ein nachhaltiges Ergebnis ist, dass der Belohnung eine wirkliche Leistung gegenübersteht.

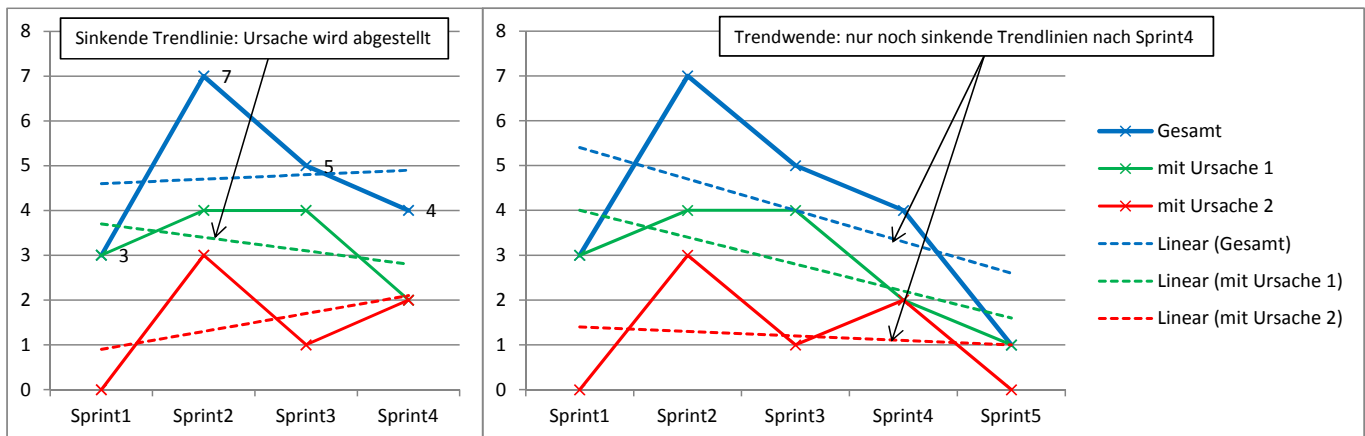


Abb. 3: Trendwende bei den Qualitätsproblemen: Änderung der Trendlinien (gestrichelte Linien) ab Sprint5