

agile@scale: Do more with LeSS or be SAFe?

Ansätze zur Skalierung – ein Überblick

Patrick Daut¹

Abstract: Als Framework für agile Entwicklung hat vor allem Scrum in großer Verbreitung Anwendung gefunden. Gerade kleine und mittelständische Unternehmen in technologienahen Märkten profitieren von verkürzter Time-to-Market und hoher Flexibilität gegenüber Kundenanforderungen und der Variabilität des Marktes. Etablierte Konzerne versuchen, es ihnen gleichzutun, Flexibilität und Innovationskraft zu gewinnen und ihre Marktanteile zu sichern. Nun beruht Scrum aber auf der Arbeit eines kleinen Teams. Es liefert zunächst keine Lösungen für den Einsatz in einer großen Organisation mit einer Vielzahl von Teams. Aus diesem Bedarf heraus sind in den letzten Jahren verschiedene Ansätze zur Skalierung agile Vorgehens entstanden: Frameworks wie das Scaled Agile Framework (SAFe) oder Large-Scale Scum (LeSS), Ansätze wie der Scaling Cycle und populäre Umsetzungen in der Praxis wie bei Spotify. Worauf beruhen diese Ansätze, was ist ihnen gemein und was sind die grundlegenden Unterschiede? Woran kann ich mich bei der Auswahl eines Vorgehens orientieren? Im Rahmen dieses Beitrags wird ein Überblick über die wachsende Landschaft der Skalierungsansätze gegeben.

Keywords: Scrum, agile, lean, scaled agile, agile@scale, skalierung

1 Aus dem praktischen Bedarf heraus sind verschiedene agile Skalierungsframeworks entstanden

Agile Produktentwicklung verspricht Schnelligkeit, Flexibilität und Qualität. In großen Organisationen beschränkt sich die Implementierung agiler Methoden jedoch oftmals noch auf Softwareentwicklung und -test. Oft sind schon Produktmanagement und das obere Management kaum eingebunden; klassisches Anforderungsmanagement ist der Entwicklung vorgelagert und beschränkt die Möglichkeit, Nutzen aus agilen Vorgehensweisen in der Entwicklung zu ziehen. Analog bestehen Schranken auch in Richtung klassischer Unternehmensbereiche wie Finance, Budgeting Controlling etc. und auch interner wie externer Stakeholder. Des Weiteren ist die Koordination mehrerer paralleler Entwicklungsteams, die ab einer gewissen Organisationsgröße nötig wird – nur unzureichend gelöst.

In den vergangenen Jahren haben sich verschiedene methodische Ansätze und Frameworks entwickelt, die die Skalierung agilen Vorgehens in großen Organisationen zu lösen versuchen. Wie unterscheiden sich diese und welches ist das Richtige für mich?

¹ Cassini Consulting, Cassini Consulting Nord GmbH, Johannissbollwerk 16, 20459 Hamburg patrick.daut@cassini.de

2 Bei der Wahl eines Frameworks sind zwei Dimensionen ausschlaggebend

Die hier betrachteten Frameworks unterscheiden sich zum Teil erheblich in Ansatz, Ausgestaltung und im Grad der spezifischen Beschreibung. Bemerkenswert ist aber, dass sie alle auf ähnlichen Ideen und Prinzipien – letztlich aus dem Bereich des Lean Thinking – beruhen. Lean Thinking wird hier verstanden als konzeptionelle Idee hinter Lean Production und dem Toyota Production System, wie Womack und Jones² sie herausgearbeitet haben in einer generellen und auch auf Wissensarbeit anwendbaren Weise: Fokussierung auf Werthaltigkeit, Wertströme, Flussorientierung, Pull-Prinzip, Qualität – basierend auf einem hohen Stellenwert des Menschen.

Daneben weisen auf den ersten Blick ganz verschiedene Ansätze große Ähnlichkeiten in den durch sie definierten grundlegenden Organisationsstrukturen auf. Steht man vor der Aufgabe der Auswahl eines der Ansätze für den Einsatz in einer Organisation, sind die folgenden beiden Dimensionen ausschlaggebend:

- Größe der Entwicklungsorganisation
- Komplexität des Produktes und Marktes sowie die Größe der Gesamtorganisation

Die These ist, dass es sich bei diesen beiden Aspekten um die wesentlichen Einflussfaktoren auf den Bedarf an Steuerungs- und Kontrollmechanismen in Aufbau- und Ablauforganisation handelt.

