

## Vorwort zur 2. Auflage

Agile Softwareentwicklung ist zum Mainstream geworden. Zahlreiche Unternehmen, die ich in den vergangenen Jahren beraten habe, wagen sich inzwischen auch an sehr große Entwicklungsvorhaben mit agilen Methoden heran. Dabei hat sich herausgestellt, dass ganz neue Herausforderungen auftreten: Es geht längst nicht mehr nur um die Veränderung der Arbeitsweise einzelner Teams, sondern um den Wandel zum agilen Unternehmen, d. h. die Kultur und die Organisation, die agile Teams formt und ihnen die Möglichkeit eröffnet, effektiv zu arbeiten.

In der Diskussion um agile Softwareentwicklung bin ich immer wieder auf ähnliche Fragen gestoßen:

- Warum muss man überhaupt skalieren? Viel Komplexität kommt insbesondere erst durch eine schlecht strukturierte Codebasis, technische Schulden und organisatorische Altlasten zustande. Wichtiger wäre es, den Hinterhof aufzuräumen. Eine skalierte Implementierung muss sich danach fragen lassen, ob sie nur den Mangel verwalten will oder tatsächlich einen tiefgreifenden Change-Prozess anstößt.
- Agilität lebt von der Veränderung auch der agilen Prozesse selbst und versteht sich als Katalysator für kontinuierliche Veränderung und Verbesserung. Vordefinierte Frameworks bergen schon durch ihre Beschreibung die Gefahr, dass sie einmal entwickelt und ausgerollt und dann als Standard festgeschrieben werden. Das wäre genau das Gegenteil einer agilen Methode.
- Für wen sind welche Schritte sinnvoll? Dieselben Ideen und Praktiken können für die eine Firma einen Durchbruch in Richtung Agilität und Produktivität bedeuten – für andere wären sie ein Schritt zurück, entweder wirkungslos oder sogar eine Verschlechterung.
- Im Fall von SAFe kommt noch eine Frage hinzu: Warum braucht man überhaupt diese Rollen und Prozeduren, wenn die agile Kultur das wichtigste Erfolgsgeheimnis ist? Ist diese Komplexität nicht generell ein Irrweg?

Die neue Version 4.5 von SAFe ist gerade erschienen und das bietet die Gelegenheit, nicht nur über die Änderungen in SAFe selbst nachzudenken, sondern auch über die Entwicklungen in der Agilität im Allgemeinen zu reflektieren.

- Neue Mitbewerber sind erkennbar: Das LeSS-Framework ist fertig – jetzt auch in deutscher Sprache dokumentiert<sup>1</sup> – und hat sich als klarer Zweiter im Markt für agile Skalierungsframeworks etabliert. Daraus ist so etwas wie ein Zweikampf entstanden. Interessant ist dabei auch die klare Abgrenzung durch die erheblichen konzeptionellen Unterschiede: Während SAFe darauf setzt, schnell zu starten und für die meisten Vorschläge eine initiale Empfehlung zu geben, konzentriert sich LeSS viel stärker auf den erforderlichen Kulturwandel als notwendige Voraussetzung. SAFe vertraut darauf, dass die kulturellen Änderungen im laufenden Betrieb umgesetzt werden. Es zeigt sich bei unseren Kunden, dass die beiden Frameworks für sehr verschiedene Ausgangssituationen interessant sind.
- Einige der Erfahrungen beim Einsatz sind auch in die Änderungen von Version 4.5 eingeflossen: Man sieht ein klares Bemühen, mögliche Gründe für Implementierungsfehlschläge zu identifizieren und zu adressieren. Insbesondere die neue Einführungsstrategie und die wesentlich erweiterte und überarbeitete Lieferpipeline sind hier zu nennen.
- Neue Diskussionen erzeugen auch neue Missverständnisse: So wie man SAFe als Blaupause verwenden kann (es sei nochmals betont: Man wird ohne ein tiefes Verständnis der systemischen Zusammenhänge keine erfolgreiche Implementierung stemmen), so werden auch Erfahrungsberichte wie die der schwedischen Firma Spotify als Blaupause angewendet, wenn man über agile Transformation nach dem Spotify-Framework spricht: Spotify hat aber kein Framework veröffentlicht, sondern eine Momentaufnahme auf dem spezifischen Weg einer Firma. Der Beitrag zur Agilität ist nicht die Umbenennung von Teams zu Squads, sondern der Beitrag ist substanzieller: Wie kann man Vertrauen und Alignment skalieren und dafür nutzbar machen, die Software-Codebasis radikal zu entkoppeln.
- Das Verständnis für agile Prinzipien zu vertiefen wird in der nächsten Zeit eher noch stärker als bisher die Agenda bestimmen. In den Unternehmen ist ein umfassender und nachhaltiger Kultur- und Strategiewandel zu beobachten. Agilität ist nicht mehr auf die da unten im »Maschinenraum« begrenzt, es umfasst agile Geschäftsentwicklung und damit den Arbeits- und Aufgabenbereich des Topmanagements: Agilität steht dafür, auf allen Ebenen neue Fähigkeiten für die Firmen zu erschließen. Dazu brauchen wir nicht nur die Frameworks als Werkzeuge, sondern auch ein umfassendes Verständnis aller Beteiligten auf jeder Ebene.

