

Enterprise Architektur für Business Agility

Stefan Toth | st@embarc.de | [st_toth](https://twitter.com/st_toth) (twitter)

online
24. Juni 2021

embarc 
Software Consulting GmbH

Stefan Toth

✉ st@embarc.de

t @st_toth

X xing.to/sto

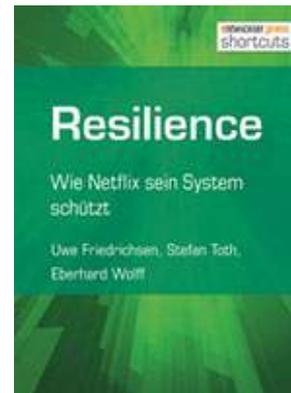


—
www.embarc.de
www.swamuster.de



embarc 
Software Consulting GmbH





Schnell-Einstieg

Business Agility



Agile ist Mainstream



COMPANY EXPERIENCE AND ADOPTION

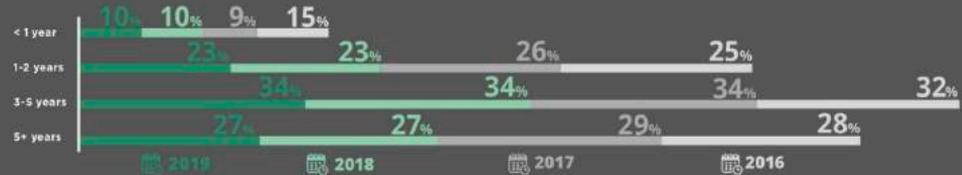
COMPANY EXPERIENCE

HOW MANY?
95% of respondents report their organizations practice Agile development methods.