3 Was bietet der Markt?

Bevor wir uns dem Vergleich der Frameworks und einer Hilfestellung für eine Auswahl widmen, soll im Folgenden ein Kurzüberblick über die betrachteten Ansätze gegeben werden:

Das Scaled Agile Framework (SAFe)³ ist ein komplexes Framework zur Einbettung agiler Entwicklung in einer großen Organisation mit Einbindung der klassischen Unternehmensbereiche. Es ist komplex in dem Sinne, dass es die folgenden Bestandteile definiert:

- Eine Aufbauorganisation in Form von Strukturen, Rollen und deren Abhängigkeiten
- Ein Prozess mit Artefakten, die zu definierten Zeitpunkten erstellt werden
- Methoden und Praktiken

² S. [WJ03] und [WJR07]. Vgl. auch das Konzept des *Lean Managements*.

³ S. [Le15] sowie [Le07], [Le10].

Kern ist eine über die Zeit stabile Organisationsstruktur, die die Arbeit mehrerer an einem (Teil-)Produkt beteiligten Teams zusammenhält und koordinative Rollen definiert. Die Teams selbst folgen einer erweiterten, angepassten Form von Scrum

Large-Scale Scrum (LeSS)⁴ definiert – vereinfacht ausgedrückt – zwei Aspekte:

- Eine einfache Struktur, die zum größten Teil auf Scrum Teams sowie einer Hierarchie aus Product Ownern besteht, wobei der Verantwortungsbereich eines Product Owners auf tieferen Hierarchieebenen kleiner wird
- Einen einfachen – ebenfalls kaum verändert auf Scrum basierenden Ablauf pro Team mit einzelnen Koordinationsmechanismen zwischen den Teams wie Scrum-of-Scrum oder teamübergreifenden Retrospektiven

Der **Agile Scaling Cycle**⁵ definiert im Gegensatz zu SAFe und LeSS weniger ein Framework für Aufbau- und Ablauforganisation als ein Vorgehen zur Entwicklung einer eben solchen Organisationsform für die jeweilige Umgebung. Er schlägt vor, zunächst mit der Minimierung von Abhängigkeiten zu beginnen, im zweiten Schritt mit den verbleibenden Abhängigkeiten in Form von Koordinationsmechanismen zwischen den Teams umzugehen um letztlich die Organisation weiterzuentwickeln, die die aufgetretenen Probleme dauerhaft löst. Anschließend beginnt der Zyklus von vorn. Hier kann – was den Ansatz angeht – ein Vergleich zu Kanban gezogen werden: Kanban beschreibt ebenfalls Schritte zur Definition eines situationsgerechten, individuellen Prozesses vorschlägt – nicht aber den Prozess selbst.

Bei der **Spotify Culture**⁶ handelt es sich um eine Beschreibung der Organisationsstruktur der Produktentwicklung bei Spotify mit einem Fokus auf der Aufbauorganisation. Es kann also weniger von einem Framework als von einem Praxisbeispiel oder einer Best Practice gesprochen werden: Ein gelebtes Beispiel einer leichtgewichtigen Organisationsform mit Fokus auf Strukturierung und Koordination von Teams. Auffälliges Merkmal der bei Spotify gelebten Struktur ist eine „Tribe“ genannte stabile Struktur, die mehrere agile Teams⁷ zusammenfasst und als koordinierende Organisationsform dient.

SAFe ist durch mehrere Bücher sowie eine inhaltsreiche Website beschrieben. Eine inzwischen starke Community wird aktiv betreut. Daneben existiert ein Trainings- und Zertifizierungsprogramm. Zusammen ergibt sich der Eindruck eines aktiv vermarkteten Produkts. LeSS basiert auf der Arbeit der beiden Autoren, zunächst in Form zweier Bücher. Auch für LeSS ist inzwischen eine inhaltgetriebene Website sowie ein Trainings- und Zertifizierungsprogramm verfügbar. Der Scaling Cycle hingegen wird in einzelnen Blogposts und Vorträgen beschrieben. Ähnliches gilt für die Spotify Culture von Kniberg – dessen Beträge allerdings sehr populär geworden sind.

⁴ S. [LV15] sowie [LV08], [LV10].

⁵ S. [Ro14a], [Ro14b], [RW14].

⁶ S. [Kn14a], [Kn14b].

⁷ In der von Kniberg vorgestellten Struktur werden die agilen Teams „Squads“ genannt.