---

1. [Larman & Vodde 2017]

Arbeiten wir gemeinsam daran weiter: Führungs- und Rollenbilder auf eine agile Grundlage zu stellen. Agilität mit Lean-Prinzipien zu kombinieren und zu einer langfristig tragfähigen Grundlage für unsere Wissensgesellschaft zu machen. Agile Transformationen durchzuführen und sie in eine kontinuierliche, agile Evolution zu verwandeln. Das Versprechen der Agilität als Win-win-Situation einzulösen: mehr Effektivität und eine angenehmere, befriedigendere Arbeitssituation zu schaffen.

### **Was will dieses Buch**

Dieses Buch beschreibt die Probleme und Möglichkeiten der Skalierung agiler Softwareprojekte und die Konsequenzen, die sich für die Kultur und Führung von Unternehmen ergeben, die agile Vorgehensweisen in großem Maßstab nutzen. Es richtet sich an Entscheider und Verantwortliche in solchen Unternehmen.

Das Buch konzentriert sich auf das Scaled Agile Framework (SAFe). Dieses Verfahren bietet einen integrierten Werkzeugkasten für die Skalierung sehr großer Projekte und ist damit ein geeigneter Startpunkt für die agile Transformation von reifen mittleren und großen Organisationen. Das Buch beschreibt die Prinzipien von SAFe und will zudem folgende Fragen beantworten:

- Welche Optionen habe ich bei der Gestaltung einer agilen oder Lean-Entwicklung. Welche Transformation ergibt sich für meine Organisation? Welche Ziele verfolge ich und welche Frameworks, Werkzeuge und Erfahrungen gibt es?
- Was sind die Ansprüche von SAFe und wie löst es diese ein? Wie agil ist SAFe? Sind die beschriebenen Strukturen und Prinzipien eine Umsetzung der ursprünglichen agilen Ideen?
- Warum sind die Strukturen, die SAFe zusätzlich zu agilen Methoden entwickelt, notwendig bzw. sinnvoll?
- Für wen ist SAFe nützlich und für wen nicht? Welche Erwartungen, Strukturen und Ziele passen zur Einführung von SAFe? Welche Fallen gibt es, welche praktischen Erfahrungen sollte man beachten?

### **Wie ist dieses Buch strukturiert**

Das Buch startet in Kapitel 1 mit der Frage, warum man überhaupt eine Entwicklung skalieren muss oder ob man diese Komplexität vermeiden kann.

Die Kapitel 2 bis 8 beschreiben SAFe. Kapitel 2 gibt einen Überblick, die folgenden Kapitel beschreiben die Ebenen von Team, Programm, Solution und Portfolio.

