

---

# WAS PASSIERT IN DER FORSCHUNG?

Sven Theobald

---

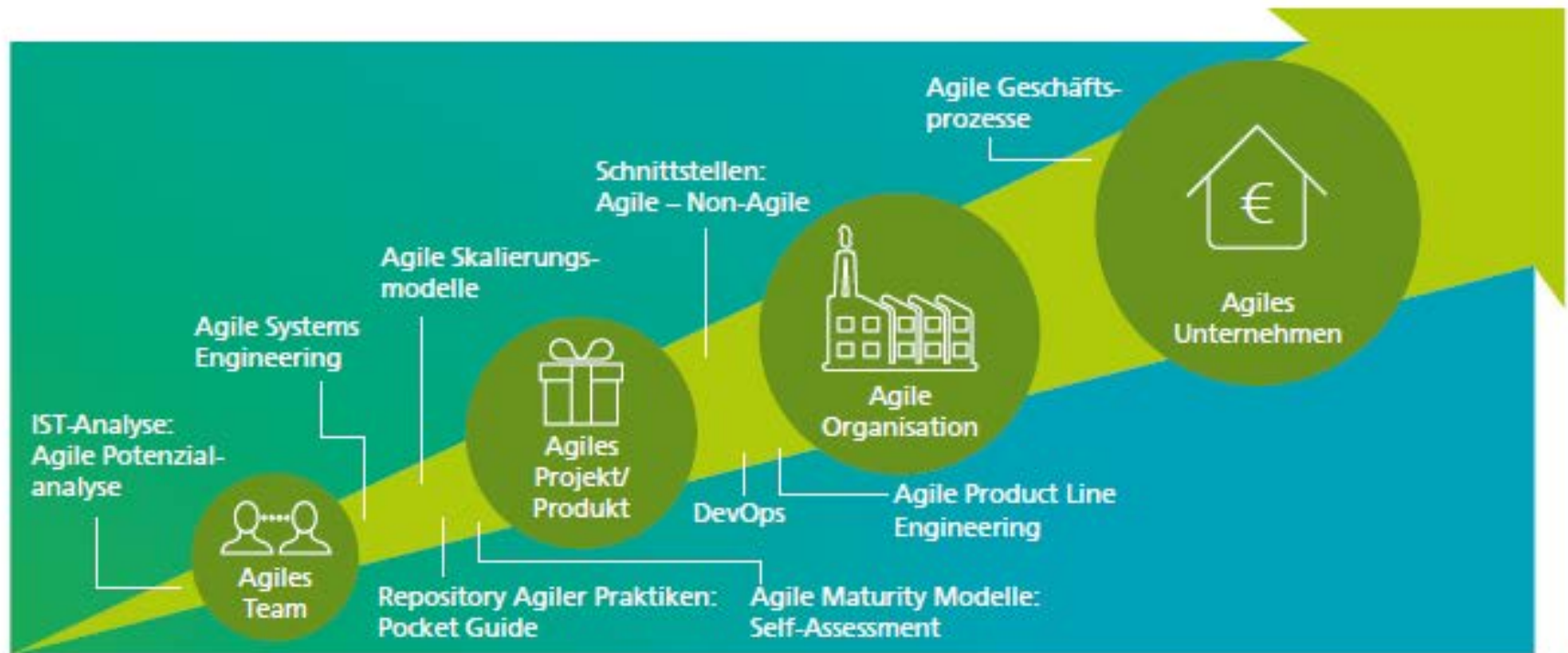


**Fraunhofer**

**IESE**

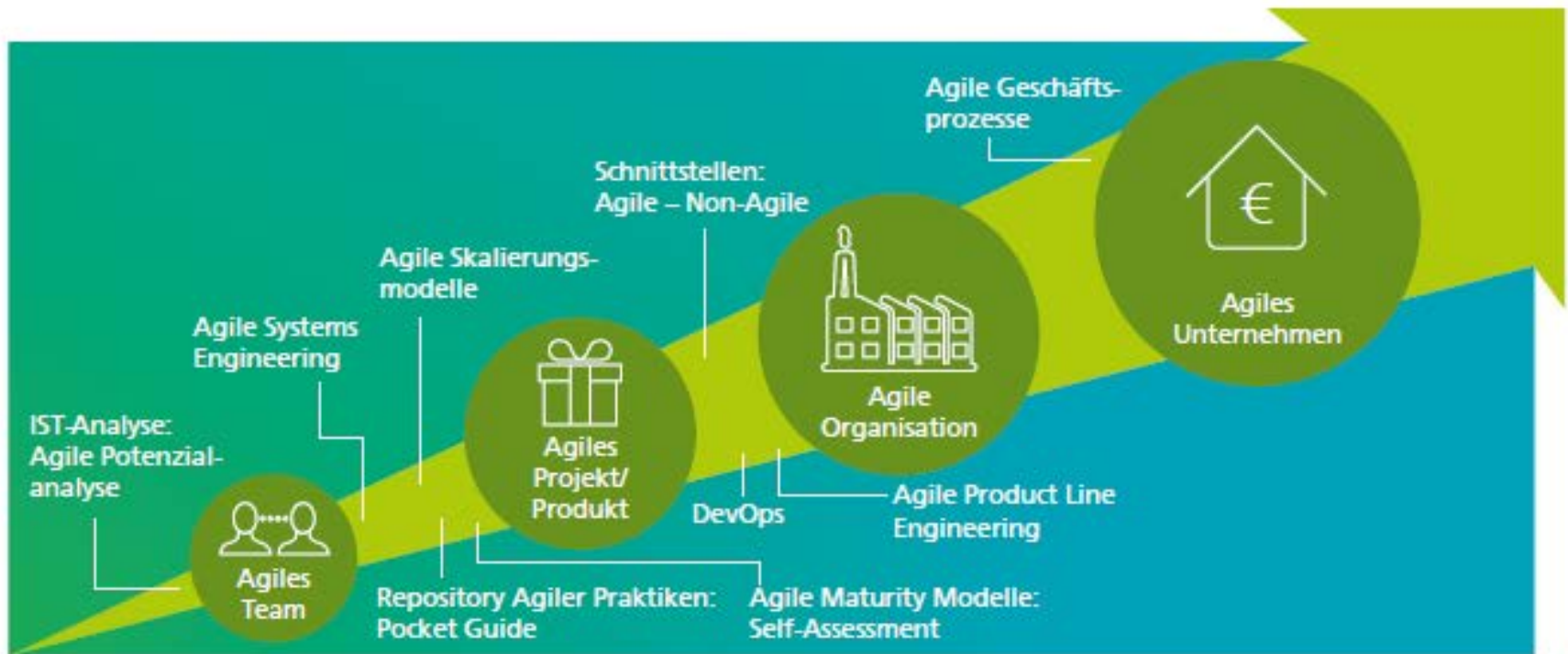


# Übersicht Agile Transition



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

# Repository agiler Praktiken



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

# Agile Software Development

## Agile Methods / Frameworks

(Scrum, XP, Kanban, Crystal, FDD, ...)

## Agile Practices

(Simple Design, Continuous Integration, Collective Ownership, Pair Programming, Coding Standards, Refactoring, Testing, ...)

## Agile Principles

(Early and continuous delivery of valuable software, Business people and developers must work together, Welcome changing requirements, Self-organization of the team, Working software is the primary measure of progress, ...)

## Core Values

(Individuals and interactions, Working software, Customer collaboration, Responding to change)

## Agile Manifesto

# Agile Praktiken

**Agile Praktiken** sind Instruktionen, wie Aufgaben, Aktivitäten oder Anleitungen, mit einem speziellen Fokus auf Aspekte der Software-Entwicklung, die sich an einem oder mehreren agilen Kernwerten oder Prinzipien orientieren [Diebold, Zehler]

Continuous Integration  
System Metaphor  
Small Releases  
Coding Standard  
Collective Code Ownership  
Daily Stand-Up  
Code Reviews  
Quick Design Sessions  
Product Backlog  
User Stories  
Burn-Down Chart  
Sustainable Pace  
Retrospective  
User Story Mapping  
Backlog Grooming  
Retrospective  
Definition-of-Done  
TDD  
Pair Programming  
Onsite Customer  
Simple Design

...

# Liste agiler Praktiken (über 120 Praktiken)

- Feature-driven Development
  - Features in iteration customer-visible
  - fitness for business purpose is the essential criterion for acceptance of deliverables
  - Frequent delivery
  - Formalize Contract Models
  - Given-When-Then
  - Holistic diversity strategy
  - Incremental Development
  - Incremental Design
  - Individual Class (Code) Ownership
  - Informal Design
  - Inspection
  - Integrated Dev/QA
  - Integration Testing
  - INVEST
  - Issue Tracking / Bug Tracking
  - Iterate to Another Artifact
  - Iteration Reviews
  - Iterative development
  - Just rules
  - Manage requirements
  - Methodology-tuning technique
  - Metrics
  - Milestone Retrospective
  - Mock Objects
  - Model with Others
  - Project Chartering
  - Project Monitoring & Control
  - Prove it with Code
  - Quick Design Session
  - reduce intermediate work products
  - Refactoring
  - Regular Builds
  - Relative Estimation
  - Requirement Prioritization
  - Requirements Management
  - Retrospective
  - Revision and Review
  - Role-Feature-Reason
  - Root Cause Analysis
  - Rules of Simplicity
  - Small Release Cycles
  - Scrum Master
  - Scrum of Scrums
  - Short Iterations
  - Shrinking Teams
  - Sign Up
  - Sign Up for Tasks
  - Simple Design
  - Single Code Base
  - Single Sourcing Information
  - Sitting Together / Common Workspace
  - Slack
- ...



# BAUSTEINE

Anforderungsworkshop	☆☆☆☆☆	Microservices >	★★★★★
Architecture Sprint >	☆☆☆☆☆	Mocking Simulation >	☆☆☆☆☆
Aufgabenorganisationssysteme >	★★★★★	On-Site-Customer >	☆☆☆☆☆
Backlog >	★★★★☆	Pair Programming >	★★★★★
Backlog Refinement >	★★★★☆	Personas >	★★★★☆
Behaviour Driven Development >	☆☆☆☆☆	Pflichtenheft >	★★★★★
Brainstorming >	★★★★★	Planning Meeting >	★★★★★
Burn Chart >	★★★★★	Planning Poker >	★★★★★
Closing Documentation	☆☆☆☆☆	Priorisierung >	★★★★☆
Code Freeze >	★★★★★	Product Canvas >	☆☆☆☆☆
Code Generation >	★★★★☆	Product Vision Board >	☆☆☆☆☆
Code Inspection >	☆☆☆☆☆	Prototyping >	★★★★★
Code Review >	★★★★★	Refactoring >	★★★★★
Coding Styleguide >	★★★★☆	Regression Testing >	★★★★★
Community of Practice >	★★★★☆	Release-Zyklus-Planung >	★★★★★
Continuous Delivery >	☆☆☆☆☆	Retrospective >	★★★★★
Continuous Deployment >	☆☆☆☆☆	Review >	★★★★★
Continuous Integration >	★★★★☆	Schätzung des Aufwands >	☆☆☆☆☆
Crowd-Testing >	☆☆☆☆☆	Scrum-of-Scrums >	★★★★★
Definition of Done >	★★★★★	Service Virtualization >	★★★★★
Definition of Ready >	★★★★☆	Spezifikation >	☆☆☆☆☆

## STAND-UP MEETING ★★★★★

Diese täglich stattfindende 15-minütige Team-Besprechung wird stehend absolviert, wodurch die Abstimmungen dynamischer und fokussierter ablaufen. Jedes Mitglied eines (Entwicklungs-)teams stellt dabei allen anderen Teilnehmern kurz folgende Punkte dar:

- Welche Aufgaben habe ich seit dem letzten Stand-Up Meeting (meist am...

[Weiterlesen](#)

[ZURÜCK ZUR BAUSTEIN ÜBERSICHT](#)

[BAUSTEIN BEWERTEN](#)

Beschreibung

Mögliche Anwendungsfälle für diesen Baustein...

### ZIELE

- Ressourcenmanagement >
- Projekttransparenz >
- Zeittransparenz >
- Projekt-Demokratisierung >
- Mitarbeitermotivation >
- Internes Wissensmanagement >

Ziele

Synonyme / Abkürzungen / Übersetzungen	+
Vor- / Nachteile	+
Anwendungsparameter	+
Chancen-Nutzen-Analyse	+
Aufwandsbetrachtung	+
Referenzen	+

Eigenschaften

## BEWERTUNGEN

★★★★★

NAME: Johannes Wahl

**BEMERKUNG:** Wir haben das Daily Stand-Up in einem Projekt das über 3 Wochen mit einem Team von 3 Mitarbeitern ausprobiert und durchwegs für gut befunden. Der regelmäßige Austausch über erledigtes, die nächsten Schritte und aktuelle Probleme hat es uns erleichtert den Fortschritt zu bewerten und dem Team ein möglichst Hindernis freies arbeiten zu ermöglichen. Probleme die frühzeitig kommuniziert werden können auch rechtzeitig gelöst werden. Sicher ein Baustein den man einfach mal ausprobieren sollte.

ZIELE:

Zeittransparenz ★★★★★  
Projekttransparenz ★★★☆☆

★★★★☆

NAME: anonym

ZIELE:

Zeittransparenz ★★★★★  
Mitarbeitermotivation ★★★★★  
Ressourcenmanagement ★★★★★  
Projekt-Demokratisierung ★★★★★  
Projekttransparenz ★★★★★  
internes Wissensmanagement ★★★★★

Bewertungen



# Extract of Stand-Up Meeting

## Anwendungsparameter

### VORBEDINGUNGEN:

Voraussetzung für das Stand-Up Meeting ist die gemeinsame Arbeit eines (Entwickler-) Teams an einem gemeinsamen Produkt oder Projekt.