Von SAFe und LeSS sind jeweils Sonderformen in der Entstehung oder veröffentlicht: LeSS Huge sowie SAFe for Systems Engineering. Beides fließt in die vorliegende Betrachtung nicht dediziert ein.

4 Wie unterscheiden sich die Frameworks?

Bereits die Kurzvorstellung hat gezeigt, dass sich die Ansätze teils stark unterscheiden. Für die Auswahl eines Ansatzes für den Einsatz in einer konkreten Organisation ist eine detailliertere Gegenüberstellung vonnöten. Tab. 1 zeigt die Ergebnisse hinsichtlich ausgewählter Aspekte.

Aspekt	SAFe	LeSS	Scaling Cycle	Spotify Culture
Ansatz	Komplexe Struktur und Vorgehensweise über 3 Ebenen	Erweiterung von Scrum um einzelne Elemente und Koordinationsmechanismen Sammlung einzelner Praktiken	Vorgehen zu Entwicklung einer individuellen Organisationsform	Vorstellung einer in der Praxis gelebten Organisationsstruktur und -kultur
Basis⁸	Feature-Teams, die im Wesentlichen nach Scrum arbeiten	Feature-Teams	Unabhängige Teams	Squads (Unabhängige, cross-funktionale Teams)
Hierarchie/ Linie⁹	Definierte Rollen mit jeweiligem Verantwortungsbereich	Hierarchie aus Entwicklung, Chief Product Owner, Area Product Owner und Product Owner	Für die jeweilige Organisation auszugestalten	Chapters

⁸ Welches sind die Organisationseinheiten an der Basis der Struktur?

⁹ Wird eine klassische Organisationsstruktur bzw. Organisationshierarchie durch das Framework mit abgebildet?

Aggregati- onsebene ¹⁰ (Teil- produkt,...)	<i>Agile Release Train</i> : stabile, langfristig angelegte Aggregation von Teams (100-120 Personen, Dunbar- Number)	Area; fachliche und/oder archi- tekturelle Zer- legung		Tribes (100-120 Perso- nen, Dunbar- Number)
Product Owner	Ein Product Owner pro Team zzgl. Koordinieren der Rollen auf Programm- und Portfolio- ebene	Team von Product Own- ern, ggf. Area Product Owner	Ein Product Owner; Teams besitzen selbst die Fähigkeit, grobe Features zu zerlegen	Ein Product Owner pro Squad
Quer- schnitts- funktionen	Definierte Rollen für Funktionen wie Archi- tektur, UX, etc, die koor- dinative und steuernde Funktion haben aber nicht die Ver- antwortung aus den Teams nehmen	Communities of Practice	Vorschlag: Communities of Practice	Guilds (und Chap- ters)
Komplexi- täts- bewältigung	Rollen und Koordinati- onsmechanis- men zum Umgang mit/ Management von Komple-	Reduzierung von Abhängig- keiten, um unabhängige Teams mit der Möglichkeit zur Selbstorga-	Reduzierung von Abhängigkeiten, um unabhängige Teams mit der Möglichkeit zur Selbstorganisati- on zu schaffen	Keine Beschrei- bung der Auflö- sung/des Manage- ments von Abhän- gigkeiten innerhalb des Tribes. Ziel ist Minimierung der

¹⁰ Welche Struktur oder welches Mechanismus sieht das Framework vor, um mehrere Teams hinsichtlich einen aggregierten gemeinsamen Ergebnisses zu koordinieren?

xität sowie auch zu des- sen Verringe- rung (Bsp.: Identifikation von Abhän- gigkeiten zwischen Teams im gemeinsamen Planning)	nisation zu schaffen; aber auch Einsatz einzelner Me- chanismen zum Umgang mit bestehenden Abhängigkei- ten	Abhängigkeiten zwischen Tribes. Mechanismen zu deren Identifikati- on und Ma- nagement
--	---	---

Tab. 1: Gegenüberstellung der Frameworks

Zum einen sind grundlegende Unterschiede festzustellen. Beispielsweise wird eine Aggregationsebene zur Koordination mehrerer Teams teils sehr verschieden gebildet. Auch wird die Einführung einer hierarchischen Organisationsstruktur – letztlich zum Zweck der Delegation - unterschiedlich gelöst. Andererseits finden sich Ideen wie Communities of Practice zur fachlichen Koordination über Teamgrenzen hinweg in fast allen der Lösungsansätze wieder. Ähnliches gilt für die Betonung der Bedeutung von unabhängigen Feature-Teams für die Skalierung. Hintergrund hier ist die Minimierung von Abhängigkeiten und somit eines minimalen Bedarfs an Koordinationsmechanismen über Teams hinweg.