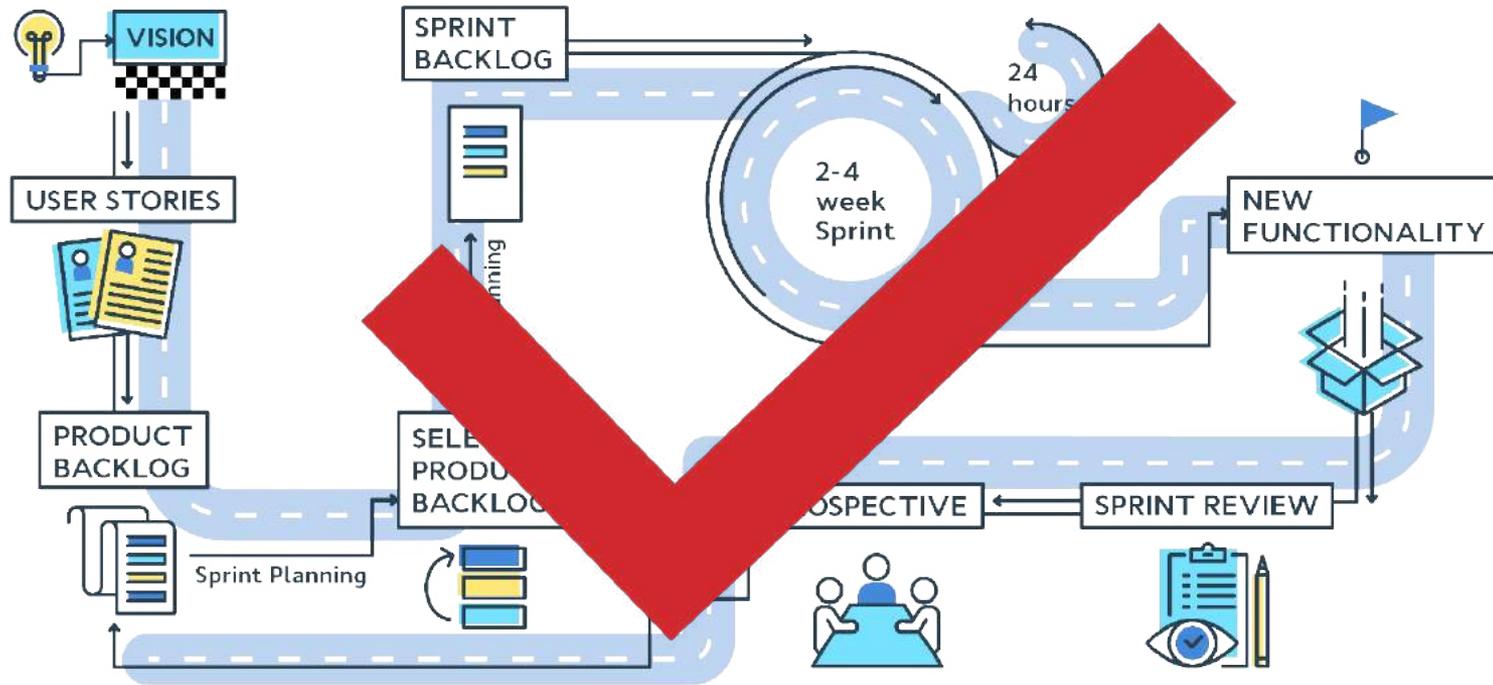


HOW LONG?

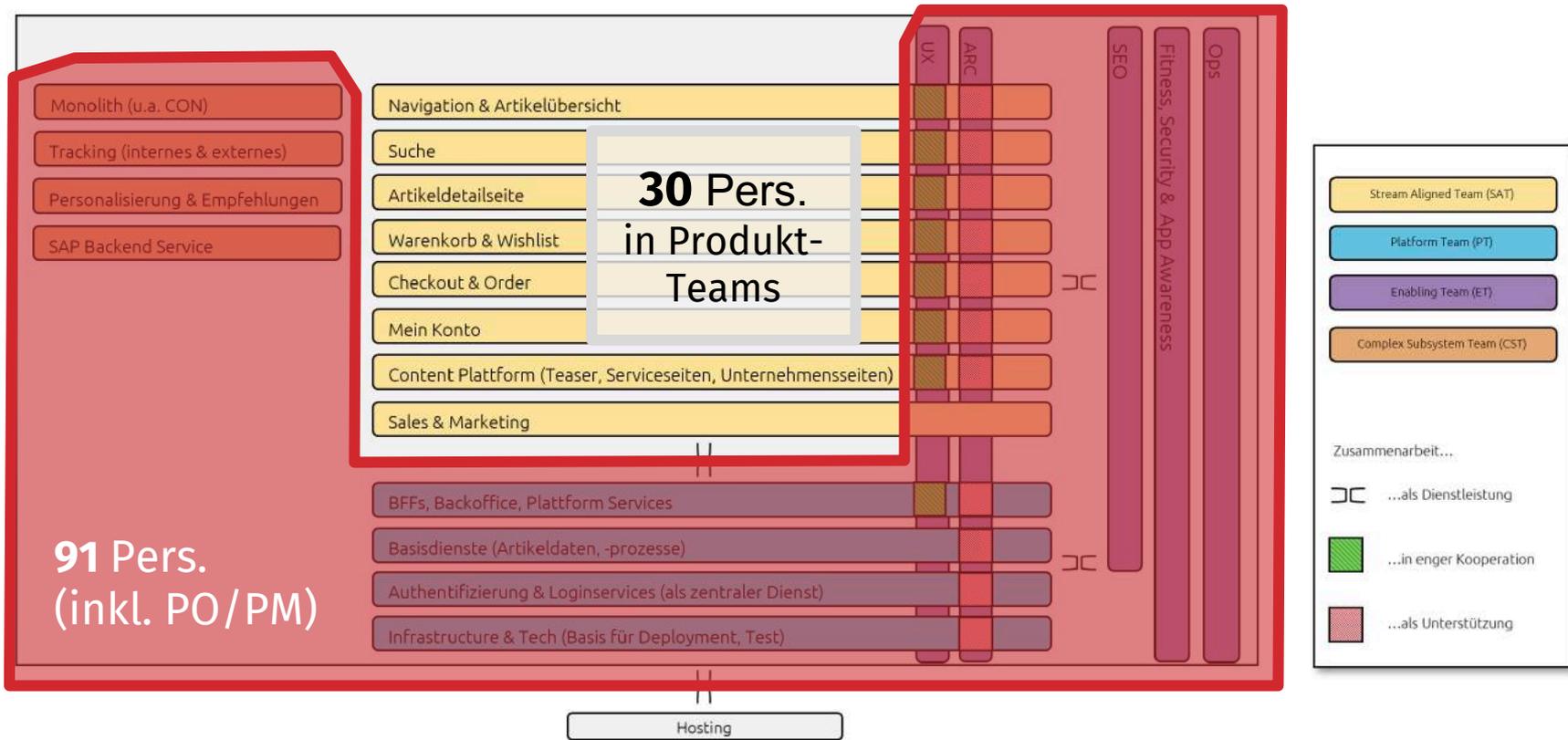
The length of time respondents' organizations have been practicing Agile development methods:



Scrum

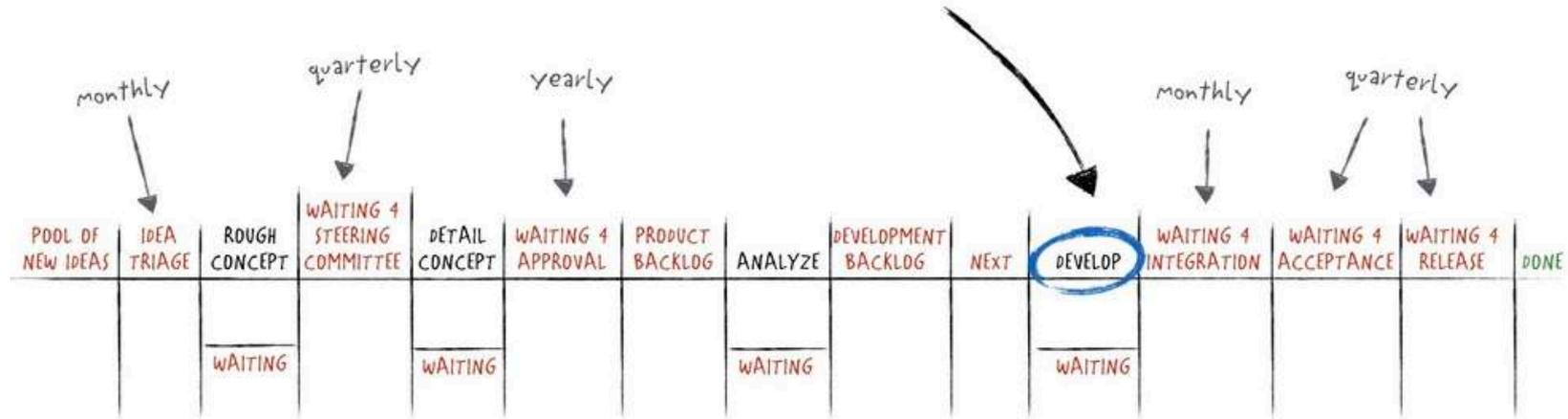


Webshop - Organisationsmodell 2021



Das drumherum mitdenken!