Außerdem ist eine wichtige Vorbedingung, dass alle Teilnehmer zur selben Zeit, idealerweise am selben Ort und in derselben Sprache das Stand-Up-Meeting durchführen (vgl. aber Variationsparameter).

Eine Grundvoraussetzung ist eine offene Umgangsform, in der Transparenz und Selbstverpflichtung geachtet werden.

### NACHBEDINGUNGEN:

Einzig systematische Nachbedingung ist ein besserer gemeinsamer Wissensstand.

### VARIATIONSPARAMETER:

Bei eingespielten Teams muss die Moderation nicht mehr unbedingt vom „Scrum Master“ (Moderator) vorgenommen werden.

Eine Besonderheit ist die Integration von Video-Telefonie (z.B. via Skype):

Die (passive) Teilnahme des „Product Owner“ (Produktverantwortlicher) ist nicht unbedingt erforderlich.

### ANWENDUNG:

	Initialisierung	Planung	Umsetzung	Controlling	Abschluss
Anforderung		✓		✓	
Entwurf		✓		✓	
Implementierung		✓		✓	
Test		✓		✓	
Deployment		✓		✓	

## Vor- / Nachteile

### VORTEILE:

Der aktuelle Stand der Arbeiten und Aufgaben jedes Teammitglieds wird allen Teammitgliedern deutlich und nachvollziehbar (Transparenz) und das Team entwickelt durch den wiederkehrenden und ritualisierten Austausch über die Aufgaben ein hohes Maß an (gemeinsamer) Selbstverpflichtung, was wiederum zu einer verbesserten Organisations-Demokratisierung führt.

Dadurch, dass alle wissen, was der andere macht, gewinnt jeder auch fortwährend Transparenz darüber, wo das Projekt steht. Übernimmt ein Kunde selbst eine Rolle im Stand-Up-Meeting bedeutet das auch für ihn eine gestiegene Projekttransparenz.

### NACHTEILE:

Einzigster Nachteil ist die Notwendigkeit der Vor-Ort-Präsenz (vgl. aber Variationsparameter), aber in jedem Fall die Notwendigkeit der zeitlichen Synchronisation.

## Chancen-Nutzen-Analyse

### FALLSTRICKE:

Gerade in den ersten „Stand-Up Meetings“ kann es immer wieder zu Verzögerungen und Unpünktlichkeiten kommen.

Ein häufiger Fallstrick ist die strikte Einhaltung der „Time Box“ von 15 Minuten. Längere Stand-Ups sollten sich alleine aufgrund des Stehens verbieten.

Teammitglieder können nicht immer strukturiert und gezielt das eigentliche Ziel der schnellen Synchronisierung und gemeinschaftlichen Selbstverpflichtung verfolgen oder respektieren.

Das Stand-Up Meeting kann zunächst durchaus als unnötige Bürde und Zeitverschwendung gesehen werden. Es erfordert zunächst durchaus ein hohes Maß an Team-Disziplin und Selbstverpflichtung, den langfristigen Nutzen hinter diesem – immerhin lediglich 15-minütigen – Synchronisationstermin zu erfahnen.

Das Stand-Up Meeting kann vom Team und vom Management als Leistungs- und Fortschrittskontrolle missverstanden und instrumentalisiert werden.

### RISIKOFAKTOR: GERING

Dauernde Leistungskontrollen – in solche können Stand up Meetings ausarten – sind arbeitsrechtlich ein Problem

### UNDO-SCHRITTE:

Da keine echte Etablierung nötig ist, sondern eine direkte Anwendung möglich, sind keine schwergewichtigen Undo-Schritte des Stand-Up-Meetings nötig. Es genügt meist, die Terminreihe schlichtweg abzusagen und das „Stand-Up Meeting“ nicht mehr zu veranstalten.

## Aufwandsbetrachtung

### AUFWAND ERSTER DURCHSTICH: SEHR GERING

Minimal erhöhter Aufwand um ein Überziehen der Timebox zu verhindern

### AUFWAND ETABLIERUNG: SEHR GERING

Überzeugungsarbeit: Der geringe Zeitverlust muss gegen die gewonnene Transparenz gestellt werden. Werkzeuge o.Ä. werden nicht benötigt.

### AUFWAND BETRIEB: SEHR GERING

Timboxing max. 15 Minuten (Scrum).

## Referenzen

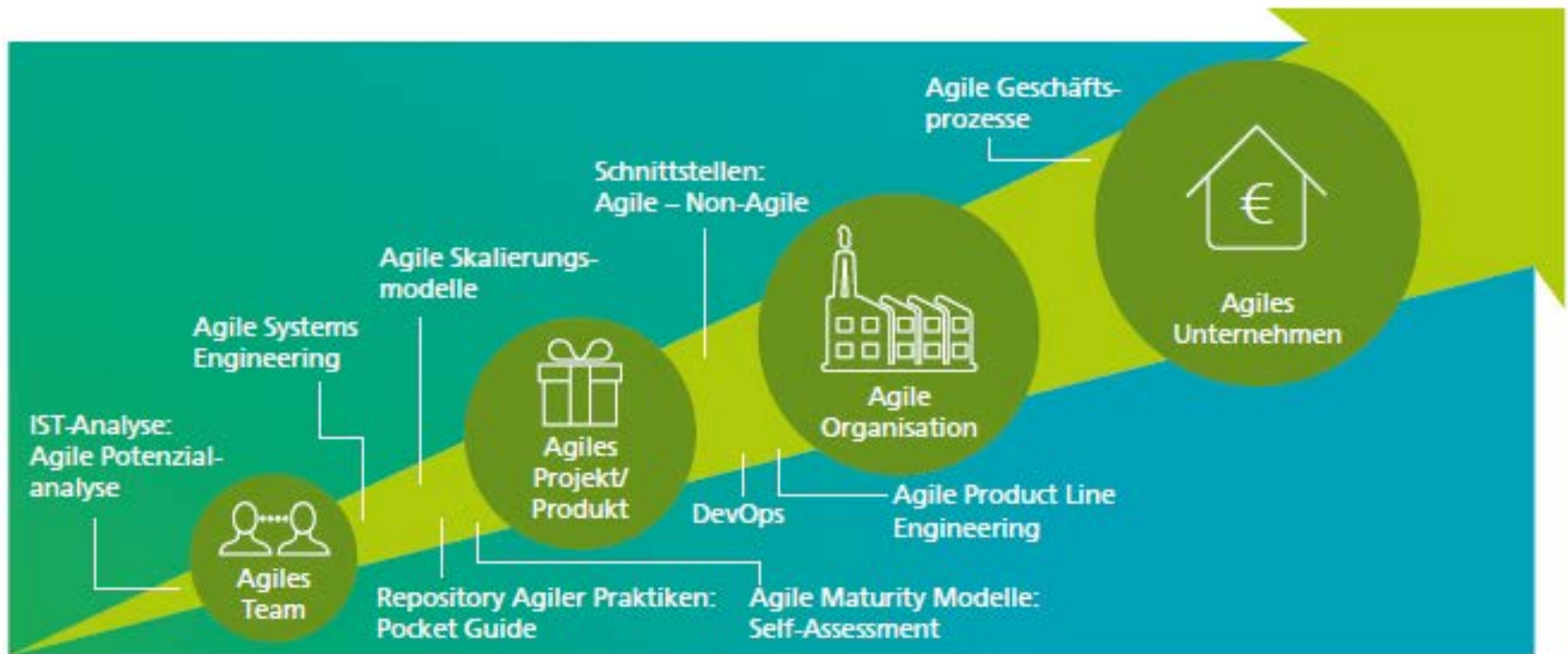
### REFERENZEN [REFERENCES]:

<https://www.mountaingoatsoftware.com/agile/scrum/daily-scrum>

<http://www.martinfowler.com/articles/itsNotJustStandingUp.html>

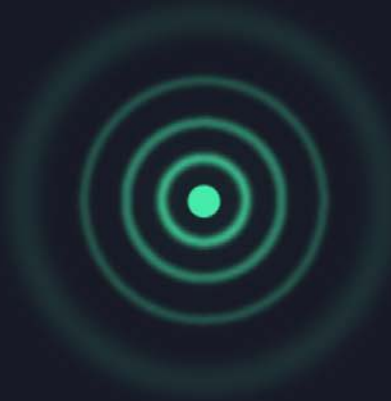
<https://infos.seibert-media.net/display/Websoftware/Standup-Meetings>

# Agile Potenzialanalyse



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

# ProKoB – Einstieg



## ZIELE

Was soll verbessert werden? Wo können noch Potenziale gefördert werden? Wie lässt sich die Effizienz steigern?

**JETZT ENTDECKEN**



## BAUSTEINE

Welche Tools & Methoden stehen mir zur Verfügung? Und wie werden sie angewendet?

**JETZT ENTDECKEN**

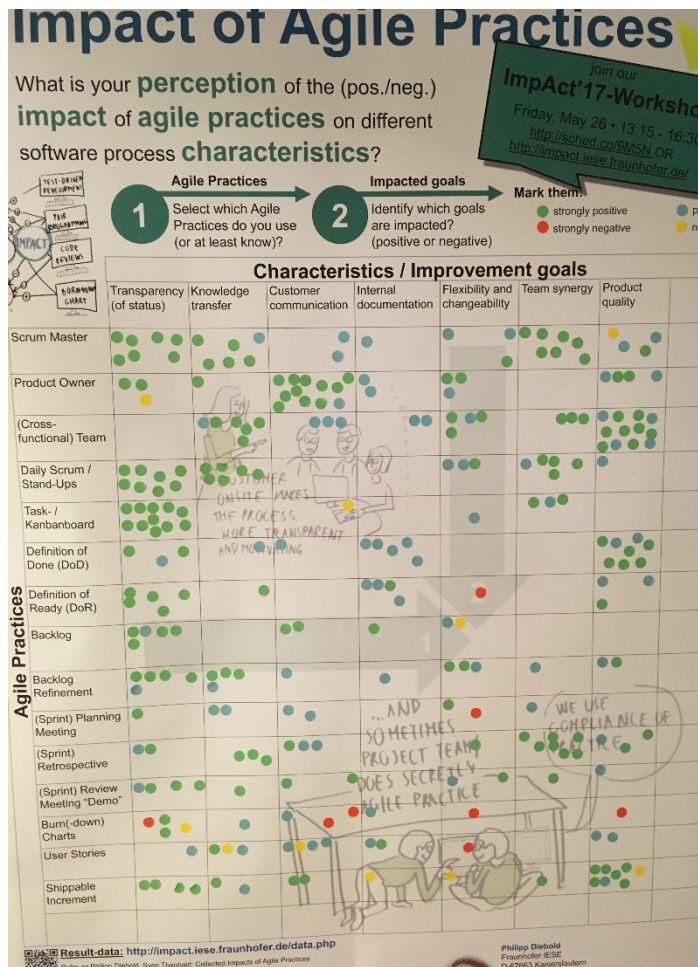


## ENSEMBLES

Welche Ensembles gibt es bereits? Wie werden Bausteine sinnvoll kombiniert? Was darf ich von welchem Ensemble erwarten?

**JETZT ENTDECKEN**

# Erhebung der Impacts agiler Praktiken



Collective Ownership	1.5 (2)			0.88 (8)		0.71 (7)	1.6 (10)	1.14 (14)	1 (1)	1.5 (12)		1.5 (6)		
Communities of Practice	1 (1)			0.8 (5)		1 (1)	1.5 (23)	1.25 (8)	2 (2)	1.2 (5)		1.33 (3)		
Continuous Delivery	1.33 (6)		1 (1)	1.5 (2)	1 (1)		0 (2)	1 (1)		1 (1)		1 (3)		
Continuous Deployment	1.5 (5)	1 (1)	1 (1)	1 (1)				1.75 (6)		1 (1)		1 (1)		
Continuous Integration	0.5 (2)	1 (1)	1 (1)	1.5 (19)	1 (1)	1 (1)	0.65 (3)	0.5 (2)	1 (1)	2 (4)	1 (1)	1.8 (16)		
Cross-Functional Team	1.29 (7)	1 (1)		1.5 (18)	1 (1)		0.57 (7)	1.45 (37)	1.71 (35)	1 (1)	1.6 (21)	1.75 (12)	1.83 (6)	
Daily Meeting	1 (7)			1 (1)	1.42 (19)	1 (1)	1.5 (4)	1.16 (3)	1.43 (14)	1 (1)	1.65 (56)	1.5 (2)	1 (2)	1.69 (30)
Definition of Done	1.33 (3)	1 (1)		1 (1)			1.21 (14)	1 (3)	1.76 (34)	1 (1)	1 (5)	1.2 (5)	1.5 (16)	
Definition of Ready	1.5 (2)	0.5 (2)					1.5 (12)	1.4 (3)	1.6 (15)	0.75 (4)	1 (1)	1.8 (8)	1.8 (9)	
Epic(s)	1.5 (12)			0.35 (3)			0.75 (4)	1 (1)					1 (4)	
Iteration Reviews	1.33 (36)	1 (1)		1.5 (10)			1.35 (3)	1.39 (18)	1.48 (31)	1.5 (13)	1 (1)	1.71 (42)		
Iterative development	1.5 (17)	1 (1)	1 (1)	1.5 (19)			1.35 (11)	1 (1)	1.5 (16)	1 (1)	1 (1)	1.7 (7)		
Minimum Viable Product	1.17 (12)			0.5 (6)			0 (2)	1.2 (15)		1 (1)		1.25 (8)		
On-site customer	1.7 (13)	1 (1)		1 (4)	1 (1)		1.45 (6)	2 (2)		0.65 (3)	2 (4)	1.5 (4)		
Pair Programming	1 (1)			0.5 (7)		1 (1)	0.5 (3)	1.75 (9)	1.5 (15)	1.5 (11)	1 (1)	0.5 (2)	2 (4)	
Peer Reviews							1 (1)	1 (1)	1 (1)	1.35 (3)		2 (2)		
Personas	1.5 (11)	1 (1)		1 (1)			1.2 (5)	1.5 (8)	1.5 (13)	1 (2)	1 (1)	1.6 (3)		
Planning meeting	1.47 (15)		1 (1)	0.45 (14)	1 (1)	1 (1)	1.2 (4)	1.4 (20)	1.5 (13)	1.21 (19)	1 (1)	1.84 (37)		
Planning Poker	1 (1)			1 (1)	1 (1)	1 (1)	1 (1)	1.5 (8)	1.5 (2)	1.35 (15)		1.35 (6)		
Product Canvas	1.5 (2)						1 (2)	1 (1)	1.5 (2)	1 (1)		1 (1)		
Product Owner	1.76 (45)			1.22 (18)			0.1 (10)	1.24 (17)	1.32 (28)	1 (1)	1.2 (10)	1.35 (3)	1.3 (17)	

