

The logo for MHP, featuring a stylized 'M' composed of three vertical bars of varying heights, followed by the letters 'MHP' in a bold, sans-serif font.

A PORSCHE COMPANY

The background is a teal-tinted image of a person's hands holding a glowing lightbulb. The lightbulb is surrounded by a network of white lines and dots, resembling a digital or neural network. The overall aesthetic is futuristic and technological.

SCALED AGILE @ SYSTEMS ENGINEERING

Wie zwei scheinbar unterschiedliche Lösungsansätze Synergien in der R&D erzeugen und die Effizienz steigern.

Inhalt

01 Einleitung	6
02 Was verstehen wir unter Agilität?	8
03 Was verstehen wir unter Systems Engineering?	10
04 Die Synthese, oder: Das Beste zweier Welten	14
05 Die Agile Systems Engineering Transformation	24
06 Ausblick	28

FLEXIBILITÄT

REAKTION

EFFEKTIVITÄT

INNOVATION

ADAPTIVITÄT

01

Einleitung

Agilität und Systems Engineering sind fester Bestandteil einer reaktionsschnellen und flexiblen Produktentwicklung.

Volatilität, Unsicherheit, Komplexität und Ambiguität prägten die Geschäftswelt der vergangenen Jahrzehnte – kurz zusammengefasst in dem Denkmodell des Akronyms „VUCA“. Ergänzt wird dieses, angetrieben durch den Klimawandel und die Pandemie, seit kurzem um das weitere Akronym „BANI“. „BANI“ steht für porös, ängstlich, nicht-linear und unbegreiflich. Es bildet einen neuen Rahmen, in dem Volatilität und Komplexität nicht mehr ausreichen, um die gegenwärtige Veränderung von schwer, hin zu völlig unvorhersagbaren Ereignissen zu beschreiben¹. Gemeinsamen stellen die Einflussfaktoren beider Denkmodelle Unternehmen vor die zentrale Aufgabe, sich an neue Rahmenbedingungen anzupassen. Dies hat vor allem Auswirkungen auf den R&D-Bereich. Er steht vor der Herausforderung, immer effektiver und effizienter arbeiten zu müssen, während gleichzeitig reaktionsschnellere und flexible Produktentwicklungen notwendig sind.

Als Beispiel hierfür ist der steigende Komplexitätsgrad in der Entwicklung von mechatronischen Systemen zu nennen – nicht nur die Anzahl an Funktionsanforderungen steigt, auch der Anspruch an Interdisziplinarität. Treiber dieses Trends sind in erster Linie die digitalen Technologien und der damit einhergehende Innovationsdruck. Der Markt fordert vermehrt Vernetzung sowie datengetriebene Geschäftsmodelle. Für die Systementwicklung bedeutet dies: Klassische Strukturen und schnittstellenoptimierte Prozesse müssen grundlegend hin zu integrativen Ansätzen überdacht werden. Unter anderem spiegelt sich dies im Wandel von einer hardware- hin zu einer stärker funktionsorientierten Entwicklung wider. Bei Unternehmen, die diesen Schritt in den vergangenen Jahren vernachlässigt haben, sind heute regelmäßig Verzögerungen bei Produktauslieferungen, hohe Nachbesserungsaufwände bei zertifizierungsrelevanten Inhalten oder gar das vollständige Ausbleiben von geplanten Entwicklungsumfängen zu beobachten.

Auf der Suche nach einer Lösung für diese und weitere Herausforderungen wie kurze Technologiezyklen, regulatorische Anforderungen und die Mitarbeitermotivation werden in der R&D derzeit zwei Modelle favorisiert – wahlweise Systems Engineering (SE) oder das skalierbare agile Vorgehensmodell SAFe® (Scaled Agile Framework). Beide Modelle besitzen eigene Stärken, offenbaren aber auch bisher unberücksichtigte Bereiche, in denen weitere Potenziale erschlossen werden können.

Ziel des vorliegenden Whitepapers ist es, die gewinnbringenden Aspekte beider Modelle in einer gemeinsamen Synthese aufzuzeigen und somit das Konzept eines agilen (adaptiven) sowie integrativen Systems Engineerings vorzustellen. Die zentrale Anwendung dafür liegt darin, technologisch komplexe Entwicklungsvorhaben zu beherrschen, die durch ein dynamisches oder unsicheres Umfeld reaktionsschnelle und flexible Anpassungsmöglichkeiten erfordern. Auf den Mehrwert dieses Ansatzes verweist ebenfalls das Standardwerk der INCOSE (International Council of Systems Engineering), die zur Untersuchung des agilen Systems Engineerings eine eigene Arbeitsgruppe gegründet haben².

1) Grabmeier, S. (2020): BANI vs. VUCA, Quelle: (<https://stephangrabmeier.de/bani-vs-vuca/>), Stand: 22.07.2020

2) INCOSE (2015): Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, Fourth Edition, John Wiley & Sons

Die Transformation verläuft in individuellen Geschwindigkeiten, erlebt aber in traditionellen Organisationen ähnliche Hürden – diese gilt es zu meistern.

In der Transformationspraxis haben viele Unternehmen mit der Implementierung von mindestens einem der beiden Modelle begonnen. Allerdings stecken sie häufig in einzelnen Stufen fest. Eine nicht repräsentative Umfrage von MHP bei 20 Klienten verschiedener Branchen aus dem Juli 2020 zeigt fünf wie-

derkehrende Fragestellungen, mit denen sich Unternehmen bei der Implementierung und Verankerung von Agilität und/oder Systems Engineering konfrontiert sehen:

- Wofür sollten wir SE/Agile einsetzen und welchen Nutzen haben wir davon?
- Theorie und Rollen sind bekannt – wie funktioniert die praktische Implementierung?
- Wie lassen sich unsere Prozesse und Tools auf SE oder Agile Methoden abstimmen?
- Synchronisation mit Organisation – wie lösen wir die Koppelung an weitere Bereiche?
- Agilität funktioniert bisher nur in der Softwareentwicklung – wie können wir das Modell skalieren?

Die konsequente Implementierung des Systemgedankens in der Entwicklung und Organisation oder die Skalierung von Teamprinzipien auf Programm- oder Portfolio-Ebene scheitern oftmals an den klassischen Prozessen und Strukturen einer funktionalen Hierarchie (Aufbauorganisation). Auch genügt es in der Praxis nicht, die zentralen Rollen wie System Architect, Function Owner oder Agile Coach nur zu beschreiben, ohne sie integrativ in der Organisation zu befähigen und substanzial in die Abläufe einzubinden. Daher wird dieser Artikel die genannten Fragestellungen aufgreifen und am Beispiel des kombinierten Ansatzes erklären.

Ebenso wird im Folgenden der potenzielle wirtschaftliche Nutzen beleuchtet.

Mit einer ausgewogenen Kombination agiler Vorgehensmodelle und Systems Engineering ist ein Einsparpotenzial von 10 bis 50 Prozent möglich.

Beim Fokus auf mögliche Einsparpotenziale werden drei Messgrößen in Betracht gezogen: Qualitätskosten (Einsparpotenzial durch nicht erforderliche Nachbesserungen), Time-to-Market/Cost-of-Delay (Umsatz- und Wettbewerbsvorteile durch frü-

heren Markteintritt der Systeme) sowie Innovationsumsatz (Umsatz durch Minimum Viable Products oder einzigartige Features mit neuartigem Kundennutzen). Selbstverständlich sind die Potenziale von der jeweiligen Branche sowie der Systemkomplexität und dem Marktumfeld abhängig und können an dieser Stelle nur als Basisindikation dienen. Weitere Potenziale wie eine höhere Mitarbeiterzufriedenheit (geringere Krankheitsraten, höhere Arbeitgeberattraktivität sowie niedrigere Fluktuation) wurden aufgrund uneinheitlicher Messmethoden der Unternehmen nicht miteinbezogen, sind als positive Nebeneffekte jedoch gegeben.

Für die Implementierung des Ansatzes werden folgende drei Themenschwerpunkte unterschieden, die im weiteren Verlauf des White Papers näher beschrieben werden.

1) Einsatz von agilen und Systems Engineering-Methoden auf Teamebene

Effizienzsteigerung ~10 Prozent
[Quelle: MHP Project Database]

2) Transformation und Skalierung auf Organisationsebene

Produktivitätssteigerung ~20-50 Prozent
[Quelle: www.scaledagileframework.com]³

3) Einführung einer agilen (adaptiven) Systemarchitektur

Effizienzsteigerung bei komplexen Systemen ~10-40 Prozent
[Quelle: MHP Project Database]

Bevor diese Aspekte detailliert betrachtet werden, soll ein einheitliches Verständnis zu Agilität und Systems Engineering geschaffen werden.