Bemerkenswert ist, dass ein für seine komplexe Spezifikation teils in der Literatur kritisiertes Framework wie SAFe gerade hinsichtlich der definierten Organisationsstrukturen Ähnlichkeit zu denen des Spotify Modells aufweist.

5 Welcher Ansatz eignet sich in welcher Umgebung?

Kommen wir nach diesem ersten Vergleich der Ansätze zur eingangs gestellten Fragestellung zurück: Was kann mir bei der Auswahl eines Frameworks als Orientierung dienen?

Der Vergleich der Ansätze zeigt Unterschiede vor allem in Komplexität der Frameworks, Detailgrad der definierten Strukturen, Umgang mit Komplexität sowie teamübergreifende Koordinationsmechanismen. Im Rahmen dieses Beitrags wird vorgeschlagen, dass eine Wahl von den folgenden Dimensionen und Kriterien abhängen sollte:

Produkt, Markt und Umgebung

- Welche Kontrollinstanzen habe ich? Bewege ich mich in einem regulierten Bereich oder liegen besondere regulatorische Anforderungen vor?
- Ist mein Geschäft eher produkt- oder projektorientiert?

- Wie komplex sind Produkte, Projekte oder Aufgaben? Wie groß sind Unsicherheiten?
- Wie groß ist die Gesamtunternehmung?

Organisation

- Wie groß ist die Entwicklungsorganisation?
- Wie ist das Unternehmen aufgebaut? Besteht Bedarf der Einbettung einer agilen Entwicklung in das Gesamtunternehmen und in klassische Unternehmensfunktionen? Welche Möglichkeit der Anpassung besteht dort?
- In welchem Maß besteht bereits Erfahrung mit agilen Methoden? Ist dies für das ganze Unternehmen neu? Gab es fehlgeschlagene Versuche? Oder ist die Entwicklung agil erfolgreich und es fehlt noch an den Schnittstellen?

Abb. 1 zeigt eine mögliche Einordnung der untersuchten Frameworks hinsichtlich der beiden beschriebenen Dimensionen.

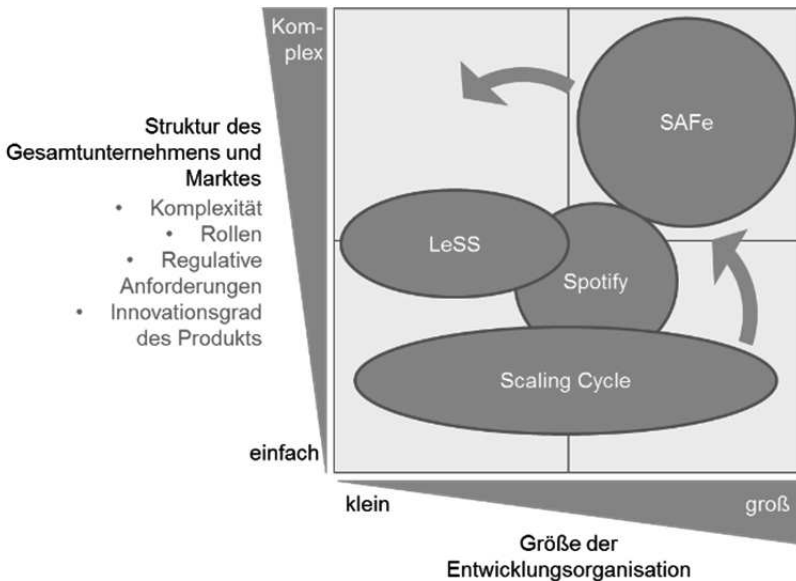


Abb. 1: Systematik der Eignung der untersuchten Frameworks

Ein Ansatz wie der Scaling Cycle kann unabhängig von der Größe der Entwicklungsorganisation eingesetzt werden – er selbst definiert noch keine feste Struktur und keinen Prozess. Mit zunehmender Komplexität der Gesamtunternehmung, des Produktes und des Marktes bringen diejenigen Ansätze Vorteile mit sich, die bereits konkrete Lösungen liefern. Diese Vorteile kommen bei wenig Erfahrung mit agilem Vorgehen in der Organisation noch mehr zum Tragen. Die durch eine solche Komplexität induzierte Vielzahl