# Scrum Praktiken und ihre Impacts

	Customer collaboration (46)	Customer value (16)	Development costs (2)	employee motivation (7)	Flexibility and ability to change (48)	Foster agile culture (12)	Internal communication (7)	Internal documentation (40)	Knowledge transfer (48)	Product quality (51)	Productivity (11)	Team Spirit (1)	Team synergy (45)	Throughput (6)	time to market (26)	Transparency (of project status) (49)
Backlog (10)	1.08 (12)	1.67 (3)			1.06 (17)		1 (2)	1 (6)	1 (6)	1 (9)			2 (1)		2 (1)	1.8 (30)
Backlog grooming (10)	1.58 (12)			2 (1)	1.31 (16)			0.63 (8)	1.6 (20)	1.44 (16)			1.33 (6)	1 (1)	2 (1)	1.83 (12)
Scrum Master (11)	1.27 (11)				1.62 (21)			0.4 (5)	1.46 (26)	1 (15)	2 (2)	2 (1)	1.66 (44)	1 (1)	1.25 (4)	1.69 (39)
Product Owner (9)	1.76 (45)				1.22 (18)			0.1 (10)	1.24 (17)	1.32 (28)	2 (1)		1.2 (10)		1.33 (3)	1.53 (17)
Self-organizing team (7)					1.83 (6)	2 (2)		-1 (4)	1.44 (9)	1.5 (4)			1.93 (14)			2 (2)
Cross-Functional Team (11)	1 (1)	2 (1)			2 (2)	2 (1)		2 (1)	1.64 (11)	1.8 (10)	2 (1)		1.29 (7)			1 (1)
Daily Meeting (13)	1 (7)			1 (1)	1.42 (19)	2 (2)	1.5 (4)	0.29 (7)	1.44 (45)	1.43 (14)	2 (1)		1.67 (54)	1.5 (2)	1 (1)	1.69 (77)
Definition of Done (9)	1.33 (3)	1 (3)			1 (2)			1.23 (13)	1 (3)	1.76 (33)			1 (6)		1.75 (4)	1.53 (15)
Planning Meeting (10)					1.5 (2)		1 (2)		1 (1)				1 (1)			1.5 (4)
Retrospective (12)	1.5 (6)			2 (1)	1.64 (44)	1.5 (2)	2 (2)	2 (1)	1.62 (29)	1.3 (23)	2 (2)		1.7 (63)		2 (1)	1.63 (30)
Shippable Increment (8)	1.57 (7)				1.4 (5)			-1 (2)	1.5 (2)	1.64 (11)			2 (2)		1.43 (7)	1.89 (9)

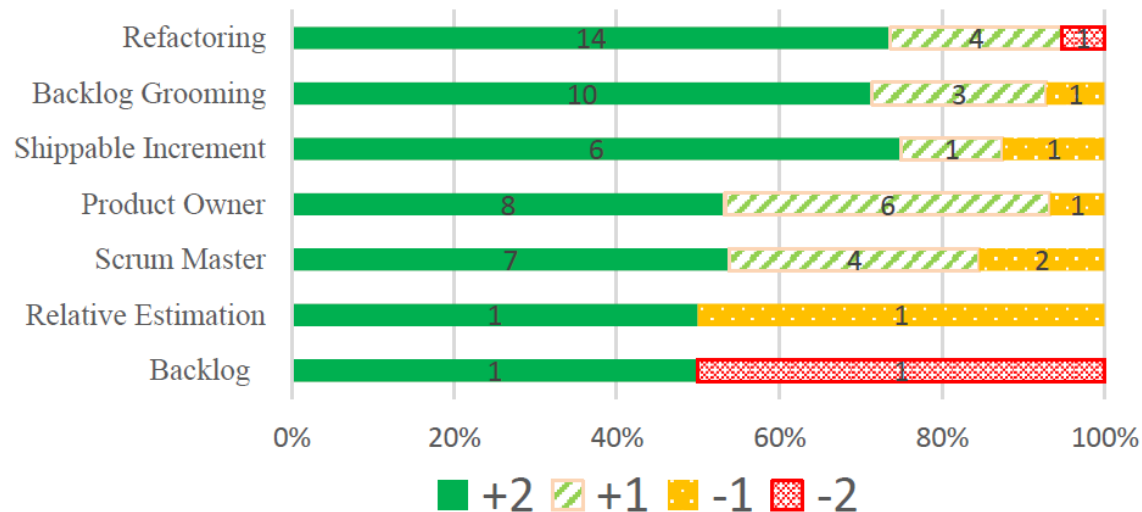
# Filter nach Produktqualität

**Table 2.** Most beneficial Agile Practices with at least 10 ratings

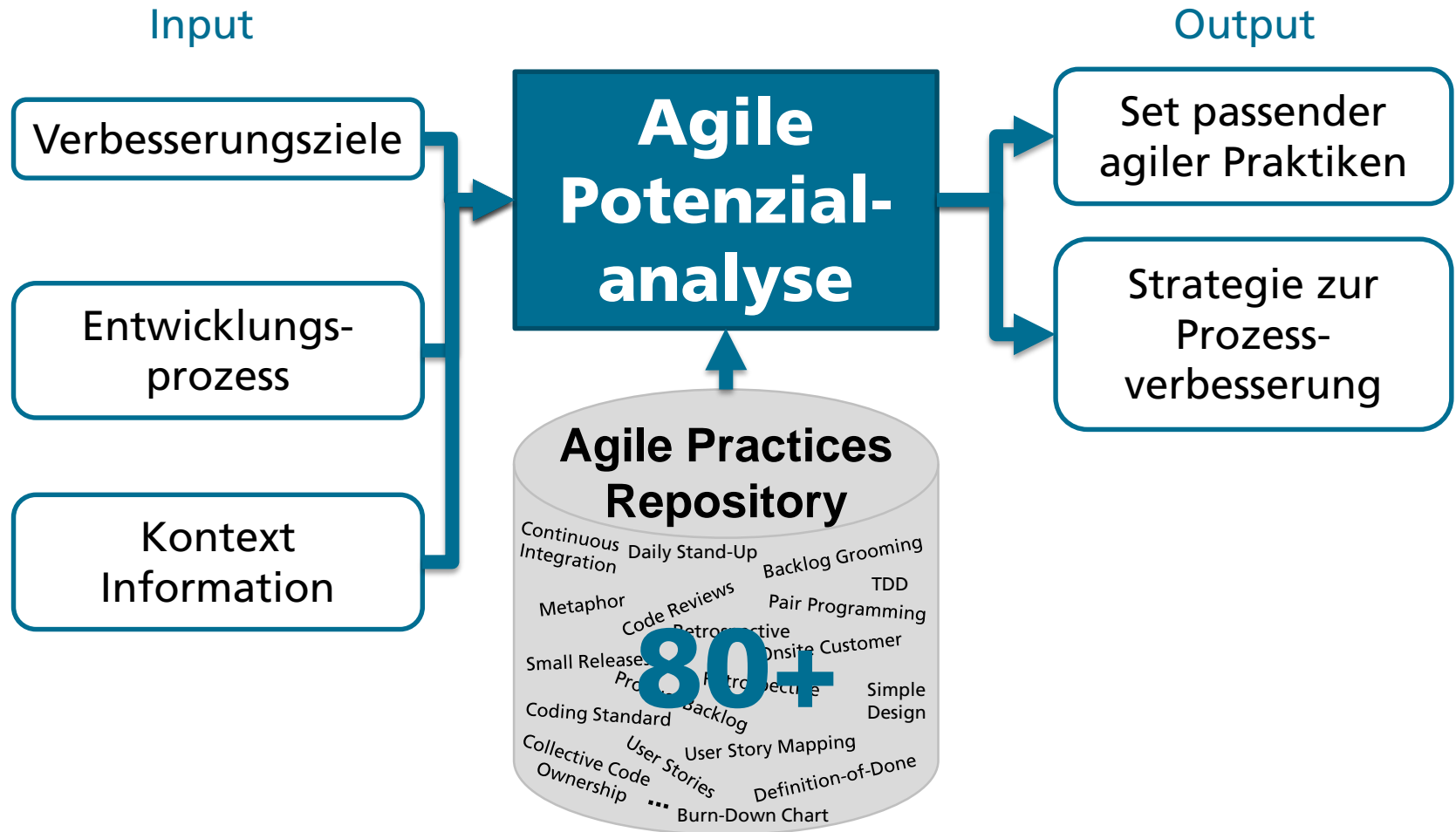
<b>Beneficial Agile Practice</b>	<b>Average</b>	<b>Count</b>	<b>+2</b>	<b>+1</b>	<b>-1</b>	<b>-2</b>
Test Driven Development	1.93	15	14	1		
Pair Programming	1.83	12	10	2		
Continuous Integration	1.77	13	10	3		
Cross-Functional Team	1.75	28	21	7		
Definition of Done	1.75	24	18	6		
Definition of Ready	1.75	12	9	3		
Unit Testing	1.64	14	9	5		
Refactoring	1.58	19	14	4		1
Backlog grooming	1.57	14	10	3	1	
Iteration Reviews	1.47	19	9	10		
Product Owner	1.4	15	8	6	1	
Retrospective	1.36	14	9	5		
Scrum Master	1.23	13	7	4	2	

# Umgang mit unterschiedlich eingestuften Praktiken

Practices with ambiguous impact on product quality: Distribution of the amount of ratings over the scale (+2,+1,-1,-2).