3) Scaled Agile – SAFe 5.0, Quelle: (<https://www.scaledagileframework.com/safe-for-lean-enterprises/>), Stand 22.07.2020

02

Was verstehen wir unter Agilität?



Dynamische Märkte erfordern dynamische Unternehmen.

Im Kontext des Projekt- und Programmmanagements wird unter dem Begriff der „Agilität“ eine Vielzahl von Vorgehensweisen, Methoden und Werkzeugen subsumiert, die teils mit unterschiedlichem Erfolg angewandt werden. Bei den skalierten Vorgehensweisen wird das skalierte agile Framework SAFe® mit 54 % im Vergleich zu LESS oder unternehmensspezifischen Eigenlösungen am häufigsten eingesetzt⁴.

Im Wesentlichen beschreibt Agilität jedoch nicht den Einsatz einer spezifischen Projektmanagement-Methode, sondern die grundlegende Fähigkeit, effektiv und kompetent auf ein operatives Umfeld mit zunehmender Unsicherheit und Unvorhersehbarkeit zu reagieren. Um dies zu ermöglichen sind zwei Grundprinzipien von zentraler Bedeutung: **Flexibilität** und **Adaptivität**.

Übertragen auf ein Projekt bedeutet dies, dass Veränderungen nicht als Störung, sondern als Teil des Problemlösungsprozesses empfunden werden. Um jedoch willkürliche Reaktionen zu vermeiden und einen planbaren Projekterfolg sicherstellen zu

können, wurden agile Vorgehensmodelle entwickelt, die mithilfe kurzer Iterationszyklen, wiederkehrender Routinen und in regelmäßigen Intervallen die Arbeitsergebnisse und Arbeitsweisen reflektieren. In einer Wettbewerbsumgebung ist Agilität somit das Nebenprodukt einer natürlichen Selektion, die zum einen auf Variation und zum anderen auf dem „Überleben des Anpassungsfähigsten“ bzw. der effektivsten/effizientesten Lösung beruht. So ermöglichen es kurze Iterationszyklen, Teillösungen zu erproben, inkrementell weiterzuentwickeln oder als ungeeignet zu verwerfen. Das Ergebnis ist ein Prozess des kontinuierlichen Anpassens, Verbesserns und Lernens.

Aus den beiden Kernprinzipien der Flexibilität und Adaptivität hat Hüsselmann⁵ die zentralen Paradigmen, Prinzipien und Ziele der Agilität abgeleitet (Abbildung 1). Sie lassen sich ebenfalls in den klassischen Projektmanagement-Standards wiederfinden, werden in der Praxis jedoch oftmals wenig angewendet.

4) Komus, A. et al. (2020): Study Status Quo (Scaled) Agile 2019/2020, Hochschule Koblenz 02/2020
 5) Hüsselmann, C.; Maibach, M. (2020): Agilisierung des Projektportfoliomanagements. Praktiken und Rollen für traditionelle Unternehmen, WI-Report Nr. 012, Gießen/Friedberg: THM, ISSN 2568-0803

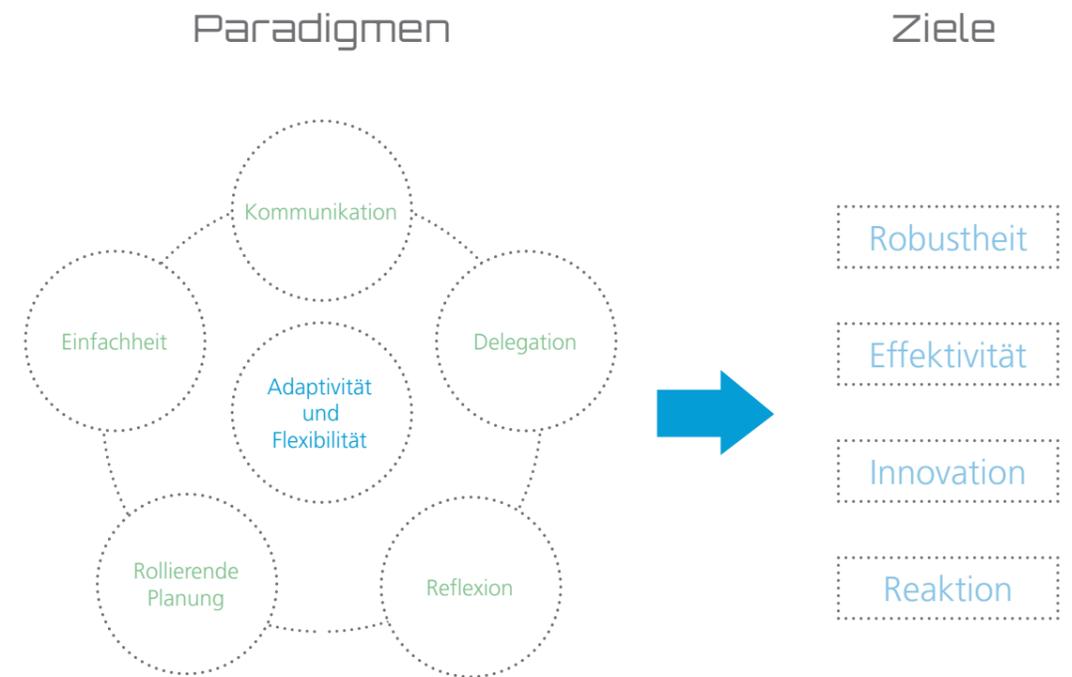


Abbildung 1: Paradigmen und Ziele der Agilität nach Hüsselmann

„Agile Vorgehensmodelle unterstützen die Anwendung agiler Werte und Prinzipien. Sie fördern die Kommunikation und Interaktion, übertragen die Verantwortung auf die betroffenen Mitarbeiter und legen Wert auf funktionierende Projektgegenstände, eine gute Zusammenarbeit mit dem Kunden sowie die Veränderungsbereitschaft im Sinne einer höheren Kundenzufriedenheit oder eines besseren Produkts.“

GPM Deutsche Gesellschaft für Projektmanagement e.V. (Hg.) (2019): Kompetenzbasiertes Projektmanagement (PM4). Handbuch für Praxis und Weiterbildung im Projektmanagement. Nürnberg: GPM Dt. Ges. für Projektmanagement, S. 163

Paradigmen im Wandel der Disruption und Ambiguität der Zeit.

Basierend auf MHP-Erfahrungen ist es unser Verständnis, dass viele Paradigmen, die in den vergangenen Jahrzehnten Struktur und Stabilität geschaffen haben, sich in einer von VUCA & BANI geprägten Welt auf dem Prüfstand befinden. Sie müssen stattdessen durch adaptive und flexible Grundhaltungen sowie gleichzeitig strukturierte Vorgehensmodelle ersetzt werden. Diese Entwicklung bezieht sich auf folgende Aspekte:

Auf Grundlage dieser Prinzipien wurden in den vergangenen Jahren zahlreiche agile Vorgehensweisen und Ansätze entwickelt. Mit SAFe® hat sich ein auf der Systemsicht basierendes, organisatorische Bedürfnisse befriedigendes und dennoch adaptives Modell etabliert, welches aufgrund seiner weiten Verbreitung hier Anwendung findet.

Von	Nach
Das Management weiß es am besten	Das Team weiß es am besten
Langfristige Strategien sind unersetzbar	Strategien enthalten adaptive Komponenten
Strategien sind für den Kunden	Strategien entstehen mit dem Kunden
Der Plan muss eingehalten werden	Adaption an Veränderungen darf den Plan verändern
Keine Experimente oder Risiken	Veränderung und Fehlerkultur schaffen Innovation



03

Was verstehen wir unter Systems Engineering?



Systems Engineering unterstützt Unternehmen nicht nur dabei, die steigende Komplexität in der Produktentwicklung zu beherrschen – sie fördert auch disziplinübergreifendes Systemdenken.