an Rollen im Unternehmen, an internen und externen Stakeholdern sowie etwaigen Reglementierungen erfordern konkrete Lösungen. Während der am Beispiel von Spotify beschriebene Ansatz eine für mittelgroße Unternehmen durchaus geeignete Aufbauorganisation definiert, bringt er nur wenig prozessuale Unterstützung mit sich. LeSS wiederum definiert ein einfaches, kaum von Scrum abweichendes Vorgehen sowie eine skalierbare Struktur. Der Fokus auf eine Hierarchie aus Chief Product Owner, Area Product Owner sowie Product Owner auf Teamebene wird jedoch ab einer gewissen Größenordnung auf Grenzen stoßen: Der Kommunikations- und Koordinationsbedarf wird in dieser Struktur stark ansteigen. LeSS betont, dass der Schnitt der Anforderungen wie auch der Teams entlang kundenrelevanter Features und nicht technische Module oder Schichten entscheidend ist. Die dadurch angestrebte Unabhängigkeit der Anforderungen untereinander und Teams voneinander ist ab einer gewissen Größe aber nicht mehr ausreichend, um den Koordinationsbedarf zu begrenzen. Product Owner auf den unteren Hierarchieebenen tragen kaum noch Produktverantwortung. SAFe hingegen ist in seiner definierten Form für kleinere Unternehmen eher zu mächtig, bringt aber eine Vielzahl konkreterer Lösungen auf prozessualer Ebene sowie eine skalierungsfähige Aufbauorganisation mit, die die Koordination einer hohen Zahl an Teams erlaubt sowie verschiedene Stakeholder und klassische Unternehmensbereiche einzubinden vermag.

Während der Scaling Cycle durch seine Anwendung durchaus erlaubt, funktionierende Lösungen auch für größere Unternehmen zu entwickeln, so zeigt die Projekterfahrung, dass auf Basis von SAFe einfachere Strukturen auch für mittelgroße Organisationen entwickelt werden können.

Literaturverzeichnis

- [Kn14a] Kniberg, H.: Spotify Engineering Culture, Blogeintrag auf blog.crisp.se, <http://blog.crisp.se/2014/03/27/henrikkniberg/spotify-engineering-culture-part-1>, 5.8.2015.
- [Kn14b] Kniberg, H.: Spotify Engineering Culture, Blogeintrag auf labs.spotify.com, <https://labs.spotify.com/2014/03/27/spotify-engineering-culture-part-1>, 5.8.2015.
- [LV15] Large-Scale Scrum Website, <https://less.works/de/less/framework/index.html>, 5.8.2015
- [Le07] Leffingwell, D.: Scaling Software Agility: Best Practices for Large Enterprises. Addison Wesley, 2007.
- [Le10] Leffingwell, D.: Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise. Addison Wesley, 2010.
- [LV07] Larman, C.; Vodde, B.: Practices for Scaling Lean and Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum. Addison Wesley, 2010.
- [LV10] Larman, C.; Vodde, B.: Scaling Lean and Agile Development: Thinking and Organizational Tools for Large-Scale Scrum: Successful Large, Multisite and Offshore Products with Large-scale Scrum. Addison Wesley, 2010.

- [Le15] Scaled Agile Framework Website, <http://www.scaledagileframework.com>, 5.8.2015
- [Ro14a] Roock, S.: Folien eines Vortrags auf dem Agile Leadership Day 2014, Slideshare, <http://de.slideshare.net/roock/agile-scaling-cycle-lightning-talk-at-agile-leadership-day-2014-zurich>, 5.8.2015.
- [Ro14b] Roock, S.: Agile Scaling Cycle: from SCaLeD principles to practices, Blogeintrag, <https://stefanroock.wordpress.com/2014/07/13/agile-scaling-cycle-from-scaled-principles-to-practices>, 5.8.2015.
- [RW14] Roock, S.; Wolf, H.: Agile Skalierung - Prinzipien statt Blueprint, Folien eines Vortrags auf der OOP 2014, Slideshare, 5.8.2015.
- [WJ03] Womack, J.; Jones, D.: Lean Thinking: Banish Waste And Create Wealth In Your Corporation. Simon & Schuster UK, New Edition, 2003.
- [WJR07] Womack, J.; Jones, D., Roos, D.: The Machine That Changed the World: The Story of Lean Production - Toyota's Secret Weapon in the Global Car Wars That Is Now Revolutionizing World Industry. Free Press, Reprint, 2007.