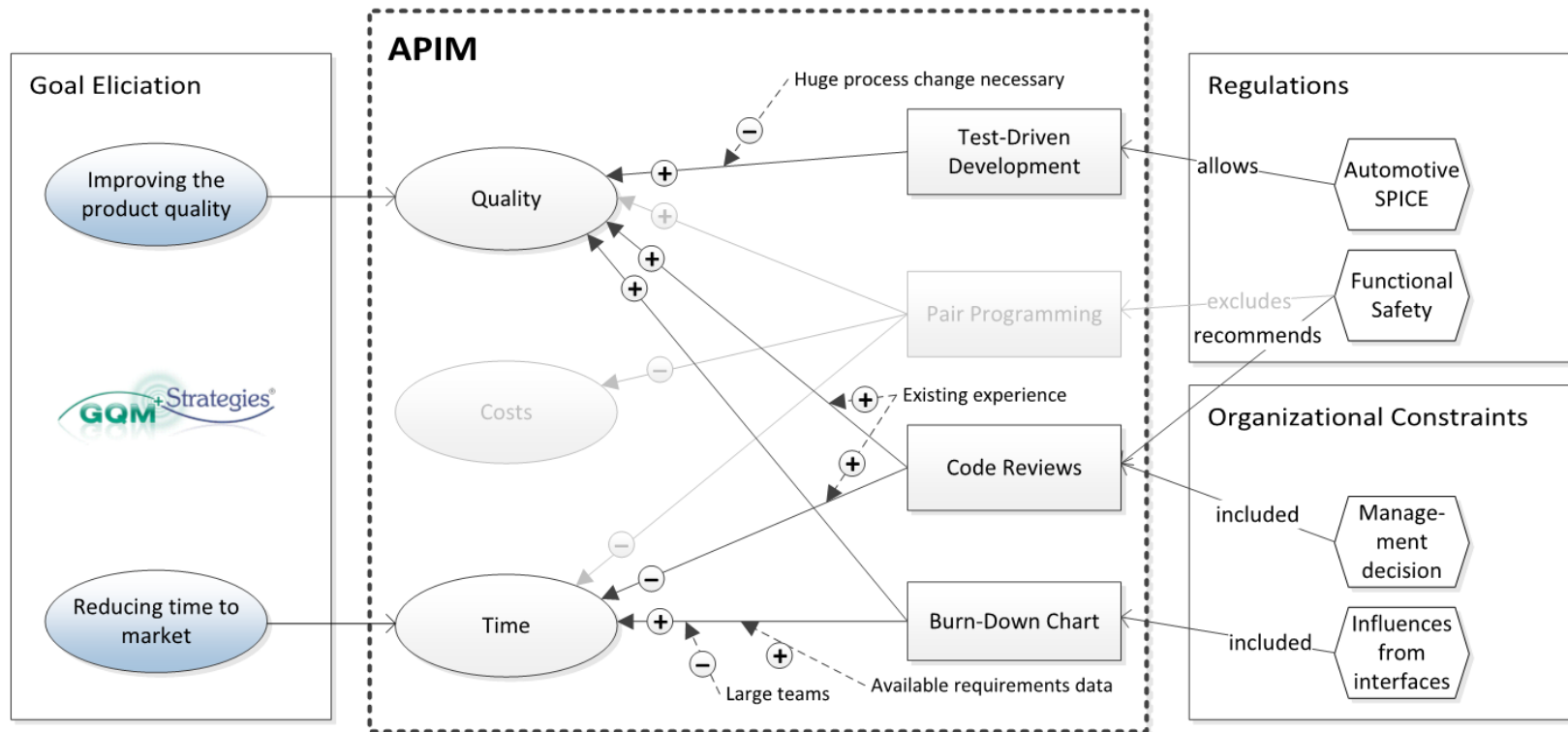


# Passende agile Praktiken finden

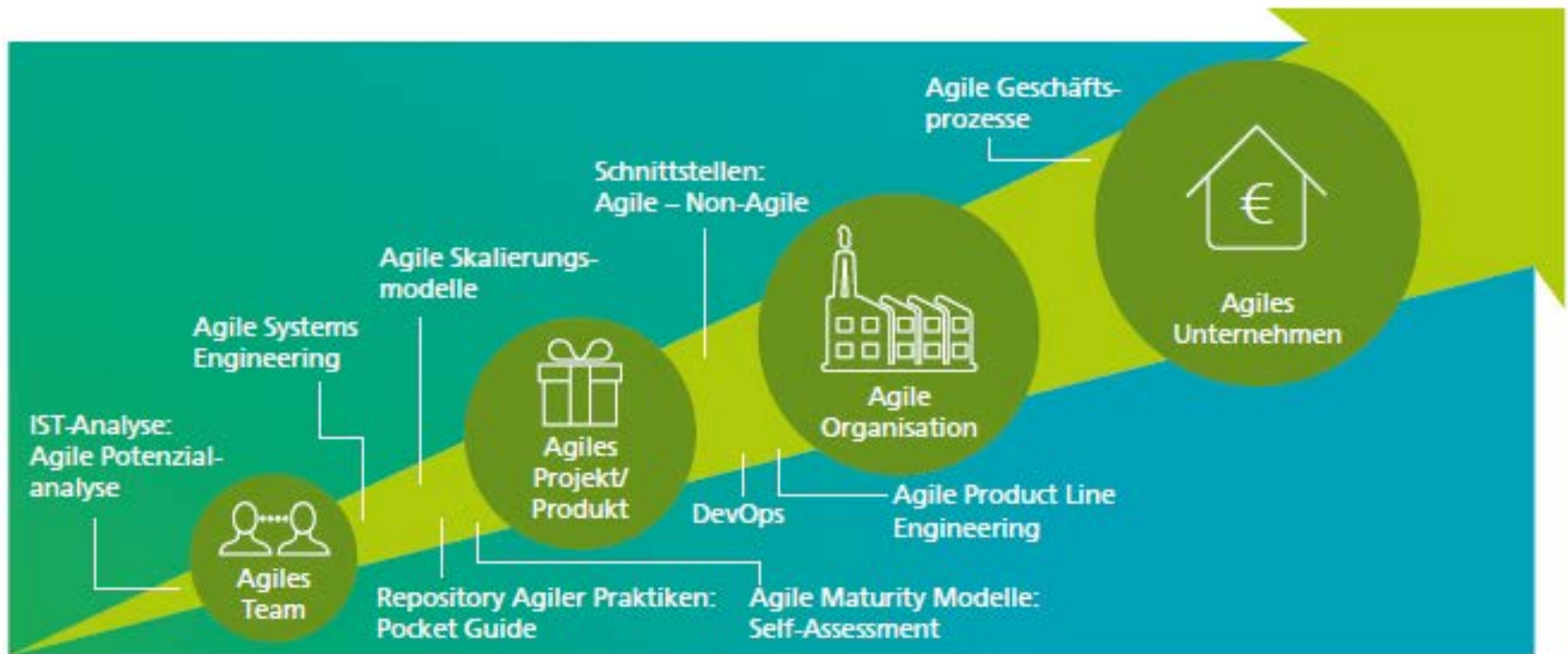




# AGILE POTENZIALANALYSE



# Agile Systems Engineering/ Agile im regulierten Umfeld



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

# Studienergebnisse: Einführung agiler Entwicklung

A stepwise (evolutionary, iterative) approach is favored when introducing Agile

## Some reasons

„change of complete process is **utopian**“,  
„**fear of changing** their current process“

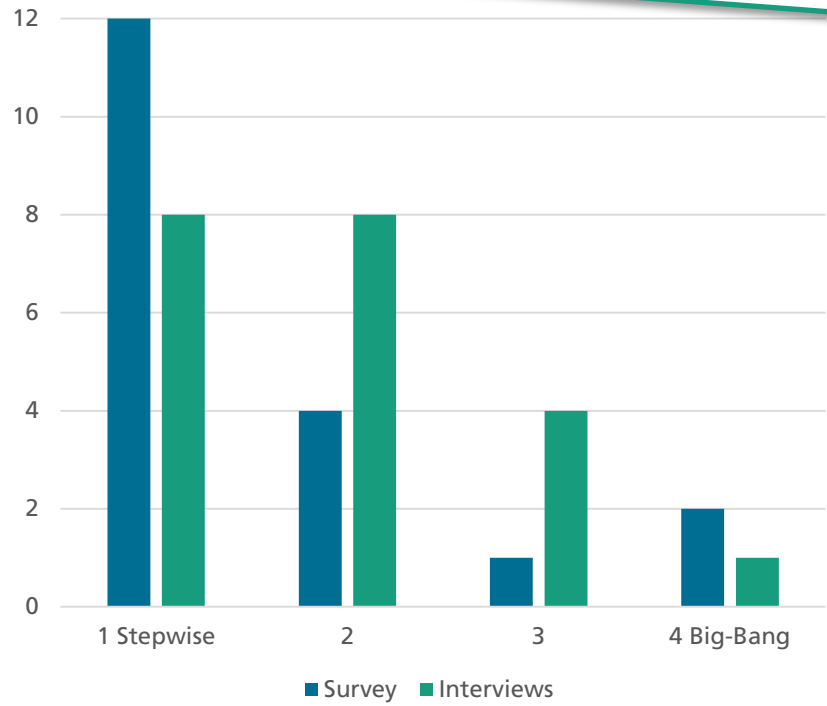
„big bang only in **emergency** situations“

„**change** is only possible **slowly**“

„**process problems** after big-bang, e.g. with regulations“

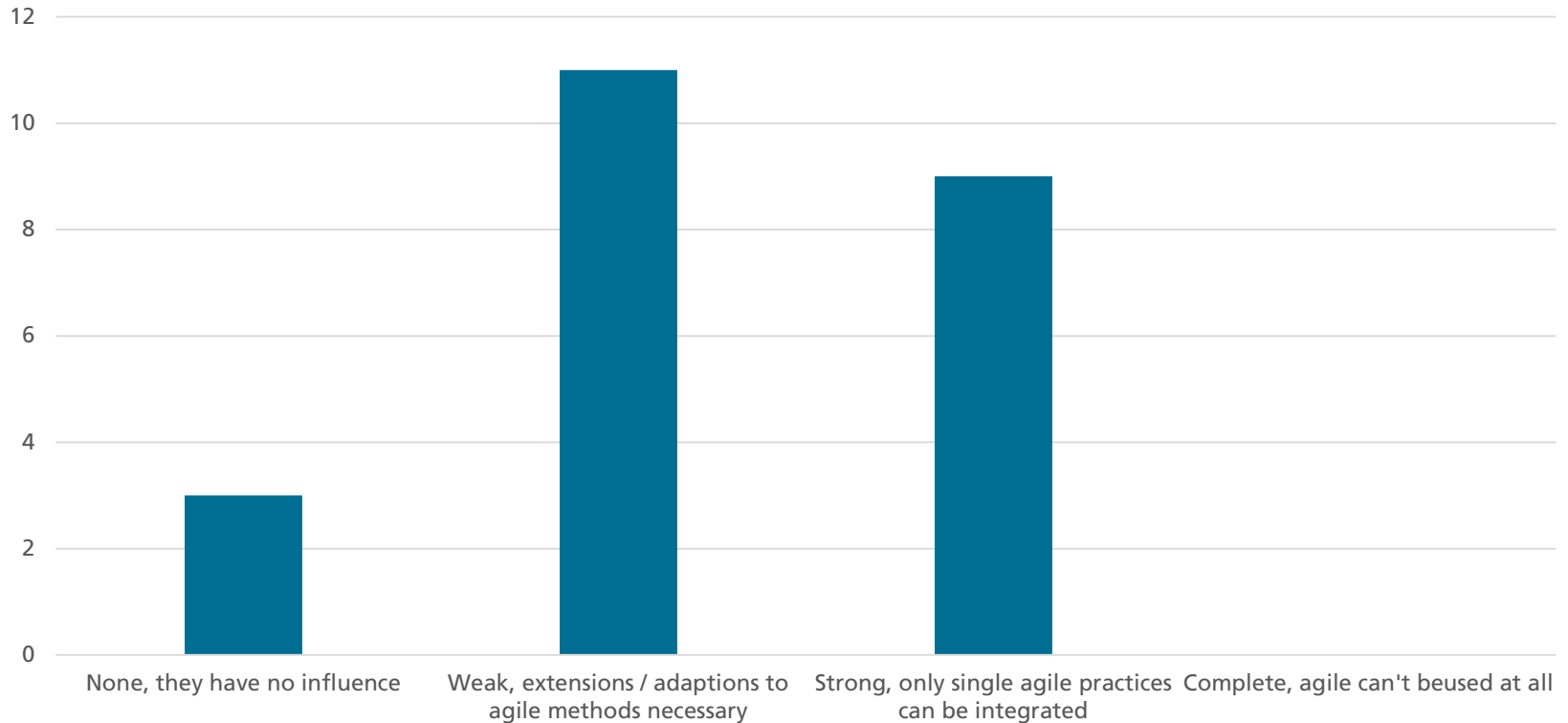
„Never seen a company where big-bang worked“

„big bang more **risky**“,  
„step-wise more **acceptance**“

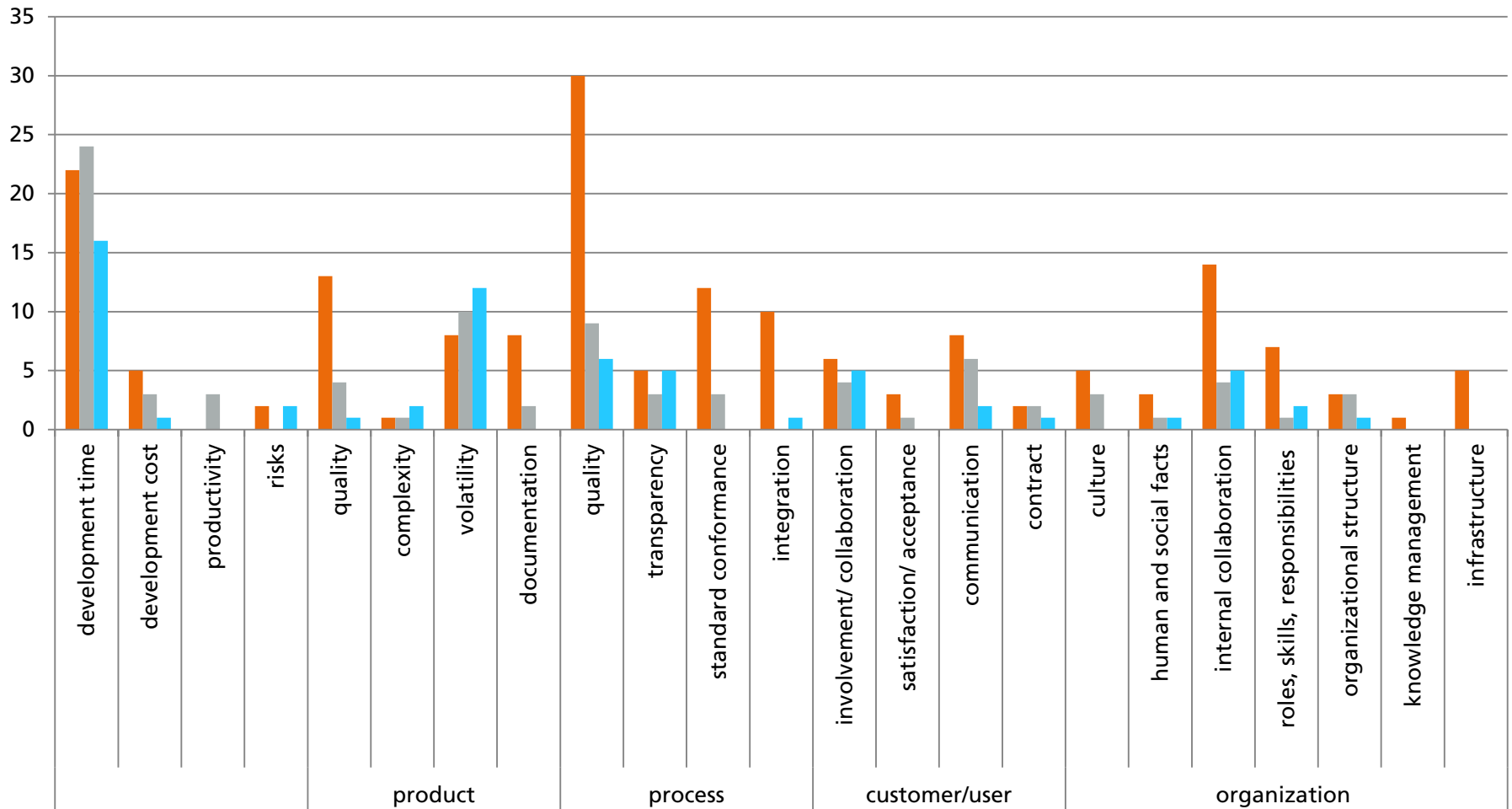


Diebold P, Theobald S. How is agile development currently being used in regulated embedded domains?. JSEP. 2018;e1935.  
<https://doi.org/10.1002/smr.1935>