Systems Engineering ist eine erprobte Herangehensweise, um komplexe Systeme erfolgreich zu entwickeln – so einfach lässt sich Systems Engineering im Kern definieren und verstehen. Vier Bausteine sind fundamental für das Systems Engineering. Sie sind nicht immer scharf voneinander zu trennen, stellen jedoch ein vollständiges Bild über die Reichhaltigkeit dieses Themas dar:

1) Systems Engineering als standardisierte Disziplin in einem Prozessmodell

Auf den ersten Blick erscheint dieser Ansatz naheliegend – er ergibt sich größtenteils aus dem „gesunden Menschenverstand“. Der Blickwinkel beschreibt Systems Engineering als intuitive und praxisnahe Disziplin. Motiviert durch die Kritikalität der frühen Raumfahrt sowie Systemen, deren Komplexität stärker anstieg, als es deren Entwicklungsmethoden beherrschen konnten, hat sich Systems Engineering etabliert. Besonders wenn die Komplexität durch die Kombination von neuen Technologien, domänenübergreifenden Funktionalitäten und softwarebasierter, automatisierter Steuerung wächst, wird Systems Engineering genutzt. Dieser formale Aspekt – Systems Management – ist als Prozessreferenzmodell für die Entwicklung komplexer Systeme international standardisiert und in der ISO15288:2015 sowie weiteren begleitenden ISO-Standards definiert.

2) Systems Engineering als Methodenbaukasten zur praxistauglichen Umsetzung der Disziplin

Der Systems-Engineering-Ansatz beinhaltet einen umfassenden Baukasten an Methoden sowie Technologien zu dessen Realisierung. Beide haben sich über die Jahre kontinuierlich weiterentwickelt. Unterschieden werden methodische Ansätze, die aus dem Systems Engineering heraus entwickelt worden sind, und solche, die im Bereich der Qualitätssicherung ihren Ursprung haben. Eng verbunden mit dem Systems Engineering sind beispielsweise Modellierungssprachen wie SysML oder OPM entstanden. Darüber hinaus lassen sich alle Methoden aus „Design for Six Sigma“ gewinnbringend mit einer systemischen Herangehensweise kombinieren.

3) Systems Engineering als fundierte Wissenschaft mit allgemeingültigen Kriterien und Prinzipien

Ergänzend beschreibt Systems Engineering auch einen wissenschaftlichen Ansatz, der als Engineering-Ausprägung der Systemtheorie zu verstehen ist. Die Identifikation von übergreifend gültigen Prinzipien hilft dabei, die Herangehensweise des Systems Engineering wissenschaftlich fundiert zu beschreiben. Dazu zählt beispielsweise die Identifikation von allgemein gültigen Systemeigenschaften, die den disziplinübergreifenden Austausch unterstützen.

4) Systems Engineering als Geisteshaltung von Systemdenkern

Zunächst wird häufig ein Zugang zum Systems Engineering über den Begriff des Systems Thinking gewählt. Dieser Begriff erscheint jedoch losgelöst von den anderen hier betrachteten Bausteinen oft als schwer greifbar. Dadurch entsteht ein Eindruck, der weit entfernt von einer Anwendung im realen Projektgeschehen ist. Ersetzen durch: Der Systemdenker verfolgt die Geisteshaltung, Fragestellungen im Ganzen zu betrachten und komplexe Ursachen- und Wirkungsbeziehungen zu identifizieren. Auf Systeme übertragen, lässt sich die Relevanz dieser Geisteshaltung für die Umsetzung der anderen Bausteine nachvollziehen und wertschöpfend einbringen. Hierfür ist die Rolle des Systemingenieurs ein wichtiger Bestandteil.

Das Systems Engineering-V-Modell wird auch abseits der Softwareentwicklung eingesetzt und bietet die Voraussetzung für eine agile Produktentwicklung.

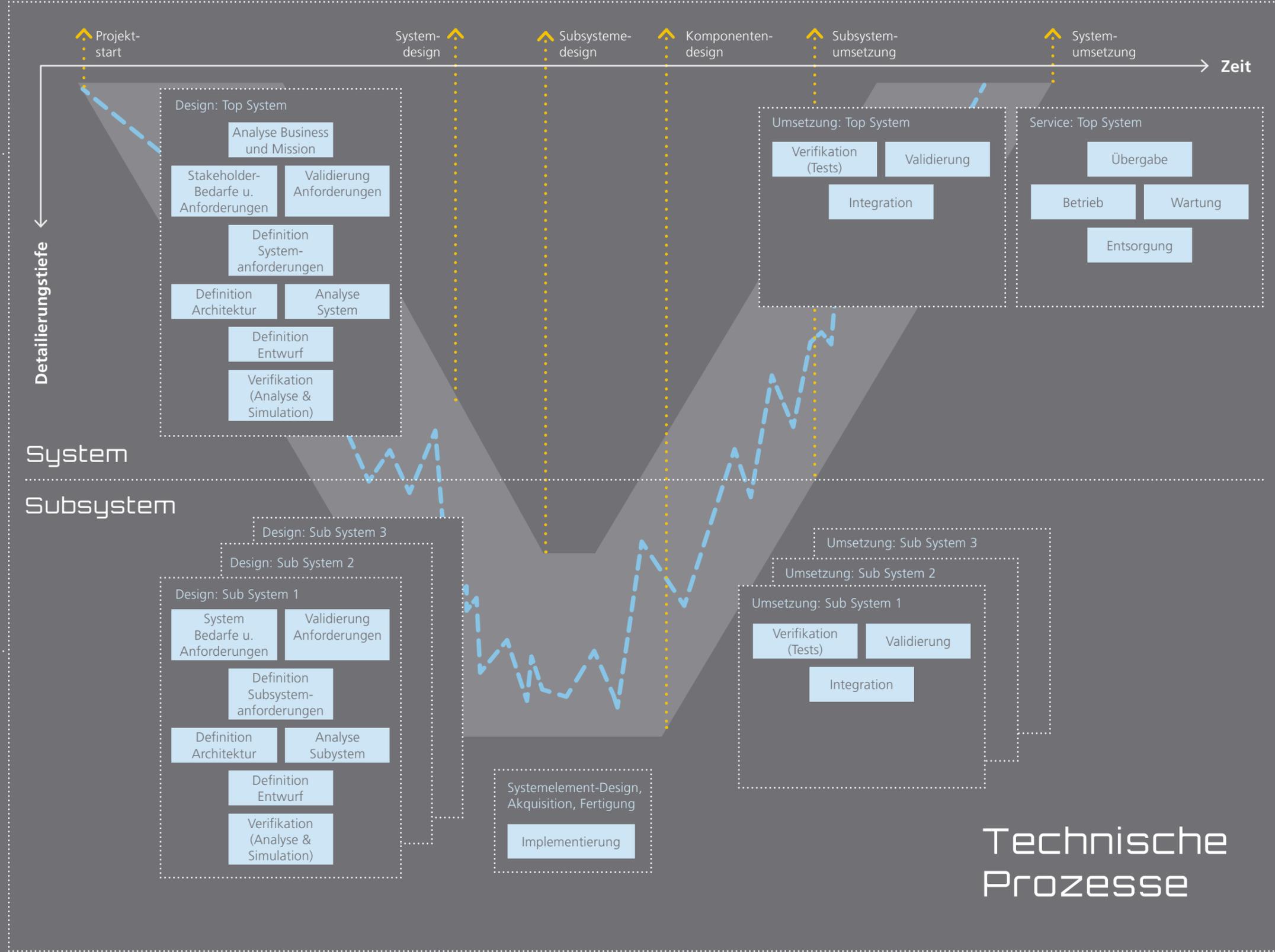
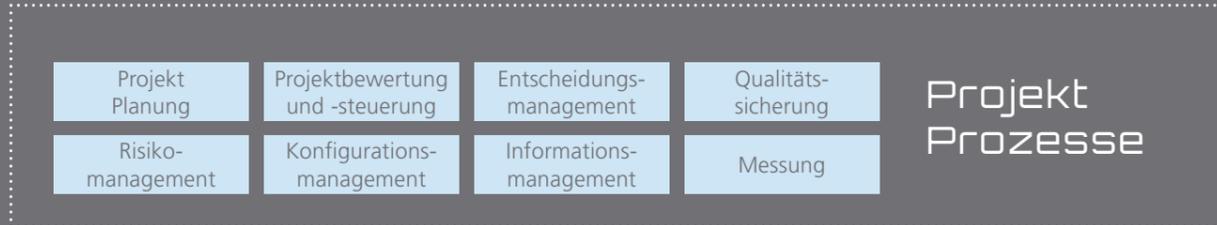
Darüber hinaus wird das V-Modell (Abbildung 2) oft stellvertretend für Systems Engineering verwendet – es repräsentiert vielerorts eine dokumentlastige, sequenzielle Arbeitsweise. Wenn das V-Modell jedoch konsequent in den Dimensionen Zeit und Detaillierungstiefe eingebettet wird und definierende Eigenschaften wie die Breite des On-Core-Bereichs sowie die Bedeutung der Kanten ernst genommen werden, ergibt sich ein anderer Blickwinkel. Das V-Modell lässt Raum für Iterationen, unterstützt eine moderne sichtenorientierte Ergebnisdokumentation und bildet dabei den notwendigen Rahmen, um systemkritische Aspekte termintreu abzubilden. Die Anwendung der Kernprozesse lässt sich nahezu beliebig iterativ und rekursiv auf den gesamten Entwicklungszyklus übertragen, was den agilen Gedanken unterstützt.