Kapitel 9 hat die Lieferpipeline zum Gegenstand, die in der Version 4.5 einen wesentlich breiteren Raum einnimmt als bei den bisherigen SAFe-Versionen.

Kapitel 10 beschreibt Metriken für die einzelnen Ebenen und diskutiert Probleme, die bei Metriken auftreten.

Kapitel 11 beschreibt das Rollenbild von Führungskräften, die eine agile und Lean-Transformation nachhaltig einführen wollen, und Kapitel 12 enthält Überlegungen zur Vorbereitung und Planung einer konkreten SAFe-Einführung.

Kapitel 13 liefert mehr Details zur Entwicklung einer Architektur und gibt einen Überblick über die softwaretechnischen Grundlagen, auf denen SAFe aufbaut.

Kapitel 14 beschreibt den Kontext von SAFe, stellt alternative Frameworks vor und diskutiert die Grenzen von SAFe.

### **Für den schnellen Überblick**

Wenn Sie sich schnell einen Überblick über SAFe verschaffen wollen, starten Sie mit Kapitel 2 » Überblick über SAFe«. Sie kennen danach die allgemeine Struktur und die Motivation von SAFe und können beurteilen, welche Teile für Ihre Organisation relevant sind.

Wenn Sie mit agilen und Lean-Konzepten noch nicht vertraut sind, helfen Ihnen die entsprechenden Abschnitte in Kapitel 14, einen Einblick in die Grundlagen zu bekommen.

### **SAFe einführen und umsetzen**

Wenn Sie sich überlegen, ob Sie SAFe einsetzen wollen, starten Sie mit Kapitel 12.

Zum Kennenlernen von SAFe im Detail sind die Kapitel 3 bis 9 gedacht.

Wenn Sie SAFe einführen wollen und an einer Übersicht aus Leitungssicht interessiert sind, sollten Sie die Kapitel 11 und 12 durchlesen, um zu erfahren, welche Herausforderungen und Planungsschritte auf Sie zukommen werden.

Wenn zu Ihren Aufgaben die Entwicklung der Softwarearchitektur gehört, erhalten Sie dazu mehr Informationen in Kapitel 13.

### **Hintergründe und Kontext**

Wenn Sie sich vor allem für die Hintergründe von SAFe interessieren, starten Sie mit dem Überblick in Kapitel 2 und springen direkt zur Betrachtung des Kontexts und der Alternativen von SAFe in Kapitel 14.

### **SAFe-Versionen**

Dieses Buch basiert auf der aktuellen Version 4.5 des Frameworks. Die erste Auflage war ursprünglich für die Version 3.0 konzipiert, parallel zum Schreiben des Buches entwickelte sich jedoch auch SAFe weiter und wurde dafür fit gemacht, auch sehr große Systeme zu managen, die nicht notwendigerweise reine Softwaresysteme sein müssen, sondern die oft Hardwarekomponenten oder weitere externe Systeme integrieren.

Version SAFe 4.0 wurde erweitert, um dieser Intention Rechnung zu tragen:

- Kanban erhält eine prominenter Rolle. Bisher wurde Kanban im Portfolioprozess genutzt, um den globalen WIP (Work-in-Progress) zu begrenzen. Jetzt werden Kanban-Systeme auf allen Ebenen integriert und bieten eine konsistente Sicht auf das Managen der aktiven Arbeitspakete.
- Kanban-Teams stehen gleichberechtigt neben Scrum-Teams. Sie nehmen allerdings genauso wie Scrum-Teams am Planungstakt der Programme teil und übernehmen die Rollen des Scrum Masters und Product Owners.
- Die Value-Stream-Ebene (in Version 4.5 in Solution-Ebene umbenannt) ist wesentlich leistungsfähiger geworden und als optionale Ebene zwischen Portfolio- und Programmebene eingebettet. Sie kann in sehr großen SAFe-Installationen Systeme, Suiten und Applikationen hoher Komplexität meistern.
- Die optionale Value-Stream-Ebene ist auch ein Einstieg in eine Modularisierung des Frameworks, um es leichter nutzbar zu machen und einen breiteren Anwendungsbereich zu erschließen. Diese Modularisierung wurde in Version 4.5 weitergetrieben.
- Einige Begriffe sind für eine bessere Verständlichkeit umbenannt worden. Der wichtigste hierbei ist der Begriff eines »Enabler«. Enablers sind technische Initiativen, um notwendige Fähigkeiten des Systems bereitzustellen. Enablers kommen in allen vier Ebenen vor und wurden vorher als Architektur-Epics, -Features und User Stories bezeichnet. Dies gilt in Version 4.5 auch für Capabilities.
- Die Verwendung des Begriffs »Architektur« hatte teilweise auch Vorteile – an manchen Stellen in diesem Buch werden daher beide Begrifflichkeiten verwendet.
- Die grundlegenden Prinzipien von SAFe sind überarbeitet und somit klarer geworden.