# Studienergebnisse: Einfluss von Regularien auf Anwendung von Agil

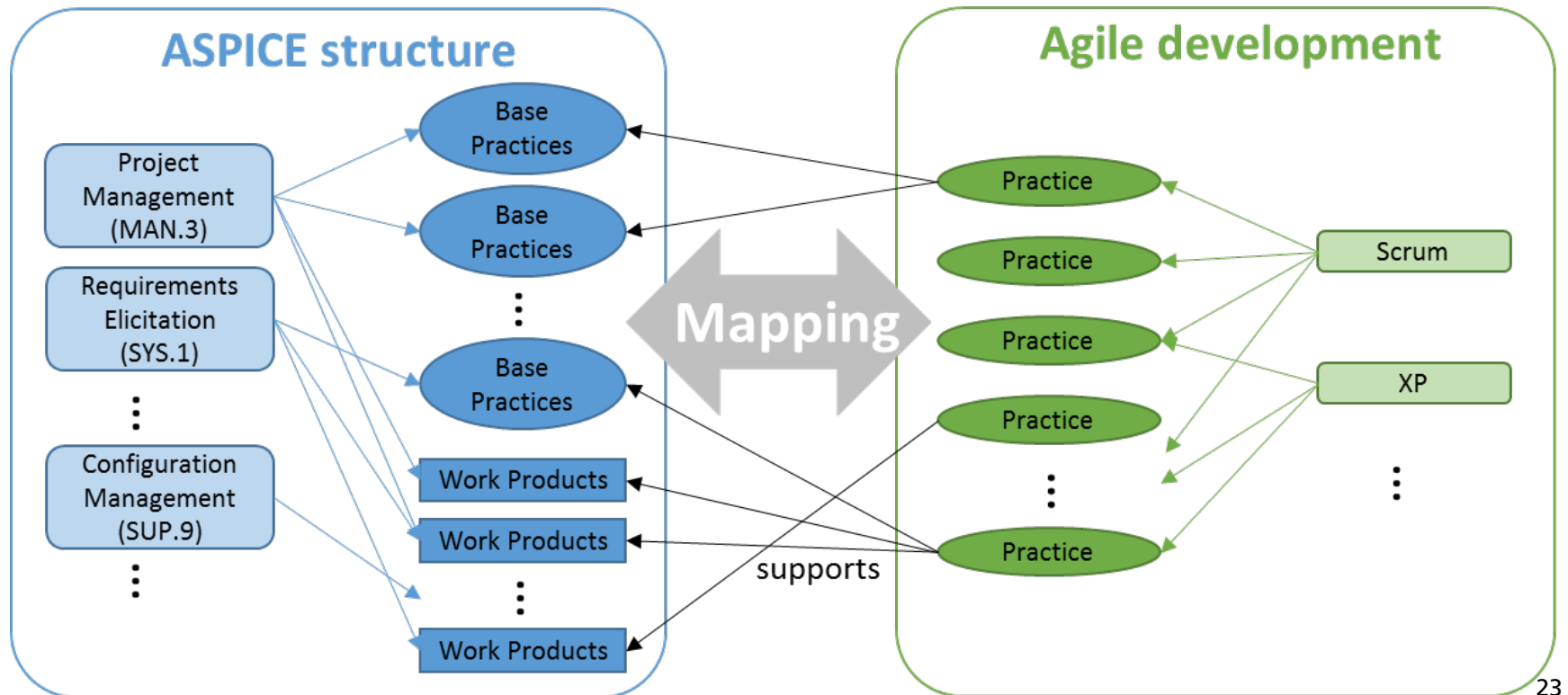


# Studienergebnisse: Vergleich: Probleme, Gründe für die Einführung von Agil, erwarteter Nutzen



# Automotive SPICE und Agile

- Struktur von Automotive SPICE und Agiler Entwicklung und Detail des Mappings:



23

# Support von einzelnen Base Practices durch agile Praktiken

	ACQ.4	SYS.1	SYS.2	SYS.3	SYS.4	SYS.5	SWE.1	SWE.2	SWE.3	SWE.4	SWE.5	SWE.6	SUP.1	SUP.8	SUP.9	SUP.10	MAN.3
BP1																	
BP2																	
BP3																	
BP4																	
BP5																	
BP6																	
BP7																	
BP8																	
BP9																	
BP10																	
#Mappings	27	44	39	23	26	25	46	30	40	38	27	21	39	17	33	33	104
Ø per BP	5,4	7,3	4,9	2,9	2,9	3,6	5,8	3,3	5,0	5,4	3,0	3,0	6,5	1,9	3,7	4,1	10,4
#APs	23	23	20	16	17	19	24	21	29	22	19	19	26	7	16	17	37

(Je dunkler der Blauton, desto höhere Anzahl an unterstützenden Praktiken)

# Beitrag von Scrum und XP Praktiken zur Erfüllung von ASPICE

Name	Agile Method	Processes (BPs)													Total	WPs	BP & WP				
		ACQ.4	SYS.1	SYS.2	SYS.3	SYS.4	SYS.5	SWE.1	SWE.2	SWE.3	SWE.4	SWE.5	SWE.6	SUP.1		SUP.8	SUP.9	SUP.10	MAN.3	Total	Total
Backlog	SCRUM																		47	14	61
Burn Chart	SCRUM																		9	2	11
Code and Test	XP																		4	2	6
Collective Ownership	XP																		7		7
Continuous Deployment	XP																		6	2	8
Continuous Integration	XP																		12	4	16
Daily Meeting	SCRUM																		12	5	17
Definition of Done	SCRUM																		17	8	25
Impediments	SCRUM																		11	8	19
Incremental Design	XP																		4	3	7
Iterative development	SCRUM																		1	4	5
Metaphor	XP																		3	1	4
Negotiated Scope	XP																		2		2
On-site customer	XP																		13	1	14
Open Workspace	XP																		9		9
Pair Programming	XP																		5	1	6
Pay-per-Use	XP																				
Planning Game	XP																		5		5
Planning meeting	SCRUM																		17	8	25
Product Owner	SCRUM																		28	2	30
Refactoring	XP																		2		2
Relative Estimation	SCRUM																		2		2
Retrospective	SCRUM																		8	2	10
Root Cause Analysis	XP																		2	2	4
Scrum Master	SCRUM																		13	1	14
Shippable Increment	SCRUM																		6	2	8
Shrinking Teams	XP																				
Simple Design	XP																		8	3	11
Single Code Base	XP																		3	1	4
Sitting Together / Common Worksp	XP																				
Slack	XP																		1		1
Small Release Cycles	XP																		2	1	3
Sustainable Pace	XP																			1	1
Team	SCRUM, XP																		4	2	6
Team Continuity	XP																				
Templates and Standards	XP																		17	3	20
Test Driven Development (TDD)	XP																		11	3	14
User stories	XP																		20	5	25

(Je dunkler der Blauton, desto höhere Anzahl an unterstützenden Praktiken)



# Detailliertes Mapping Agilen Praktiken auf Base Practices am Beispiel MAN.3

Base Practice / Work product			Agile Mapping				
ASPICE3.0 ID	ASPICE2.5 ID	NPLF Name	ID	Name	Contribute	Remark	Source
MAN.3.BP1 MAN.3.BP1 Define the scope of work			31	Backlog	+	If the (product) backlog is us	[3]; [8]
			32	Backlog grooming	+		[1]
			72	Product Owner	+	Responsible for continuous r	[8], [2]
			154	Negotiated Scope	+ +		DR
			171	Planning Game	+	[11]	
			184	Project Chartering	+	[8]; [10]	
			239	Planning meeting	+	[8]	
			251	Metaphor	+ +	PD	
MAN.3.BP2 MAN.3.BP2 Define project life cycle			55	Continuous Deployment	+		TZ
			56	Continuous Integration	+		PD
			121	Incremental Design	+ +		TZ
			119	Incremental Development	+ +		PD
			138	Iterative development	+ +		TZ
			210	Small Release Cycles	+ +		PD
			213	Scrum Master	+	for controlling the life-cycle	DR
			215	Scrum of Scrums	+		[10]
			239	Planning meeting	+	In Scrum the phases are a re	[2]
279	Timebox	+ +		[6]			
MAN.3.BP3 MAN.3.BP3 Evaluate feasibility of the project			31	Backlog	+	incl. estimations	[10]
			171	Planning Game	+		[11]
			172	Planning Poker	+		[11], [3]
			198	Relative Estimation	+	Initial estimation should al:	[9]; [3]; [10]
			239	Planning meeting	+	Estimation during each sprii	[6]
			256	Team	+		PD