Systems Engineering gelingt nur, wenn ein einheitliches Verständnis von Systemen und Modellen erlangt wird, sowie mit einem kontinuierlichen, vom Management getragenen Kulturwandel.

Die Vermittlung der vier definierten Systems-Engineering-Bausteine muss rollenspezifisch angepasst werden. Eine ausgewogene Qualifizierung der Elemente ist für systemrelevante Rollen eine essenzielle Voraussetzung, um die systemischen Konzepte zu realisieren. Zusätzlich muss die Organisation entlang einer ebenenorientierten Systemstruktur ausgerichtet und das Arbeiten mit stakeholderrelevanten, aussagekräftigen Sichten etabliert werden. Modelle sind die Entwicklungssprache im Systems Engineering.

Diese vier Bausteine spiegeln sich in unserem Verständnis des Systems Engineering wider – und genau in diesem Verständnis kombinieren wir es mit den Aspekten der Agilität, um die bestmögliche Herangehensweise für die Herausforderungen von heute und morgen zu schaffen.

Rekursive und iterative Anwendung der Prozesse der ISO 15288:2015 am zeitlich orientierten V-Modell



Technische Prozesse

Abbildung 2 Eigene Darstellung in Anlehnung an: Dove R.; Schindel B. (2016): Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern, 26th Annual INCOSE International Symposium, Edinburgh

04

Die Synthese Oder: Das Beste zweier Welten



Die Vorteile einer gemeinsamen Implementierung beseitigen wesentliche Schwächen und fördern die Stärken der einzelnen Modelle.

In der folgenden Tabelle sind die Vorteile einer Synthese aus Systems Engineering und des agilen Vorgehensmodells SAFe® abgebildet. Dies unterstreicht die ergänzende Kompatibilität der beiden Modelle.

Als Frage bleibt jedoch, wodurch sich der Einsatz der kombinierten Vorgehensweisen indizieren lässt?

Eine Entscheidungshilfe für den wirtschaftlichen Einsatz agiler Prinzipien im Systems Engineering bietet das CURVE-Modell.

INCOSE⁶ liefert mit dem **CURVE**-Modell eine einfache und universell anwendbare Indikation für eine sinnvolle Implementierung beider Modelle im Programm- oder Produktumfeld. Die

Buchstaben des Akronyms stehen hierbei für Capriciousness (Unvorhersehbarkeit), Uncertainty (Unsicherheit), Risk (Risiko), Variation (Variabilität) und Evolution. Werden für ein Portfolio, Programm oder Produkt jeweils eindeutige Sinnfälligkeiten in diesen Bereichen identifiziert, empfiehlt INCOSE die Anwendung von Agile Systems Engineering.

Zur näheren Beschreibung dieses Ansatzes unterscheiden wir drei Dimensionen: Methoden und Prozesse, Organisationsentwicklung sowie agile Systemarchitektur. Auf dies wird im Folgenden detailliert eingegangen.

6) Dove, R.; Schindel W.; Hartney, R. (2017): Case Study: Agile Hardware/Firmware/Software Product Line Engineering at Rockwell Collins, 11th Annual IEEE International Systems Conference Montréal

	Systems Engineering	SAFe®	SAFe® und SE
Entwicklungsprozesse	++	0	++
Steuerungsprozesse (Ablauforganisation)	0	++	++
Aufbauorganisation	0	+	+
Rollenbeschreibungen	++	++	++
Compliance-Sicherstellung	++	0	++
Adaptivität	+	++	++
Flexibilität	0	++	++
Komplexitätsmanagement	++	0	++
Systemischer Ansatz	++	+	++
Tool-Unterstützung	+	+	+

Tabelle 1: Gegenüberstellung Systems Engineering und SAFe®, basierend auf den Standards der jeweiligen Modelle (0 = keine Angabe, + = gut geeignet, ++ = sehr gut geeignet)

Exemplarisch: Prozesse, Liefergegenstände und Informationsflüsse im skalierten Agile Systems Engineering

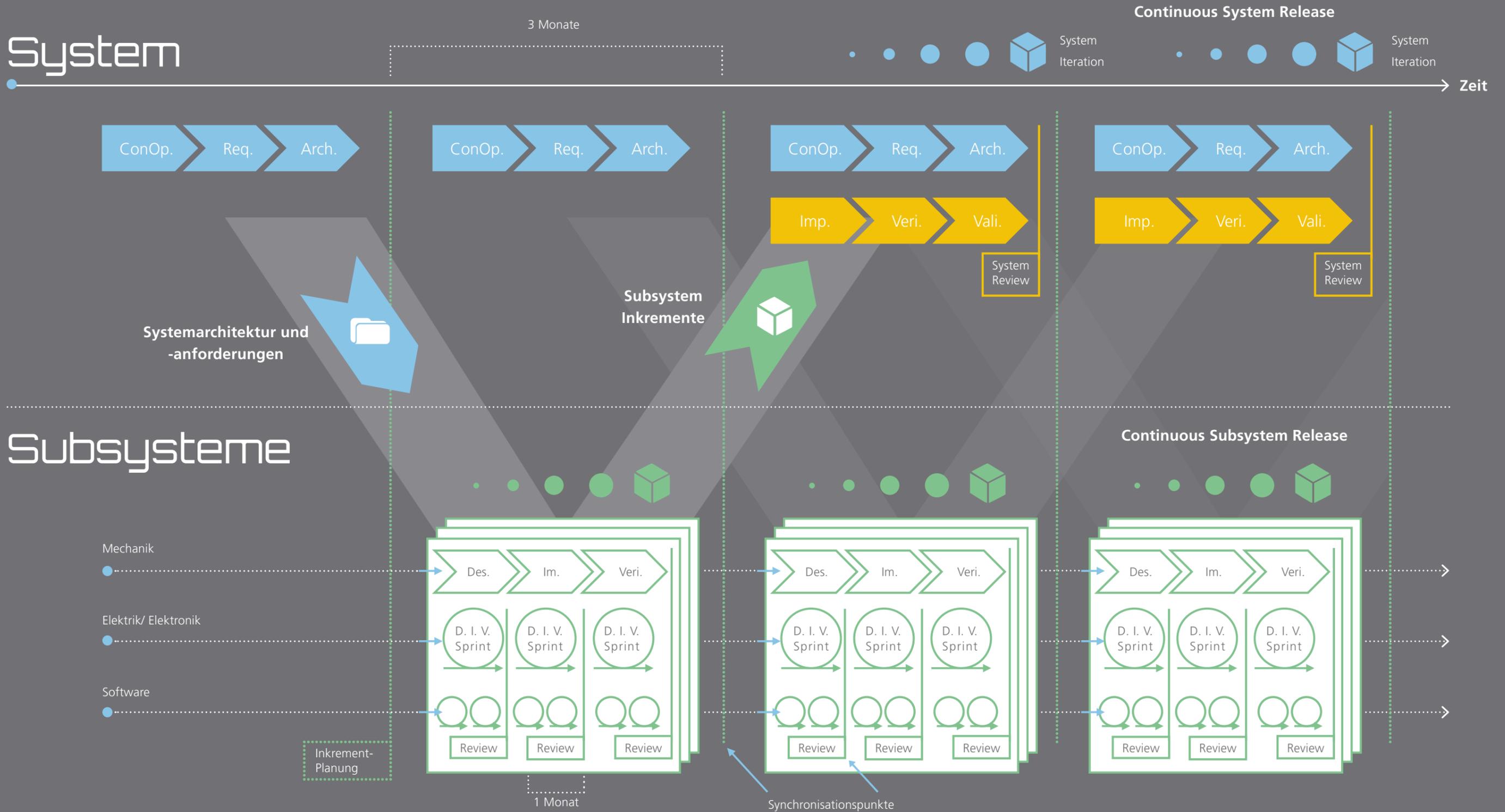


Abbildung 3

4.1 Agile Systems-Engineering-Methoden und Prozesse

Die Etablierung von agilen und System-Engineering-Methoden sowie der prozessualen Verankerung entlang des Wertstroms bilden das Fundament einer erfolgreichen Transformation.