### **Die wichtigsten Änderungen für SAFe 4.5**

SAFe 4.5 ist im Juni 2017 freigegeben worden und bringt wiederum einiges Neues mit sich. Manche der Änderungen sind kosmetischer Art, aber es gibt auch wichtige Erweiterungen und Klarstellungen.

### **Neue Liefer- oder Delivery Pipeline**

Die wohl wichtigste Erweiterung ist die Liefer- oder Delivery Pipeline. Sie baut nach wie vor auf dem Mantra »Develop on Cadence, Release on Demand« auf, mit dem sichergestellt werden soll, dass das System jederzeit lieferfähig ist. Offenbar wurde das aber bisher nicht in jeder Implementierung erfolgreich umgesetzt, sodass jetzt sehr viel Sorgfalt in die detaillierte Beschreibung fließt, wie eine erfolgreiche Pipeline aussehen muss. Die Beschreibungselemente sind:

- **Continuous Exploration**, wo sichergestellt werden soll, dass regelmäßig neue Anforderungen des Marktes und der Kunden betrachtet werden und in Kooperation mit Kunden und anderen Stakeholdern neue Epics bzw. Features gefunden werden.
- **Continuous Deployment**, die Infrastruktur, die sicherstellt, dass neue Features auch zuverlässig, in hoher Qualität und in kleinen Inkrementen bis in eine Staging-Umgebung geliefert werden können.
- **Continuous Delivery** beschreibt, was nötig ist, um die Features in die Staging- und Produktionsumgebungen liefern zu können, damit man jederzeit lieferfähig ist. Dies wird jetzt deutlicher als bisher von den Releasezyklen getrennt, die von externen Marktanforderungen, von strategischen Entscheidungen, Compliance-Freigaben und von vielem mehr beeinflusst werden können.

### **Integration von Lean-Startup-Strategien**

Nicht weniger wichtig ist die Integration von Ideen aus dem Lean Startup, mit der das Problem angegangen wird, dass der Fluss von Epics durch das Gesamtsystem sehr linear aussah. Vor allem in der Explorationsphase wird deutlich, dass die Beschreibung jetzt wesentlich iterativer als bisher ist. SAFe bezieht sich auf Lean-Startup-Strategien und das Lean-UX-Modell. Das heißt, dass ein Epic keinen Business Case mehr bekommt, sondern eine Hypothese. Diese wird in einer ersten Version bis zur Marktreife entwickelt (als MMF, minimal marketable feature), dann wird entschieden, ob und wie oft weitere Versionen/Erweiterungen entwickelt werden.

Dadurch kann der endgültige Umfang deutlich von der ursprünglichen Planung abweichen und es wird ein expliziter Lernzyklus in die Entwicklung eines Epics eingebaut – manche Epics werden dadurch auch nie fertig, sondern werden kontinuierlich weiterentwickelt.