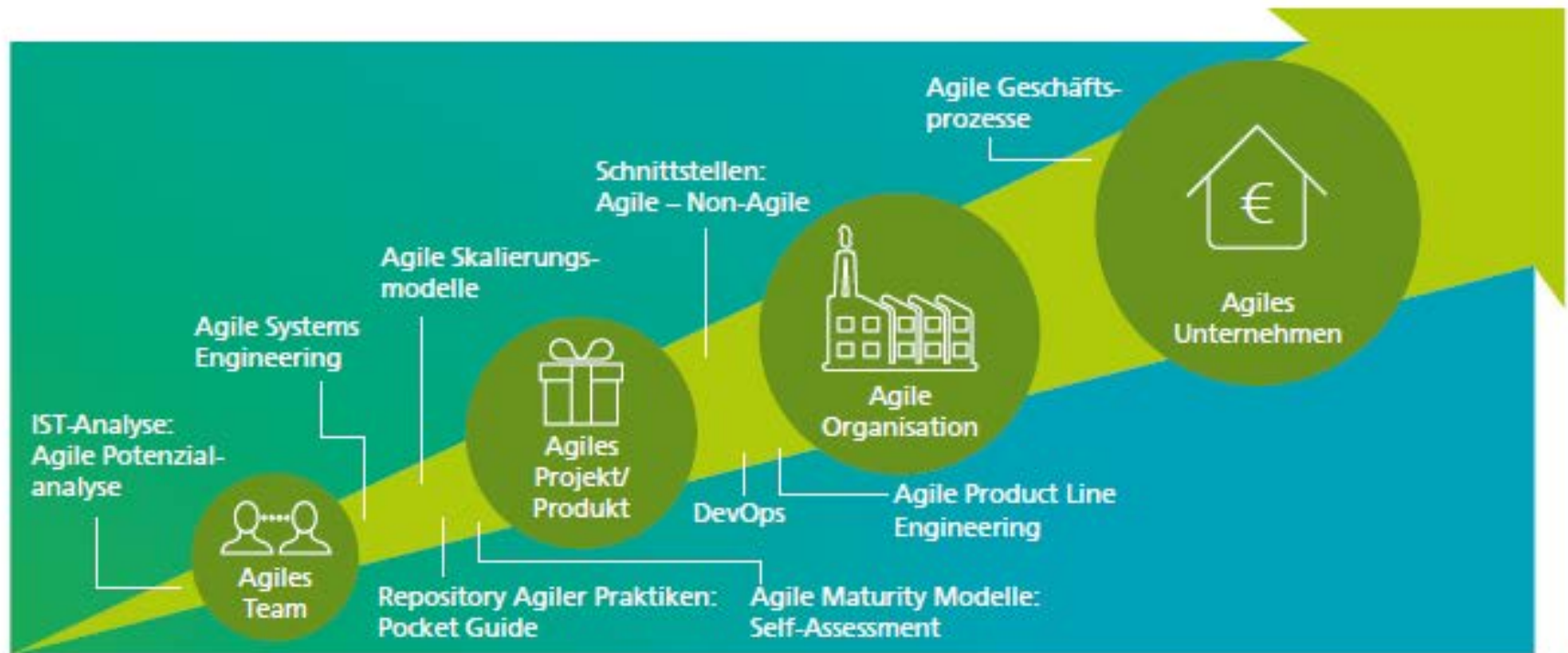
ASPICE Process

Grad der Unterstützung

ASPICE Base Practice

Zugeordnete agile Praktik

# Agile Skalierungsansätze



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

---

# UNTERSUCHTE AGILE SKALIERUNGSFRAMEWORKS

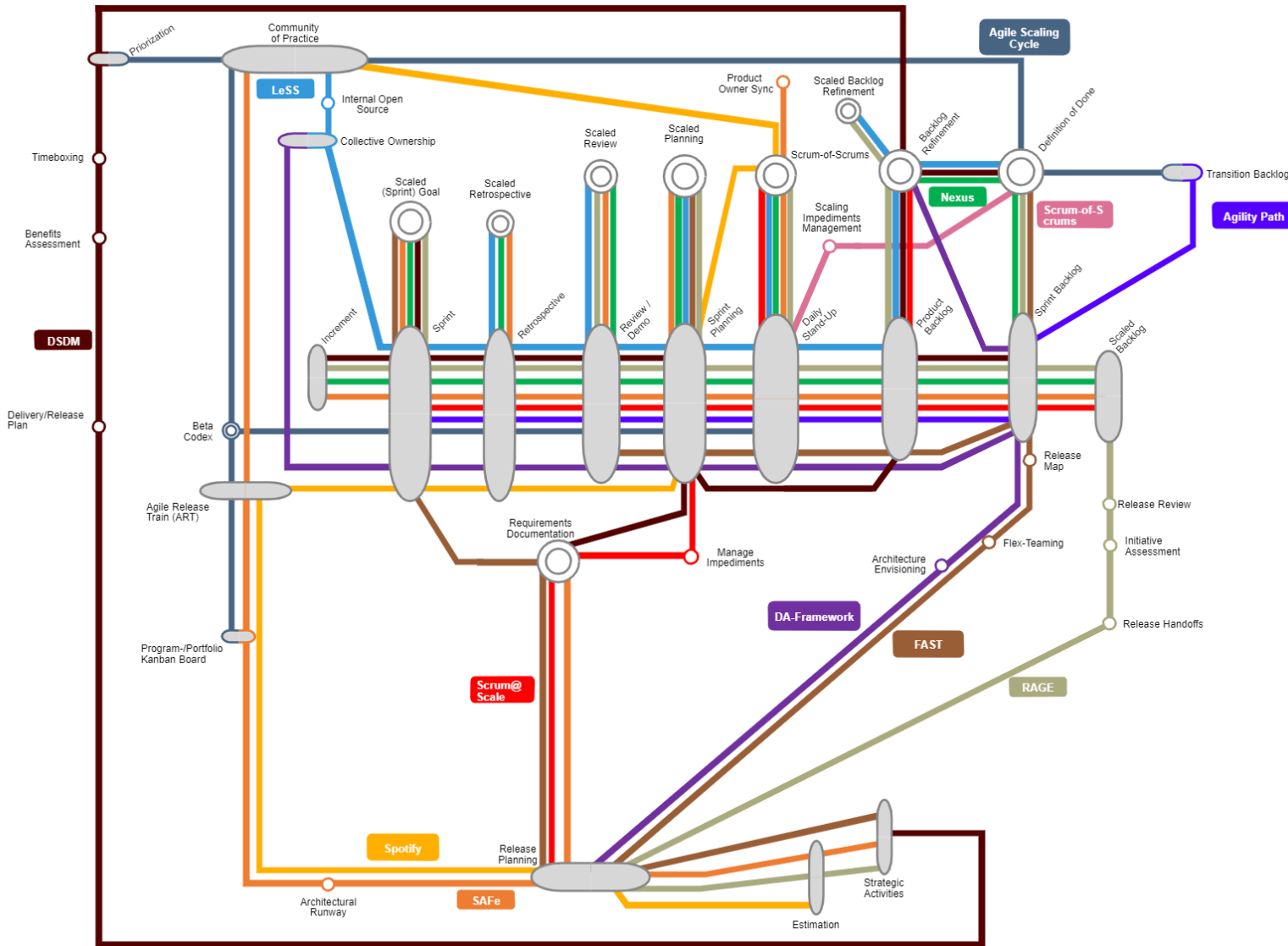
---

- Dynamic Systems Development Method (DSDM), 1994
- Scrum-of-Scrums, 1996
- Disciplined Agile (DA) Framework, 2009
- Spotify, 2012
- Recipes for Agile Governance (RAGE), 2013
- Agility Path, 2013
- Large Scaled Scrum (LeSS), 2014
- Scrum@Scale, 2014
- Agile Scaling Cycle, 2014
- Nexus / Scaled Professional Scrum, 2015
- FAST Agile Scaled Technology (FAST), 2015
- Scaled Agile Framework (SAFe), 2015

# Kriterien zum Vergleich von Skalierungsframeworks

Cat.	Comparison Criterion	Answer Options	Change
<b>Method Overview</b>	Description	Text	-
	Appearance year	Text	New
	Popularity (General) (fame & frequency of usage)	Low, Med., High	-
	Underlying Agile Method (Scrum, Kanban, XP, etc.)	Text	-
<b>Method Applicability</b>	Completeness of coverage of "levels":		Answer Removed
	Portfolio	Low, Med., High	-
	Program Structure	Low, Med., High	-
	Inter-team Coordination	Low, Med., High	-
	Team Level	Low, Med., High	-
	Scale / Target size	Small, Med., Large	-
	Usage in Domain (Automotive)	Low, Med., High	New
	Control (central vs. distributed)	Central, Mixed, Distributed	Changed from free text
<b>Distinguishing Features</b>	Degree of Flexibility (prescriptive vs. emergent)	Low, Med., High	-
	Focus	Text	-
	Agile thinking / Culture	No/Yes	New
	Team / Structure	No/Yes	New
	Technical Practices	Low, Med., High	-
	Big Positives / Key Differentiators	Text	-
<b>Available Information/ Support</b>	Key Risks / Concerns	Text	-
	Availability of Details & Support	Low, Med., High	-
	(certified) Coaches	Text	New
	Certification	No/Yes	New
	Trainings / Coaching	No/Yes	Changed from free text
	Coaching Effort	Low, Med., High	New
Web Link	Text / URL	-	

# Subway Map zum Vergleich von Skalierungsframeworks



# Praktiken aus der Gruppe „Team Level“

Category	Subcategory	Practices
Meeting Types	<i>Daily Stand-Up</i>	Daily Scrum, Daily Stand-Up, Weekly Scrum, Stan Up Meeting, Daily Coordination Meeting
Planning Meeting	<i>Sprint Planning</i>	Iteration Planning, Sprint Planning Part 1, Sprint Planning and Investigation, Phase Planning, Sprint Planning, Planning Session, FAST Meeting - Part 2, Marketplace in Open Space style, Kick-Off
	<i>Iterations Modeling</i>	Iterations Modeling
Backlog Preparation	<i>Product Backlog</i>	Backlog, Product Backlog, Tribe Product Backlog, Team Backlog
	<i>Sprint Backlog</i>	Sprint Backlog, Iteration Backlog,
	<i>Backlog Refinement</i>	Backlog Grooming, Product Backlog Refinement, Backlog Decomposition, Backlog Prioritization, PE Inspection (in Sprint), Look-ahead Planning,
Iterative Procedure	<i>Sprint</i>	Sprint, Synchronous Sprints, Iteration
	<i>Cadence</i>	Cadence, Iteration Cadence, Program Increment Cadence, Release Cadence
<i>Modelling</i>		Look-Ahead Modeling, Model Storming, (Agile) Modelling
Architecture	<i>Solution Architecture Definition</i>	Solution Architecture Definition
<i>Non-Solo Development</i>		Non-Solo Development, Mob-Programming
Continuous Acts	<i>Continuous Integration</i>	Continuous Delivery, Continuous Deployment, Continuous Integration
Test Activities	<i>Acceptance TDD</i>	Acceptance TDD (ATDD)
	<i>Test-Driven Development (TDD)</i>	Test-Driven Development (TDD)
	<i>Test-First</i>	Test-First
	<i>Parallel Independent Testing</i>	Parallel Independent Testing
	<i>Developer Regression Testing</i>	Developer Regression Testing
	<i>Sandboxing</i>	Sandboxing

Release Activities	<i>Product Deploy Validation</i>	Product Deploy Validation
	<i>Deployment</i>	Deployment, Sprint Consolidation
Lessons Learned	<i>Retrospective</i>	Retrospective, Sprint Retrospective, Iteration Retrospective, Team Retrospective
	<i>Lessons Learned</i>	Lessons Learned
Review / Demo	<i>Review / Demo</i>	Sprint Review, Sprint Review Record, Iteration Review, Production Readiness Review, Light-Weight Milestone Review, FAST-Meeting - Part 1: Review (show and tell), Project Review
	<i>Quality Review Records</i>	Business Quality Review Records, Technical Quality Review Records
Progress Activities	<i>Task Management</i>	Breakdown Chart, Task Board, Task Breakdown, Tasks
	<i>Work-In-Progress (WIP)</i>	Work-In-Progress (WIP) Limit
	<i>Progress Monitoring (e.g., using Burn Charts)</i>	Burn-Up-Chart, Burn-Down-Chart, Cumulative Flow Chart, Monitoring, Progress Monitoring
	<i>Definition of Done</i>	Definition of Done
Refactoring	<i>Definition of Ready</i>	Definition of Ready, Deploy Ready
	<i>Refactoring</i>	Refactoring
Others		<i>Feasibility Investigation, Facilitated Workshop, Automated Metrics</i>

# Praktiken aus der Gruppe „Scaling Agile“

Categories	Subcategories	Practices
Meeting Types	<i>Scrum-of-Scrums</i>	Scrum-of-Scrums-Meeting, Scrum-of-Scrums, Nexus Daily Scrum, Cross-Team Coordination, Inter-Team Coordination Meeting
	<i>Product Owner Sync</i>	Product Owner Sync
Planning Meeting	<i>Scaled Planning</i>	Program Increment Planning, Sprint Planning Part 2, Nexus Sprint Planning, Portfolio Planning Meeting, Multisite Sprint Planning Part 1
	<i>Scaled (Sprint) Goal</i>	FAST Meeting - Part III: Announcements and Alignment of Vision, Nexus Sprint Goal, Program Increment Objective, Terms of Reference, Agile Charter
Backlog Preparation	<i>Scaled Backlog</i>	Program Backlog, Sync Backlog, Portfolio Backlog, Nexus Sprint Backlog
	<i>Scaled Backlog Refinement</i>	Joint Light Product Backlog Refinement, Multisite Product Backlog Refinement, Portfolio Grooming Meeting
Manage Impediments	<i>Scaling Impediments Management</i>	Impediments (Backlog)
Delivery	<i>Agile Release Train</i>	Agile Release Train, Release Train
Architecture	<i>Architectural Runway</i>	Architectural Runway
Open Source Data	<i>Collective Ownership</i>	Collective Ownership
Release Activities	<i>Release Planning</i>	Release Planning, Release Management, Release Planning Meeting
	<i>Release Handoffs</i>	Release Handoffs
	<i>Release Review</i>	Release Review
Lessons Learned	<i>Scaled Retrospective</i>	Joint Retrospective, Nexus Sprint Retrospective, Inspect & Adapt Workshop
Review / Demo	<i>Review / Demo</i>	Quality Assessment
	<i>Scaled Review</i>	Overall Sprint Review, Multisite Sprint Review, Staging Readiness Review, Nexus Sprint Review, System Demo
Progress Activities	<i>Portfolio/Program Kanban Board</i>	Portfolio Kanban, Program Kanban
Others		<i>Initiative Assessment, Flex-Teaming, Beta Codex, Automated Metrics</i>

# Praktiken aus der Gruppe „Team Level and Scaling Agile“

Categories	Subcategories	Practices
Meeting Types	<i>Timeboxing</i>	Timeboxing
Planning Meeting	<i>Prioritization</i>	Prioritization Meeting, MoSCoW, Prioritized Requirements List
Backlog Preparation	<i>Transition Backlog</i>	Evaluation Backlog, Transition Backlog, Practice Backlog
	<i>Release Map</i>	Release Map
Manage Impediments	<i>Manage Impediments</i>	Impediment Removal, Impediment Backlog, Continuous Impediment Removal
<i>Requirements Documentation</i>		User Stories, Portfolio Epic, Epic, Story Document, Requirement Document
<i>Community of Practice</i>		Community of Practice
Iterative Procedure	<i>Increment</i>	Increment of Change, Integrated Increment, Evolutionary Development, Pre-and Post Program Increment
Architecture	<i>Architecture Envisioning</i>	Architecture Envisioning
Open Data Source	<i>Internal Open Source</i>	Internal Open Source
Release Activities	<i>Delivery/Release Plan</i>	Delivery/Release Plan
<i>Strategic Activities</i>		Decision making Framework, Lean-Agile Budgeting, Value Stream, Roadmap, Strategic Themes, Business Case, Decision Matrix, Funding Decision, Project Map
<i>Estimation</i>		Estimation, Forecasting
Others		<i>Benefits Assessment</i>



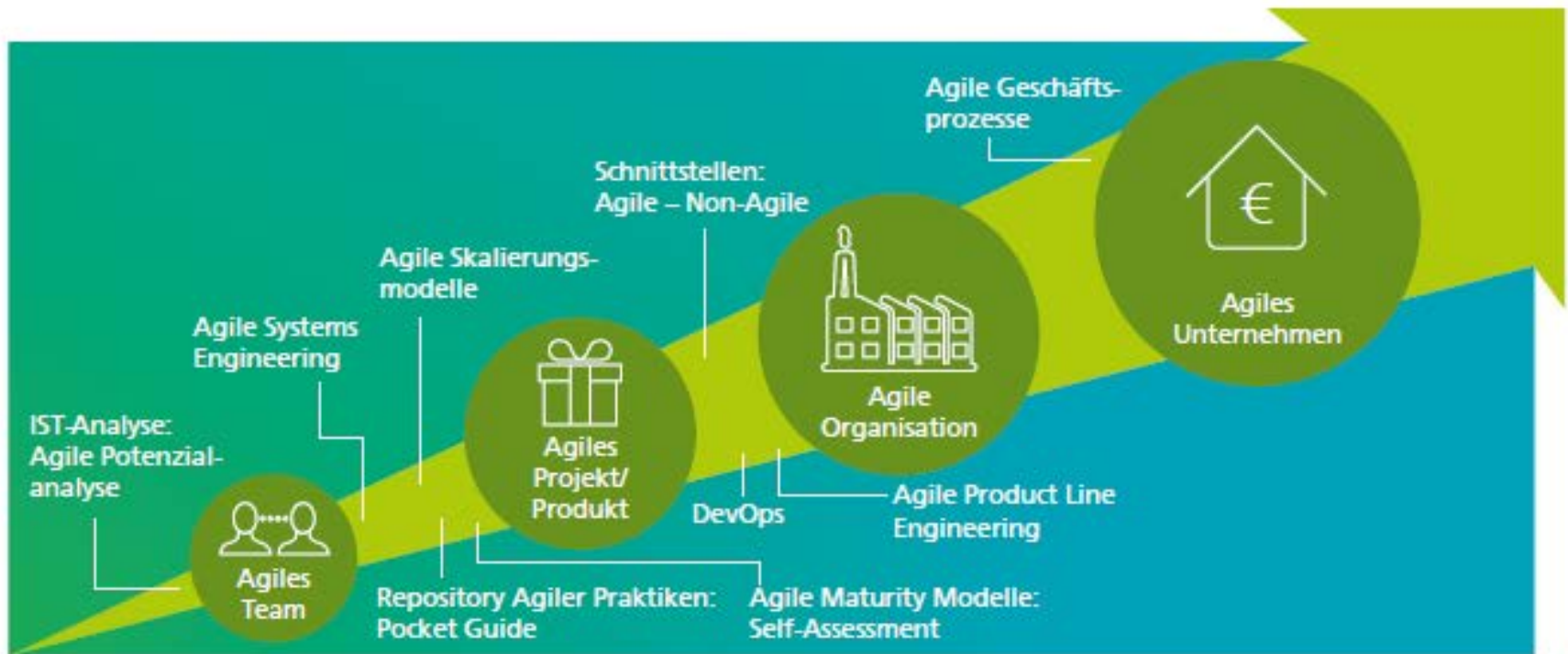
# Häufigkeit von Scrum und Skalierungspraktiken über die ausgewählten Skalierungsframeworks