Die Synthese aus Agilität und Systems Engineering bezeichnen wir als „Scaled Agile @ Systems Engineering“ (SA@SE) – es vereint die Stärken beider Teildisziplinen. Kurz zusammengefasst liefert das Systems Engineering hierfür die Prozesse sowie die Struktur, nach denen die interdisziplinäre und ganzheitliche Entwicklung von Systemen erfolgt, während agile Prinzipien zur Gestaltung der Zusammenarbeit auf Team- sowie Organisationsebene genutzt werden. Für den R&D-Bereich bieten beide Themengebiete umfassendes Prozess- und Methodenwissen, wobei der Fokus des Systems Engineering – einfach ausgedrückt – auf dem „was“ und der der Agilität auf dem „wie“ entwickelt wird liegt.

Abbildung 3 zeigt eine exemplarische Darstellung des kombinierten Arbeitsmodells nach SA@SE. Dieses verbindet jeweils das prozessuale und methodische Wissen aus dem SE in Form des V-Modells mit den adaptiert-agilen Ansätzen nach SCRUM und SAFe®.

Im Detail wird das Entwicklungsvorhaben – dem SE-Systemgedanken folgend – sowohl in eine übergeordnete System- als auch eine untergeordnete Subsystem-Ebene unterteilt. Auf der System-Ebene überwiegen weitestgehend die Prozesse und Methoden des SE im Sinne der Prozessschritte nach ISO15288:2015. Auf diese Weise werden Stakeholder-Bedürfnisse in Systemanforderungen und Architekturen überführt, die als Grundlage für die Aufteilung in Subsysteme und deren spezifische Anforderungen dienen. Zusammen mit der späteren Implementierung der Subsystem-Inkrementen zu einem vollständigen System sowie der Verifizierung und Validierung wird ein effektiver Rahmen für die ganzheitliche Systementwicklung gebildet.

Auf Subsystem-Ebene überwiegen dagegen agile Methoden und Routinen, die es je nach Komplexität der Subsysteme notwendig machen, auf mehrere SE-Entwicklungsteams zurückzugreifen. Hierbei ist es sinnvoll, fachspezifische Teams zu bilden, um die jeweiligen Bedürfnisse der Subsystem-Elemente in Form von Entwicklungs-, Fertigungs- und Testzeit in unterschiedlichen Zyklenlängen berücksichtigen zu können. Zum Gelingen dieser Aufteilung ist die Synchronisation der SE-Teams mithilfe übergreifender agiler Zeremonien wie dem Review und der Inkrement-Planung von besonderer Bedeutung. Hierdurch bleiben auch außerhalb der eigenen Fachdomäne die Transparenz, Kommunikation und das gemeinsame Commitment erhalten. Die übrigen agilen Zeremonien, wie das Daily oder Backlog Refinement, bestehen weiterhin auf Teamebene. Hierfür besitzt jedes SE-Team einen eigenständigen Backlog, der die entsprechenden Systemanforderungen auf Subsystem-Ebene überführt. Mithilfe dieses Arbeitsmodells gilt es, spätestens alle drei

Monate die funktionierenden Subsystem-Inkrementen für eine System-Iteration zur Verfügung zu stellen. An dieser Vorgabe orientiert sich der Rhythmus des gesamten Modells – sowohl auf System- als auch Subsystem-Ebene. Dadurch wird eine parallelisierte und wiederkehrende Routine geschaffen, welche die kontinuierliche Freigabe von Subsystem-Inkrementen und darauffolgenden Systemiterationen als Minimal Viable Systems zum Ziel hat. Auf dieser Basis werden in regelmäßigen Zeitabständen nicht nur die Systemanforderungen an die Bedürfnisse von Kunden und Stakeholdern angepasst, sondern auch die Erkenntnisse vorangegangener Entwicklungszyklen für eine höhere Systemreife genutzt. Ein Erfolgsfaktor für die Umsetzung dieses Arbeitsmodells stellt die Vorbereitung adäquater skalierungsfähiger Organisationsstrukturen dar.

4.2 SAFe® – eine Systems-Engineering-Organisation

Die Skalierung in SAFe® orientiert sich am Menschen und bildet optimale Kommunikationsstrukturen nach.

Die Korrelation zwischen der Organisation und Systementwicklung beschreibt Conway⁷ wie folgt: **“Organisationen die Systeme entwerfen [...] sind gezwungen, Entwürfe zu produzieren, die Kopien der Kommunikationsstrukturen dieser Organisation sind.”**

Für eine erfolgreiche Entwicklung von komplexen Systemen ist somit die zugrundeliegende Organisation der Kommunikationsflüsse eine zentrale Basis. Hierzu wird SAFe® als Grundlage für die organisatorische Ausprägung von SA@SE herangezogen (Abbildung 4). In jeder Ebene der skalierten Aufbauorganisation sind verschiedene Sichten beziehungsweise Granularitätsstufen des identischen System Backlogs (Liste anstehender Systemfunktionalitäten) wie auch die jeweiligen Rollen und das umfassende Arbeitsmodell der Skalierung definiert. Auf der untersten Ebene arbeiten unterschiedliche SE-Teams abhängig von ihrer Domäne im Arbeitsmodus nach SA@SE an den Aufgaben des Subsystem- oder Funktionsbacklogs. Per agiler Definition des Begriffs „Team“ sind hier alle, zur eigenverantwortlichen Erreichung einer Umsetzung der Subsystem- oder Funktionsentwicklung involvierten Rollen, integriert. Die Synchronisation der SE-Teams erfolgt über die Programm- oder Systemebene, welche in regelmäßigen Zyklen von drei Monaten die Fortschritte bewertet und in einem Inkrement-Planung weiter, für die folgenden drei Monate, planen. Erfahrungsgemäß ist es mit diesem Ansatz möglich, Teamgrößen von bis zu 125 Personen im Rahmen einer agilen Systementwicklung effektiv zu organisieren. In umfangreicheren Entwicklungsprojekten mit mehreren Systemen und mehr als 150 Beteiligten kann es darüber hinaus erforderlich sein, eine dritte Stufe, die Solution Ebene, nach dem identischen logischen Organisationsschema hinzuzufügen (Abbildung 4, rechte Seite).

7) Conway M. E. (1968): How Do Committees Invent?, Quelle: (<http://www.melconway.com/>), Stand: 22.07.2020



Vereinfachte Darstellung der Rollen und Organisation nach SAFe® in einer Adaption auf Systems Engineering



Abbildung 4

Systemingenieur und Systemarchitekt sind elementare Rollen beider Modelle und werden durch die Synthese weiter gestärkt.

Trotz teilweise unterschiedlicher Benennungen sind wesentliche Rollen des Systems Engineering mit denen des SAFe®-Modells inhaltlich deckungsgleich (Abbildung 4, linke Seite). Auffällig in SAFe® ist, dass auf allen Stufen der Skalierung, der Dreiklang der Rollen aus Marktsicht (Product Management), Techniksicht (System Engineer) und Prozesssicht (Release Train Engineer) den zentralen Erfolgsfaktor in der wirtschaftlichen Entwicklung bildet.

Die übergreifende technische Verantwortung kommt dem Systemingenieur zu. Auf Solution-Ebene stimmt er sich mit dem Solution Engineer zur Planung der nächsten Zyklen anhand der jeweiligen Sichten auf die gemeinsamen Backlogs ab. Auf der Systemebene bricht er die technischen Systemanforderungen in verständliche Arbeitspakete für die Subsystem-Ebene herunter. Nach erfolgter Bearbeitung durch die SE-Teams nimmt er die Ergebnisse in Form ihrer Subsystem Inkrementen wiederum ab. Ergänzend stehen dem erfolgreichen Entwicklungsablauf die RTEs (Release Train Engineers) sowie der STE (Solution Train Engineer) zur Verfügung.

Drei fundamentale Aufgaben des unternehmensweit vernetzten RTEs sind an dieser Stelle besonders hervorzuheben:

- 1) Kontinuierlichen, kurzzyklischen Verbesserungsprozess fördern und fördern
- 2) Adressierung und Nachverfolgung von Eskalationsthemen
- 3) Moderation der übergreifenden Planungs- und Abstimmungstermine

Der Bedarf an dieser Rolle wurde in SAFe® zuerst identifiziert und beschrieben. Sie existiert im Systems Engineering derzeit nicht. Für die weiterführende Beschreibung der spezifischen Aufgaben und Verantwortungen der einzelnen Rollen in SAFe® sei an dieser Stelle auf das aktuelle Framework SAFe 5.0⁸ verwiesen.