### **Scalable DevOps als Kultur-, Organisations- und technische Aufgabe**

DevOps wird primär als organisatorische und kulturelle Aufgabe beschrieben: Der wichtigste Punkt ist das Aufbrechen traditioneller Silos (Entwickler, Tester etc.) in cross-funktionale Teams, die gemeinsam eine häufige Lieferung neuer Funktionalität sicherstellen können. Erst auf dieser Grundlage können die Werkzeugketten (die dann aber auch nötig sind) Wirksamkeit entfalten.

### **Die Einführungsstrategie ist konkreter geworden**

Bei der Einführung werden wesentlich deutlicher die Erfolgsbedingungen und die Rollen bei einer SAFe-Implementierung genannt als bisher.

## Viele kleine Änderungen

- **Das Big Picture ist weniger komplex.** Das Big Picture wurde bereinigt und ist dadurch übersichtlicher geworden: Zu viele Details drohten die Übersichtlichkeit zu gefährden, vor allem, da im Laufe der Zeit immer mehr Informationen dazugekommen sind. Dazu gehört auch das Herauslösen der verschiedenen optionalen Rollen in die Randleiste, die Spanning Palette – dadurch wird klarer, dass z.B. das Systemteam je nach Bedarf auf verschiedenen Ebenen angesiedelt werden kann. Pro Konfiguration werden auch verschiedene Rollen angeboten, was Erfahrungen aus verschiedenen Anwendungen repräsentiert.
- **Neue Konfigurationen.** Die SAFe-Konfigurationen sind überarbeitet worden, um den Aufwand zur Anpassung zu reduzieren und sie zielgerichteter auf Anwendergruppen zuzuschneiden. Die vier neuen Konfigurationen sind Essential SAFe, Portfolio SAFe, Large Solution SAFe und Full SAFe und blenden nur die Aspekte des Big Picture und der unterstützenden Rollen ein, die für eine typische Implementierung des jeweiligen Zuschnitts benötigt werden.
- **Umbenennungen.** Verschiedene Umbenennungen sollen Quellen für Missverständnisse beseitigen. Die wichtigsten sind:
  - Die Value-Stream-Ebene heißt jetzt **Solution-Ebene**, insbesondere gibt es jetzt den Solution Train, der die Agile Release Trains zusammenfasst.
  - Das Budget-Verfahren heißt jetzt **Lean Budgeting**.
  - SAFe bekommt einen neuen Namen – es heißt jetzt »SAFe for Lean Enterprises« – damit soll auch ein erweiterter Anspruch für die Entwicklung komplexer »Systeme von Systemen«, etwa aus Hardware- und Softwarekomponenten, formuliert werden.

## Danksagung

Ich möchte mich an dieser Stelle herzlich bei folgenden Personen für ihre Unterstützung bei der Erstellung dieses Buches bedanken: Hedi Buchner, Stuart Fish, Conny Grieshaber, Pia Grund-Ludwig, Wolfgang Keller, Holger Neumann, Horst Peterjohann, Michael Stump, Eva Maria Wennmacher haben mir fantastischen Input zum Inhalt und zur Sprache gegeben.

Hedi Buchner, Stuart Fish und Wolfgang Keller haben meine frühen Versuche zurechtgestutzt und mir sehr dabei geholfen, den Fokus des Buches zu schärfen und die richtige Richtung zu finden.

Conny Grieshaber, Pia Grund-Ludwig, Horst Peterjohann und Eva Maria Wennmacher haben unglaublich viel Zeit investiert, um meine Formulierungen geradezubiegen. Conny hat zusätzlich noch viele Zeichnungen überarbeitet.

Michael Stump, Managing Director EMEA von Scaled Agile Inc. hat – last not least – an vielen Stellen geholfen, Zusammenhänge klarer darzustellen.

**Viel Erfolg beim Umsetzen**

Auf jeden Fall wünsche ich Ihnen viel Erfolg beim Umsetzen der Konzepte und insgesamt auf Ihrer Reise zu einer effektiven Nutzung von agilen und Lean-Konzepten. Möge das Buch Ihnen dabei behilflich sein.

Ich freue mich auch über Feedback zu meinem Buch.

*Christoph Mathis*  
München, im September 2017