#	Practice	#	Practice	#	Practice
11	Sprint Planning	4	Increment	1	Manage Impediments
11	Sprint	4	Scaled Review	1	Scaling Impediments Management
10	Retrospective	4	Strategic Activities	1	Architectural Runway
10	Review / "Demo"	4	Estimation	1	Architecture Envisioning
9	Daily Stand-Up	3	Agile Release Train	1	Internal Open Source
8	Product Backlog	3	Release Planning	1	Delivery/Release Plan
7	Definition of Done	3	Scaled Retrospective	1	Release Handoffs
6	Scrum of Scrums	2	Prioritization	1	Product Deploy Validation
6	Sprint Backlog	2	Transition Backlog	1	Release Review
6	Backlog Refinement	2	Scaled Backlog Refinement	1	Beta Codex
5	Scaled Planning	2	Collective Ownership	1	Facilitated Workshop
5	Scaled (Sprint) Goal	2	Portfolio/Program Kanban Board	1	Flex-Teaming
5	Requirements Documentation	1	Product Owner Sync	1	Initiative Assessment
4	Scaled Backlog	1	Timeboxing	1	Benefits Assessment
4	Community of Practice	1	Release Map		

# Wie definiere ich meinen individuellen Ansatz?

- Framework finden, das am besten zu meinem Kontext passt
  - Nicht zwingend das aktuell gehypte/bekannte Framework
- Kein Blueprint, muss angepasst werden
- Praktiken identifizieren
  - Mit Hilfe des Vergleichs der Frameworks
    - Welche Praktiken werden oft verlangt (good practices)?
    - Welche eher individuellen Praktiken können mir helfen?

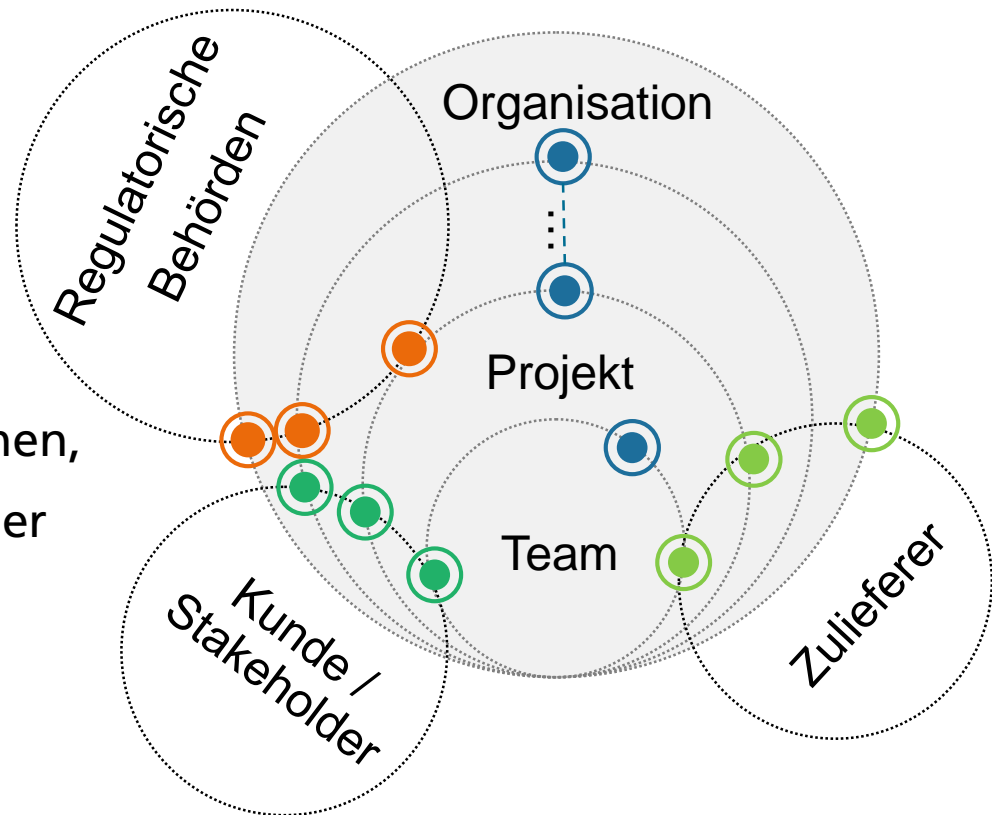
# Schnittstellen Agile/ Non-Agile



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

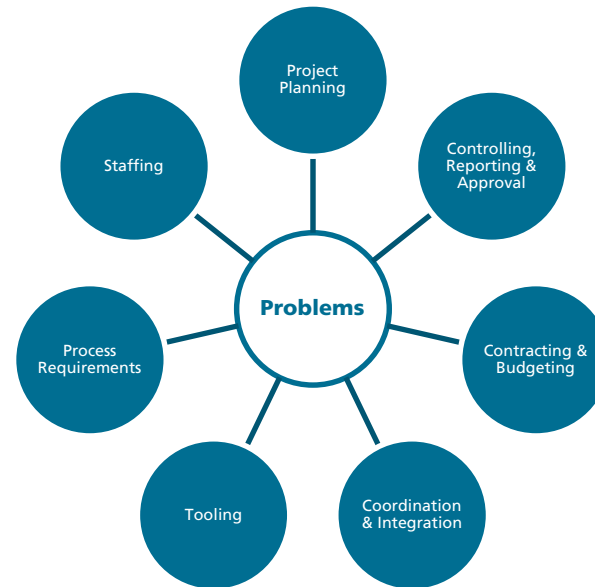
# Existierende Schnittstellen

- Interne 
  - Projekt-intern
  - Organisations-intern
- Externe   
  - Zu anderen Organisationen, Z.B. Kunde, Stakeholder oder Zulieferer
  - Zu anderen Projekten

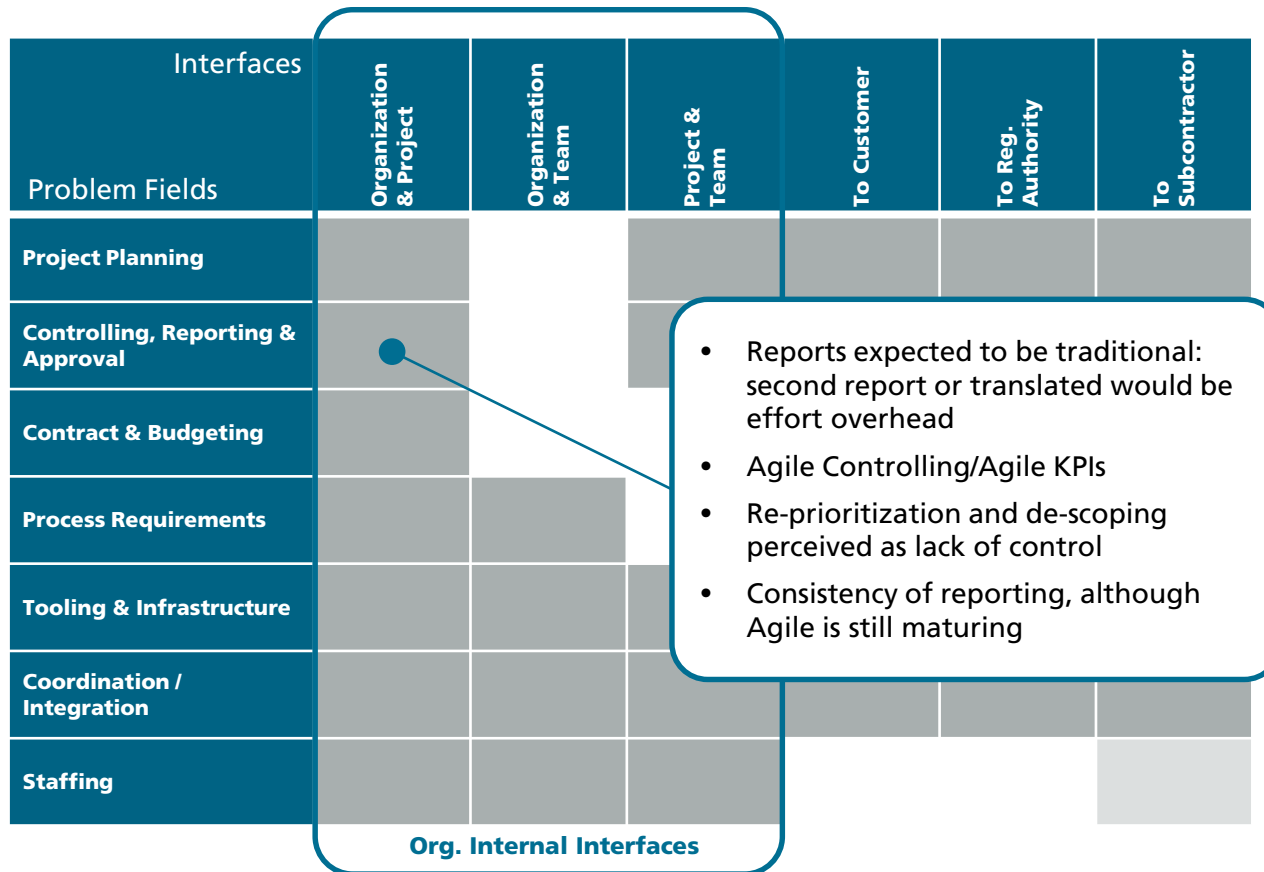


# Probleme, die an Schnittstellen auftreten können

- Project Planning
- Controlling, Reporting & Approval
- Contracting & Budgeting
- Coordination / Integration (content- & time wise)
- Tooling
- Process Requirements
- Staffing



# Beispielprobleme



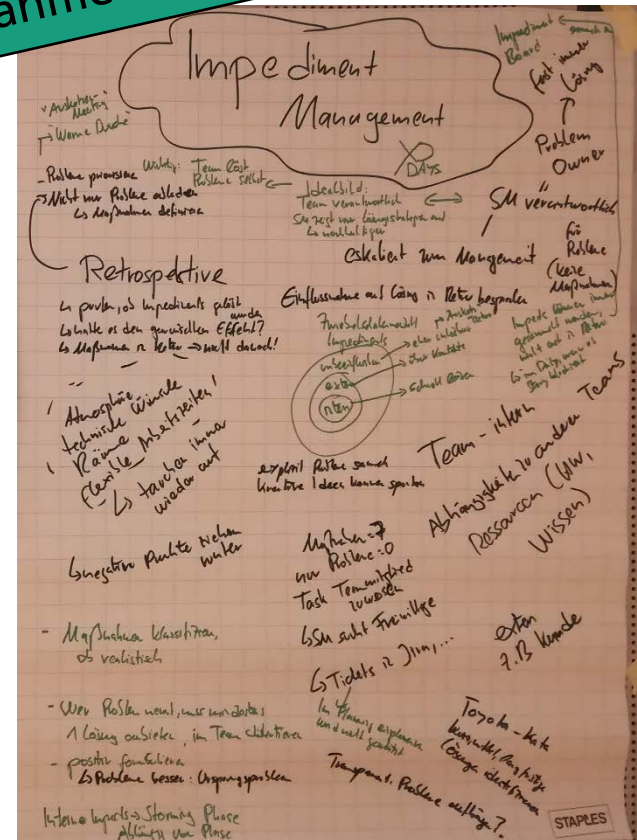
Sven Theobald, Philipp Diebold. (2018). Interface Problems of Agile in a Non-Agile Environment. 19th International Conference on Agile Software Development