8) Scaled Agile – SAFe 5.0, Quelle: (<https://www.scaledagileframework.com/>), Stand: 22.07.2020

4.3 Die agile Systemarchitektur

Die konsequente Umsetzung einer agilen Systemarchitektur ermöglicht maximale Flexibilität bei der Umsetzung neuer Markt- und Kundenanforderungen an die System- und Funktionsgestaltung.

Um die Vorteile des agilen Arbeitsmodells im Rahmen des Systems Engineerings nutzen zu können, ist es ebenso notwendig, die Grundlagen zur Entwicklung von agilen Systemen und deren Funktionalitäten zu legen. Andernfalls sind die Entwicklungsteams in ihrem Vorgehen zwar anpassbar an sich verändernde Umgebungszustände und Anforderungen – nicht jedoch die Systeme selbst, die sie entwickeln. Die Agilität eines Systems wird als die Fähigkeit definiert, in einem unsicheren und sich unvorhersehbar entwickelnden Umfeld zu bestehen, um sowohl auf Chancen als auch auf Bedrohungen wirksam reagieren zu können. Agile Systeme werden daher auf Veränderungen ausgelegt und sind optimiert für eine dynamische Betriebsumgebung. Sie zeichnen sich durch folgende Eigenschaften aus:⁹

- Erweiterbarkeit um neue funktionale Fähigkeiten
- Fähigkeit zur Umstrukturierung der internen Beziehungen zwischen den Teilsystemen
- Skalierbarkeit nach oben und unten zur wirtschaftlichen Bereitstellung der Funktionalität
- Umgestaltungsfähigkeit zum Wiedergewinn der Kompatibilität oder Synergie bei Formänderungen der Umgebung

Diese Arten von Veränderungen sind struktureller Natur und erfordern eine agile Architektur, die dem strukturellen Wandel Rechnung trägt. Hierbei umfasst der Architekturbegriff nach ISO 42010 die grundlegenden Konzepte oder Eigenschaften eines Systems – realisiert durch die Elemente, Beziehungen und Prinzipien seines Entwurfs sowie seiner Weiterentwicklung.

Unter der agilen Systemarchitektur wird so ein sofort einsetzbarer modularer Baukasten verstanden, der über „Drag and Drop“ – sowie „Plug and Play“-Beziehungen verfügt. Verdeutlichen lässt sich dies am Beispiel der Spielzeug-Konstruktionssets von LEGO. Deren Baukästen bestehen aus verschiedenen Arten von Komponenten, die in definierter Weise miteinander verbunden werden und interagieren können. Auf diese Weise werden der Aufbau und die Veränderung von unterschiedlichsten Systemen mitsamt ihrer Funktionalität ermöglicht. Obwohl sich der Baukasten für einige Konstruktionstypen besser eignet als für andere, unterliegt er einem einheitlichen architektonischen Muster. Ein Muster dieser Art bildet die Grundlage für den Aufbau agiler Systeme und wird daher als agile Systemarchitektur bezeichnet.¹⁰

Zur Beschreibung einer agilen Architektur werden folgende drei Gestaltungselemente herangezogen (Abbildung 5).¹²

1) Module: Module sind eigenständige, gekapselte und vollständige Einheiten mit klar definierten Schnittstellen. Sie kön-

nen zu reaktionsfähigen Systemkonfigurationen zusammengefügt oder aus diesen entfernt werden, wobei ihre Beziehung zu anderen Modulen durch die passive „Plug and Play“-Infrastruktur bestimmt wird. Module sind so gekapselt, dass ihre Funktionalität nicht von der anderer Module abhängig ist – außer die passive Infrastruktur schreibt dies vor.

2) Passive Infrastruktur: Die passive Infrastruktur beschreibt die Interaktions- und Schnittstellenstandards der Module. Sie werden in Form von Normen und Regeln festgelegt, die sich auf die physischen Verbindungen und Datenverbindungen, Security- und Sicherheitsstandards sowie den Service der Module beziehen. Ziel ist es, ein Gleichgewicht zwischen der erforderlichen Vielfalt und Sparsamkeit zu finden, sodass Modulverbindungen erleichtert und gleichzeitig innovative Systemkonfigurationen zugelassen werden.

3) Aktive Infrastruktur: Die aktive Infrastruktur legt die Verantwortlichkeiten und Prozesse fest, um die agile Einsatzfähigkeit aufrechtzuerhalten. Sie stellt sicher, dass bei sich verändernden Anforderungen jederzeit neue Systemkonfigurationen erstellt werden können.

9),10), 11) Dove R.; LaBarge R. (2014): Fundamentals of Agile Systems Engineering, INCOSE Las Vegas 2014

Diese agile Systemarchitektur bringt einen Wandel mit sich – vom monolithischen zum Systemdenken nach den Prinzipien Wiederverwendbarkeit, Rekonfigurierbarkeit und Skalierbarkeit.

Im Zuge dessen stellt eine geeignete agile Systemarchitektur die Grundvoraussetzung für den Einsatz agiler Vorgehensmodelle in der Entwicklung von Hardware-Produktsystemen dar. Ein Großteil der agilen Standardwerke vernachlässigt diesen Zusammenhang jedoch, da sich ihr inhaltlicher Fokus nur auf die Softwareentwicklung bezieht. Entwicklungstechniken wie das Objektorientierte Programmieren (OOP) besitzen hier die strukturellen Voraussetzungen für die Anpassungsfähigkeit eines Systems. Auf diese Weise ermöglichen sie das einfache Ersetzen, Erweitern oder Neukonfigurieren von Elementen von Softwaresystemen in aufeinander folgenden Sprints. So werden durch iteratives Lernen Anpassungen vorangetrieben. Diese inhärente Anpassungsfähigkeit ist einer der zentralen Gründe für die Akzeptanz und rapide Verbreitung der agilen Ansätze in der Softwareentwicklung.¹² Die Architektur eines komplexen Systems muss sich darüber hinaus an vielen, oft konkurrierenden Kriterien ausrichten. Das Kriterium der Agilität ist hierbei nur ein Aspekt, der durch Kriterien wie Sicherheit, strukturelle Stabilität oder Kosten häufig in den Hintergrund gerät. So gibt es – dem bereits erwähnten LEGO-Beispiel folgend – durchaus Gründe, weshalb Gebäude oder Fahrzeuge nicht aus LEGO gebaut werden – obwohl sich damit ein hohes Maß an Agilität des Systems erreichen ließe. Eine ausgewogene Gewichtung der Kriterien im Architekturprozess ist somit ein entscheidender Faktor für eine erfolgreiche Systemgestaltung.

12) Dove R.; LaBarge R. (2014): Fundamentals of Agile Systems Engineering, INCOSE Las Vegas 2014

Übersicht der agilen Systemarchitektur am Beispiel des Spielzeug-Konstruktionssets Lego

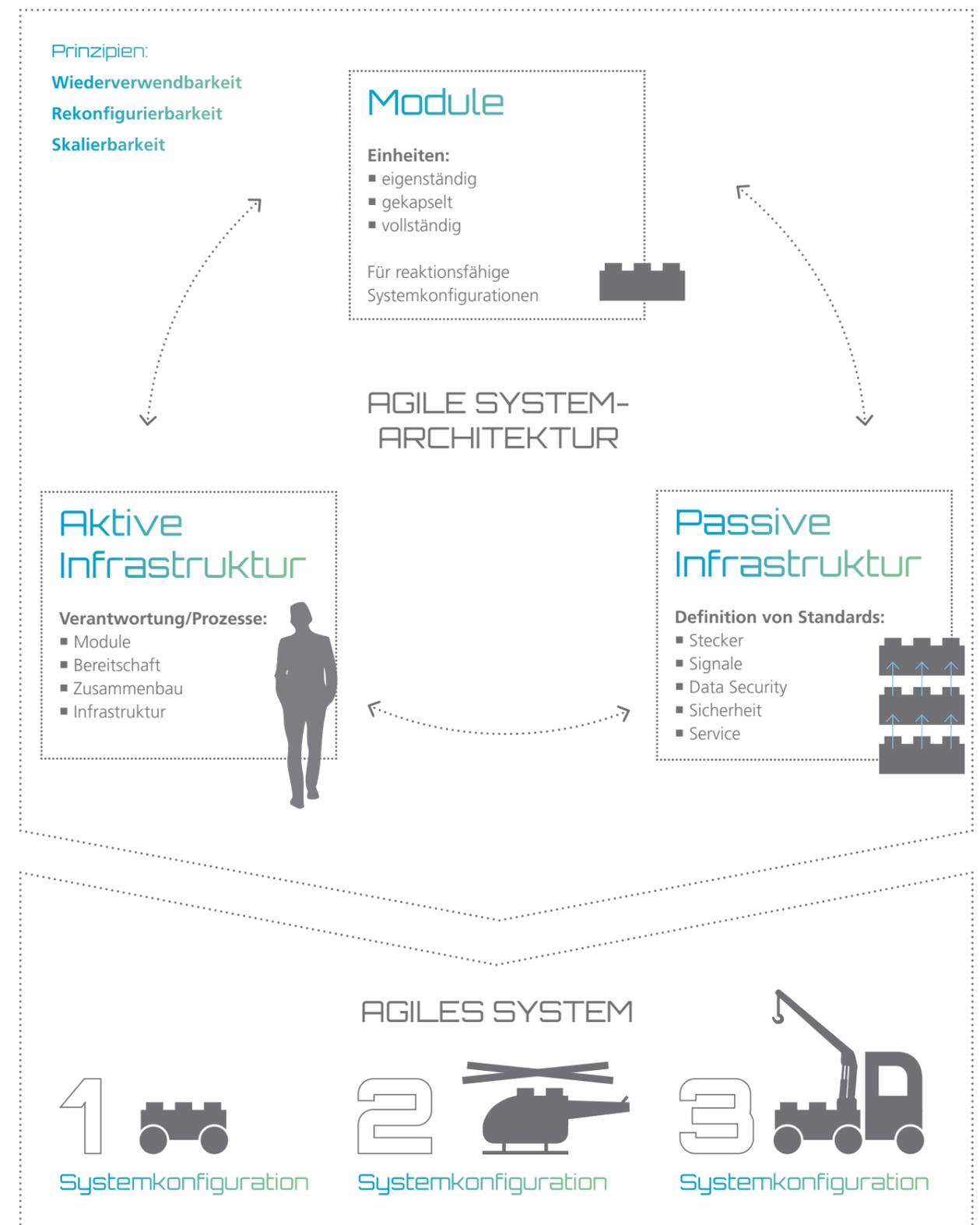


Abbildung 5

05

Die Agile Systems Engineering Transformation



Abschließend soll nun die Frage beantwortet werden, wie dieser kombinierte Ansatz konkret seine Umsetzung im Unternehmen finden kann. Hierzu bieten wir als MHP einen ganzheitlichen Ansatz mit Raum für kundenindividuelle Gegebenheiten und Ansprüche, der in Abbildung 6 verdeutlicht ist.

Die Transformation zu einem Scaled Agile @ Systems Engineering durchläuft drei Stages, wobei der Startpunkt von der individuellen Kundensituation und den Belangen des Unternehmens abhängt.

Mithilfe des MHP R&D Assessment kann ermittelt werden, wo die individuelle Transformation zu einem skalierten Agile Systems Engineering beginnt und wo sie beschleunigt werden kann.

In jedem Unternehmen sind die systemische Produktentwicklung sowie agile Arbeitsweisen unterschiedlich tief und breit verankert. Während in der Regel ein unternehmensweiter Produktentstehungsprozess vorhanden ist, variieren die agilen Prinzipien je nach Fachabteilung oder einzelner Team stark. Hier gilt es zunächst den Status zu erfassen. Bewährt hat sich hier das MHP Assessment: Es erfasst den strukturierten Zielzustand und der individuelle Fahrplan kann in Abstimmung mit dem Kunden abgeleitet werden.

Der individuelle Fahrplan definiert die Umsetzung in verschiedenen Geschwindigkeiten oder Reifegraden – abhängig von der bestehenden Projektorganisation. Die Flexibilität in der Transformationsgeschwindigkeit der drei Dimensionen (agiles Arbeitsmodell, agile Organisation und agile Systemarchitektur) ermöglicht eine adaptive Stages-Anpassung in den Ausprägungen des Scaled Agile @ Systems Engineering. So werden beispielsweise in einem Umsetzungsprojekt bei bereits etablierten agilen Projektteams ohne fundierte SE-Herangehensweise die agilen Elemente der Stages 2 und eventuell 3 vorgezogen. Zeitgleich werden die prozessualen und methodischen Grundlagen in den SE-Kernbereichen wie dem Anforderungsmanagement und in der Systemarchitektur gelegt. Ziel ist es, dass der Fahrplan den effizientesten Transformationsweg ermöglicht.

Stage 1: Für die Transformation und Beschleunigung der Veränderungsbereitschaft ist es wichtig, das Ziel für alle Beteiligten verständlich zu machen und die neuen Fähigkeiten zu fördern.

Nachdem der Fahrplan von den Stakeholdern verabschiedet und in die Teams kommuniziert wurde, kann Stage 1 beginnen. Hier gilt es, die agilen Prinzipien und Werte zu etablieren. Parallel beginnt ein Ramp-up der prozessualen und methodischen SE-Grundlagen in die Team-Backlogs. Flankierend sind hier Schulungen der Mitarbeiter sowie Coachings nötig. In der Regel starten die ersten Teams in bereits laufenden Projekten. Die Priorisierung der SE-Prozesse orientiert sich stark an der aktuellen Projektphase im Systemlebenszyklus. Daran orientiert werden die Standardisierung, Qualifizierung und Anwendung der Methoden in den Sprintplänen in die Backlogs eingearbeitet und umgesetzt. Parallel werden die Abläufe und Werte der agilen SE-Organisation adaptiert und implementiert.

Die Begleitung aller Beteiligten – insbesondere bei den Werten Offenheit und Transparenz – sowie der Führungskräfte im Agile Leadership sind die Basis für den nachhaltigen Umsetzungserfolg. Die Implementierung des kontinuierlichen Verbesserungsprozesses im Rahmen der Retrospektiven optimiert zudem die lernende Implementierung.

Diese Stufe kann mit geringer Varianz für die ersten fünf Pilotteams sechs Monate in Anspruch nehmen.

Stage 2: Sind die neuen Methoden in den Prozessen verankert und die neuen Fähigkeiten bei allen Beteiligten etabliert, kann die Transformation auf die Systemebene erweitert werden.

Nach der Team-Implementierung steht in Stufe 2 die Skalierung auf Systemebene im Fokus. Das agile Arbeitsmodell wird auf weitere Teams und gegebenenfalls weitere Unternehmensbereiche ausgeweitet und die Abläufe der Aufbauorganisation werden synchronisiert. Neben der Softwareentwicklung, die in der Regel bereits nach agilen Vorgehensmodellen arbeitet, und der Produktion, die schon Shop Floor einsetzt, folgen weitere Fachbereiche. Typischerweise sind QM, PM, Einkauf, Test und auch die Unternehmensleitung in den Fahrplänen enthalten. Hier werden angepasste agile Werkzeuge wie Scrumban oder Kanban etabliert und begleitet. Zeitgleich werden im SE die Grundlagen aus Stufe 1 weiter ausgebaut. Organisatorisch ist durch die Kombination der Teams auf einer aggregierten Ebene der entscheidende Schritt für die Skalierung zu gehen.

Mit der Team-Skalierung ist es zwingend notwendig, das passende IT-Toolset zu etablieren. Der Transformationsfahrplan muss die entsprechenden Aktivitäten in Bezug auf IT-Architektur, Umsetzung der Bebauung und Deployment mit dem Gesamtvorgehen synchronisiert enthalten.

Die Gesamtsystemarchitektur dient als Blaupause für die Informationsflüsse und muss sich in der Aufteilung und Definition der Teams sowie deren Wechselwirkungen untereinander abbilden. Zeitgleich wird die Systemarchitektur stärker auf das Kriterium der Agilität ausgerichtet, indem vollständige, wiederverwendbare, skalierbare und rekonfigurierbare Module zum Einsatz kommen. Schnittstellen werden standardisiert, um die Kompatibilität zu maximieren. Die Anpassung der Organisation und Gesamtarchitektur korreliert stark und ist nur durch einen lernenden iterativen Prozess mit Retrospektiven zu realisieren. Insgesamt liegt der Fokus auf dem Mindset sowie der Umsetzung in den Projekten.

Diese Stufe dauert etwa sechs bis 24 Monate – abhängig von der Komplexität der Systeme.

Stage 3: Das Ziel der Transformation ist die Etablierung einer adaptiven Agile Systems Engineering Aufbau- und Ablauforganisation sowie die konsequente Umsetzung der agilen Systemarchitektur.