# Studie über Impediments

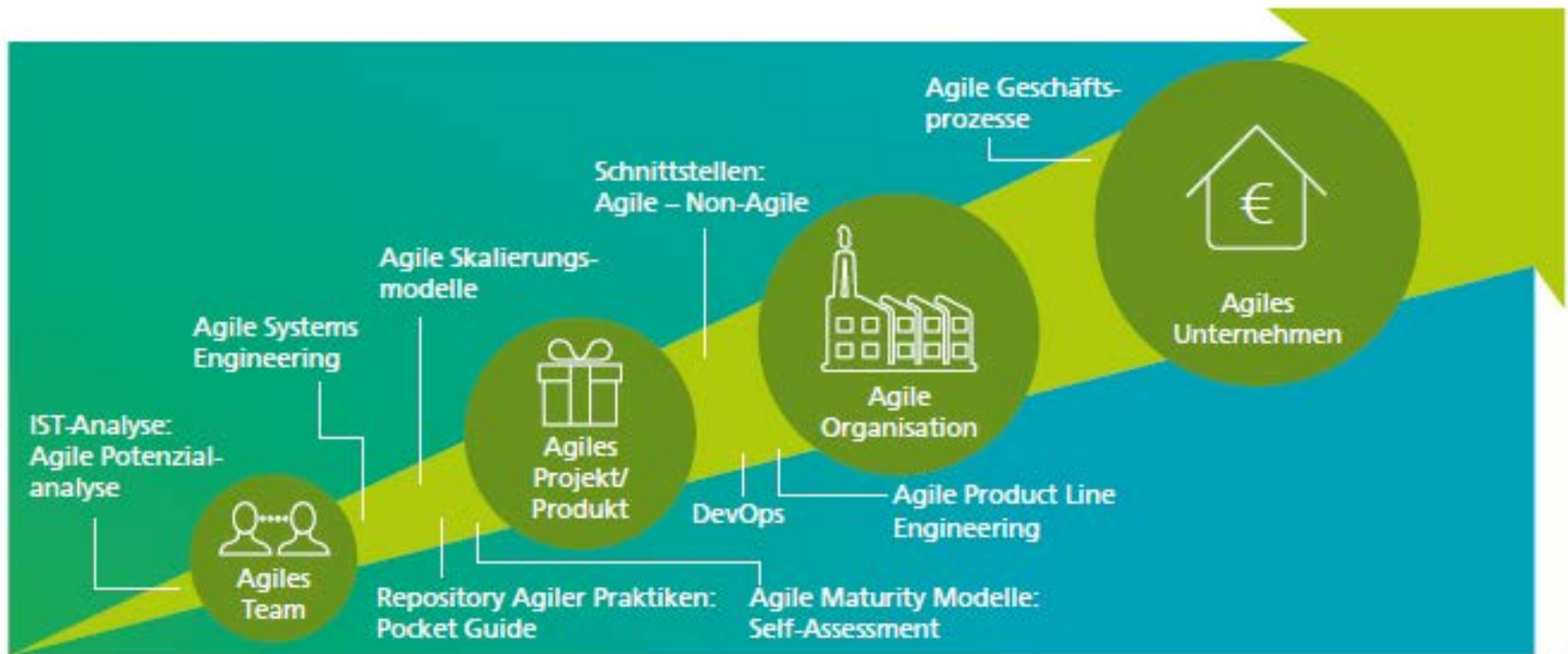
Kontaktieren Sie mich, für

- Interview
- Teilnahme an einer Online-Umfrage

- Wie werden Impediments
  - Identifiziert
  - Dokumentiert
  - Gelöst
- Welche (Klassen von) Impediments
  - Tauchen oft auf
  - Sind einfach oder schwer zu lösen
  - Beeinträchtigen den Prozess stark



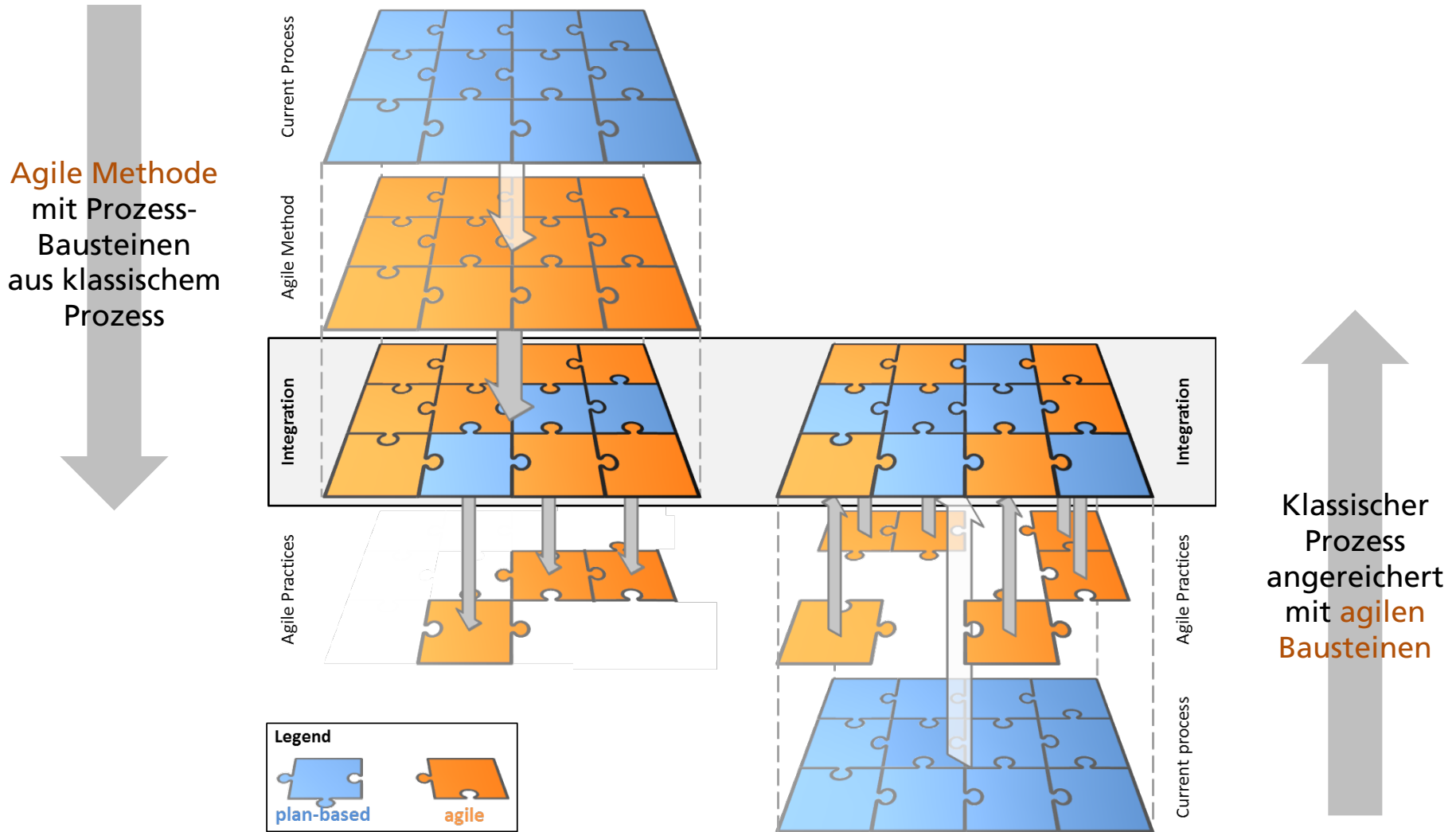
# Agile Transition und agile Organisation



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

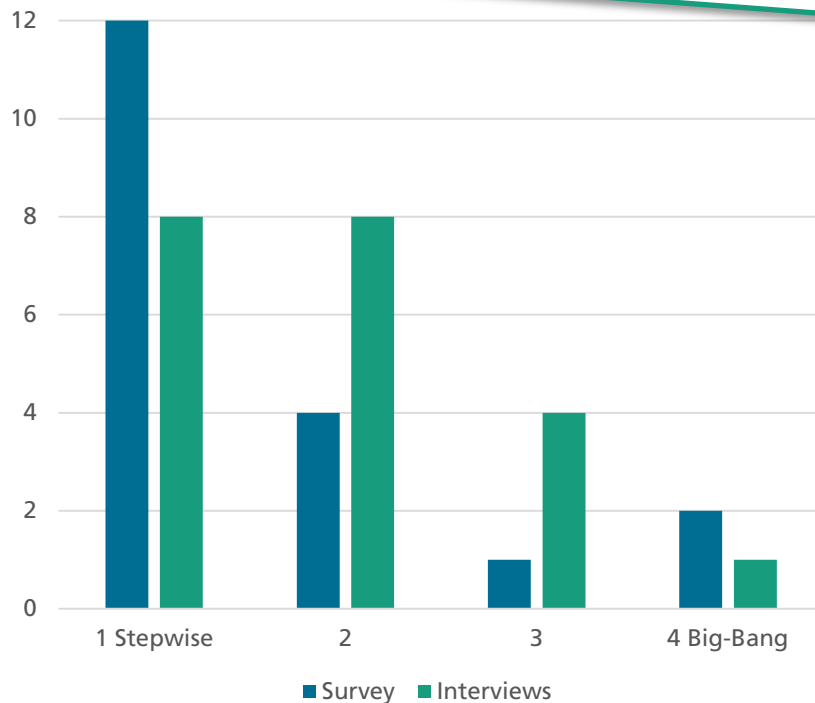


# Revolution vs. Evolution



# Studienergebnisse: Einführung agiler Entwicklung

A stepwise (evolutionary, iterative) approach is favored when introducing Agile



## Some reasons

„change of complete process is **utopian**“,  
„**fear of changing** their current process“

„big bang only in **emergency** situations“

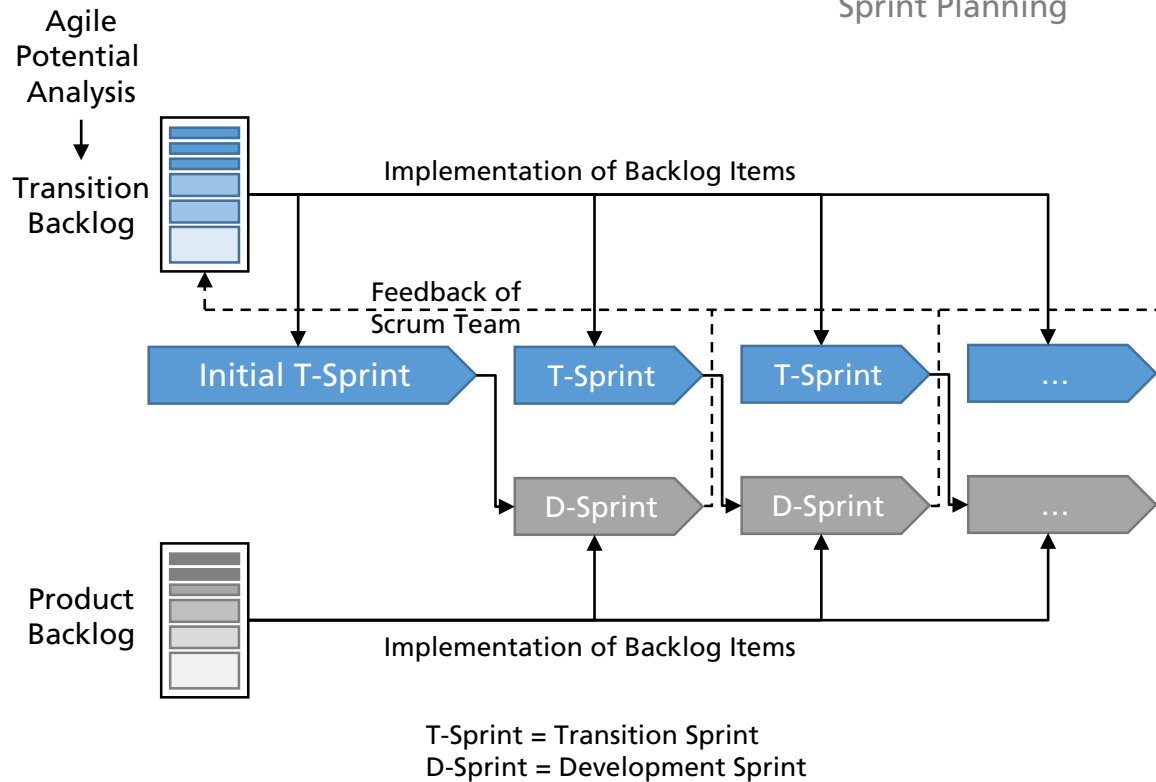
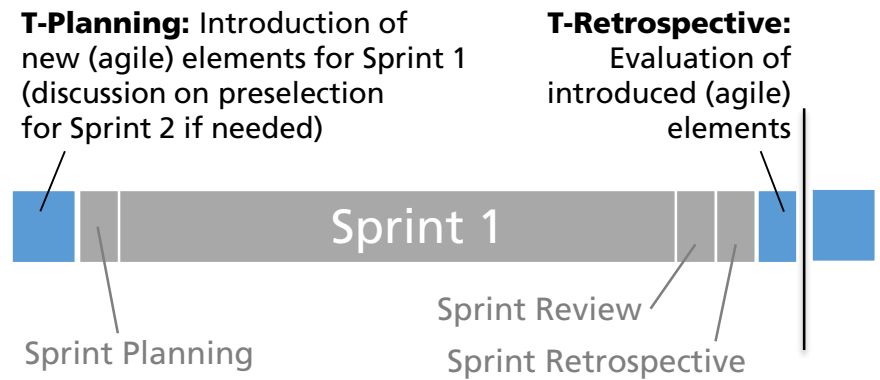
„**change** is only possible **slowly**“

„**process problems** after big-bang, e.g. with regulations“

„Never seen a company where big-bang worked“

„big bang more **risky**“,  
„step-wise more **acceptance**“

# Agile Transition



# Agile in Geschäftsprozessen

- Agile Praktiken sind nicht nur in Softwareentwicklung hilfreich
- Untersuchung von der Anwendung agiler Praktiken in Geschäftsprozessen
- Systematische Literaturreview zur Erhebung des State of the art
- Verbesserung von Geschäftsprozessen aus allen Bereichen eines Unternehmens möglich

# Studie agile Organisation

- Forschungsfragen
  - Was definiert eine agile Organisation?
  - Was ist der Status Quo von Unternehmen bezüglich ganzheitlicher Agilität?
  
- Systematische Literaturreview zum Thema „Agile Leadership“ und „agile Management“ abseits von Entwicklungsteams
  - 351 Paper gefunden
  - Nach Bewertung von Titel und Abstract: 40 Quellen
  
- Online Umfrage geplant

# Übersicht Agile Transition



Agile Transition: vom agilen Team bis hin zum agilen Unternehmen

# Kontakt und Feedback

**Sven Theobald**



Abteilung Process Engineering  
Fraunhofer IESE  
Fraunhofer-Platz 1,  
67663 Kaiserslautern, Germany  
[sven.theobald@iese.fraunhofer.de](mailto:sven.theobald@iese.fraunhofer.de)