In Stufe 2 sind bereits die relevanten Weichen gestellt, sodass der finale Schritt unternehmensweit umgesetzt werden kann: die nachhaltige Vernetzung der verschiedenen Einheiten sowie die Synchronisation aller SE-Teams. Mit diesem Reifegrad entfaltet sich der Nutzen der kontinuierlichen Lieferung von Mini-

mal Viable Systems in Dreimonatszyklen. Auch die agilen und SE-Prozesse sind final vollständig zu adaptieren und umzusetzen. Oftmals wird in der Stufe 3 auch die Aufbauorganisation an die Bedürfnisse des SA@SE angepasst, um den Abläufen und Rollen Rechnung zu tragen. Dies ist jedoch für ein funktionsfähiges SA@SE nicht zwingend erforderlich – sofern die Prozesse sowie das Mindset aus den vorhergehenden Stufen konsequent gelebt werden. Aus SE-Sicht steht die weitere iterative Anpassung der Organisation und vor allem die der Systemarchitektur im Vordergrund. Die Etablierung der Modulari-

sierung, Schnittstellenstandardisierung, passiven Infrastruktur sowie die Erweiterung um eine aktive Infrastruktur verbessern die agile Einsatzfähigkeit in der Praxis. Das finale Ergebnis ist eine selbstlernende Organisation, die im komplexen Umfeld moderner Produktentwicklung flexibel und nachhaltig im Markt agieren und bestehen kann.

Eine erfolgreiche Umsetzung ist an einer eingeschwungenen, kontinuierlichen und dauerhaften Selbstorganisation der Aufbau- und Ablauforganisation sowie der Systeme erkennbar.

Vollständige Transformation nach dem MHP Beratungsansatz

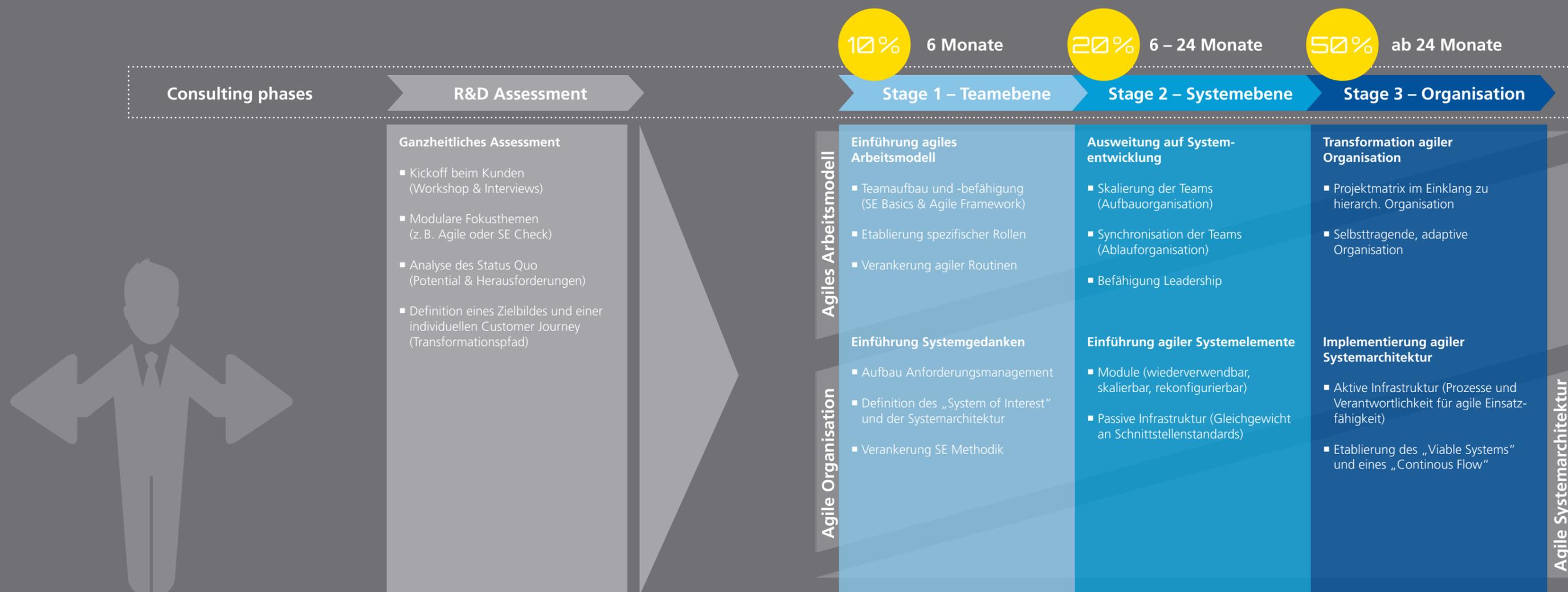


Abbildung 6



Ausblick

Wir sind überzeugt: Scaled Agile @ Systems Engineering kann alle Antworten für eine erfolgreiche Transformation im R&D-Bereich liefern. Besonders in Zeiten, in denen die Wettbewerbs- sowie die Reaktions- und Anpassungsfähigkeit stärker in den Fokus rückt und in denen Software in vormals mechanischen und elektrischen Produkten gleichermaßen neue Chancen und Handlungsfelder eröffnet.

Unser White Paper macht deutlich, dass Agilität und System Engineering als zentrale Bestandteile einer reaktionsschnellen und adaptiven Produktentwicklung anzusehen sind. Die Fragestellungen und Herausforderungen, die uns von der Management- bis zur Arbeitsebene unserer Kunden gespiegelt werden, lassen sich mit SA@SE zukunftsorientiert adressieren.

Folgende drei Erfolgsfaktoren sind an dieser Stelle nochmals abschließend hervorzuheben:

- 1) Die Etablierung von agilen und System-Engineering-Methoden sowie der prozessualen Verankerung entlang des Wertstroms bilden das Fundament einer erfolgreichen Veränderung.**
- 2) Die Agile Systems-Engineering-Organisation bedient sich abhängig von den Rahmenbedingungen bei Best Practices des etablierten Frameworks SAFe®.**
- 3) Die konsequente Umsetzung einer agilen Systemarchitektur ermöglicht maximale Flexibilität bei der Umsetzung neuer Markt- und Kundenanforderungen an die System- und Funktionsgestaltung.**

Aktuell gibt es viele Entwicklungen, die die Zukunftsthemen Agile und Systems Engineering weiter verschmelzen. Insbesondere seien hier die Arbeitsgruppen der INCOSE und deren deutscher Ableger genannt, die GfSE. MHP ist in den Arbeitsgruppen beider Institutionen (INCOSE Agile Systems und SE sowie GfSE Agile Systems Engineering) vertreten und gestaltet das Thema entscheidend mit.

Ebnen auch Sie sich diesen nachhaltigen und gewinnbringenden Weg. Unser Autorenteam aus Systems Engineering und Agile-Experten steht Ihnen gern für weitere Fragen zur Verfügung.

Ansprechpartner:



Dr. Sebastian Schröter
Service Developer
Systems Engineering
Sebastian.Schroeter@mhp.com



Andreas Feil
Service Developer
Agile Engineering
Andreas.Feil@mhp.com

Autorenteam:

Dr. Sebastian Schröter, Andreas Feil, Daniel Haase, Patrick Reimund

An aerial photograph of a winding asphalt road that curves through a dense, lush green forest. The road has white dashed lines in the center and solid lines on the edges. Two cars are visible: a dark-colored car on the upper curve and a red car on the lower curve. The forest is composed of various types of trees, creating a rich, textured green canopy.

ENABLING YOU
TO SHAPE A BETTER
TOMORROW >>>

Fotocredits

Cover @shutterstock – sutadimages
Seite 9 @shutterstock – Rawpixel.com
Seite 10 @shutterstock – GLYPHstock
Seite 19 @shutterstock – WHYFRAME
Seite 24 @shutterstock – Chan2545
Seite 30-31 @iStock – cicerocastro

Layoutgestaltung
Freiland Design

MHP: DRIVEN BY EXCELLENCE

16 MHP Offices in Germany, United Kingdom, USA, China and Romania



Germany

Ludwigsburg
(Headquarters)
Berlin
Essen
Frankfurt a. M.
Ingolstadt
Munich
Nuremberg
Wolfsburg

International

Atlanta (USA)
Birmingham (United Kingdom)
Cluj-Napoca (Romania)
Timișoara (Romania)
Shanghai (China)
Tel Aviv (Israel)

www.mhp.com