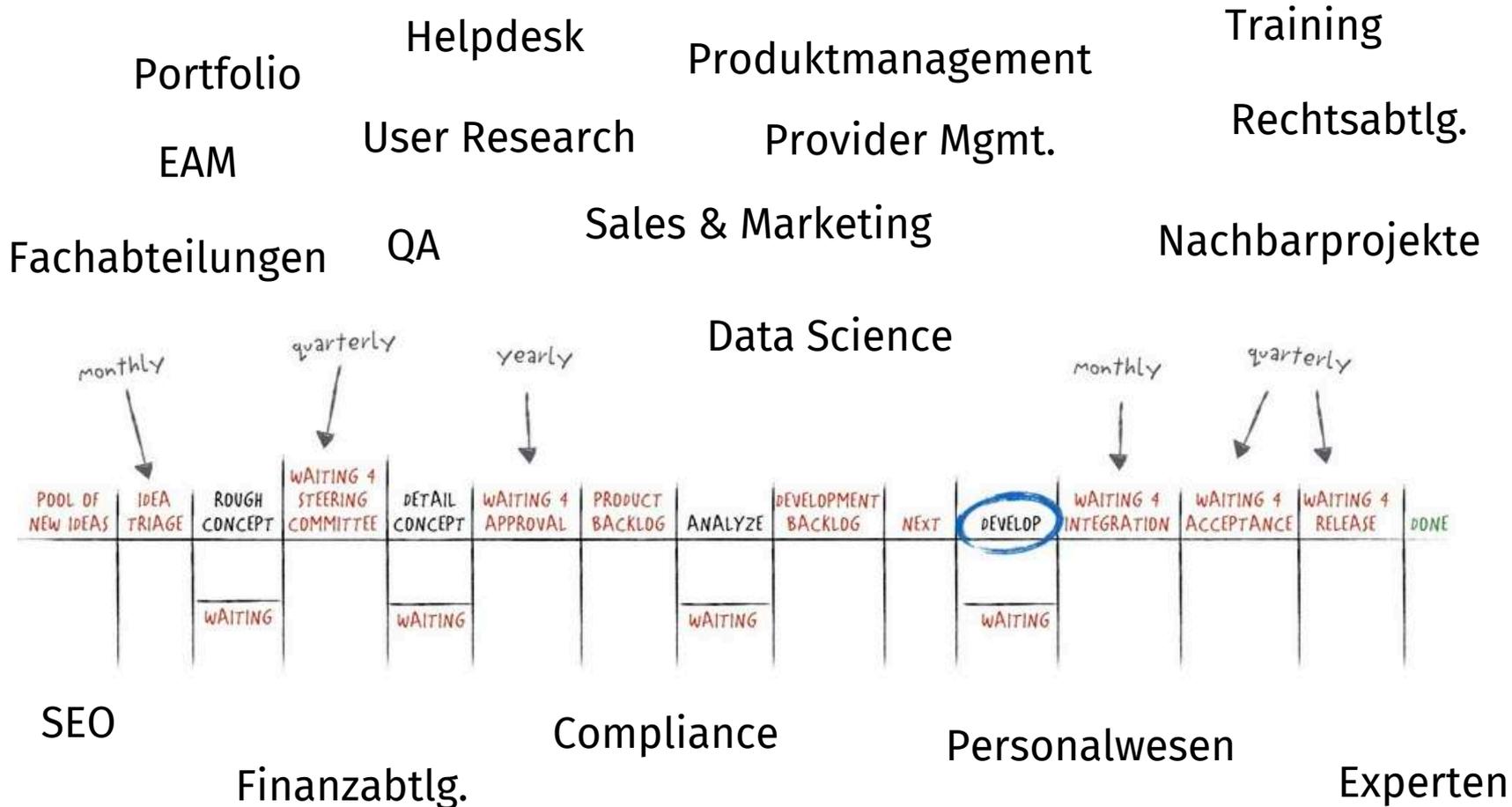
we are so fucking AGILE, yay!!



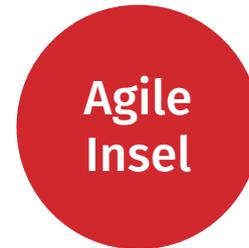
Quelle: Klaus Leopold

Agilität neu denken: Warum agile Teams nichts mit Business Agilität zu tun haben

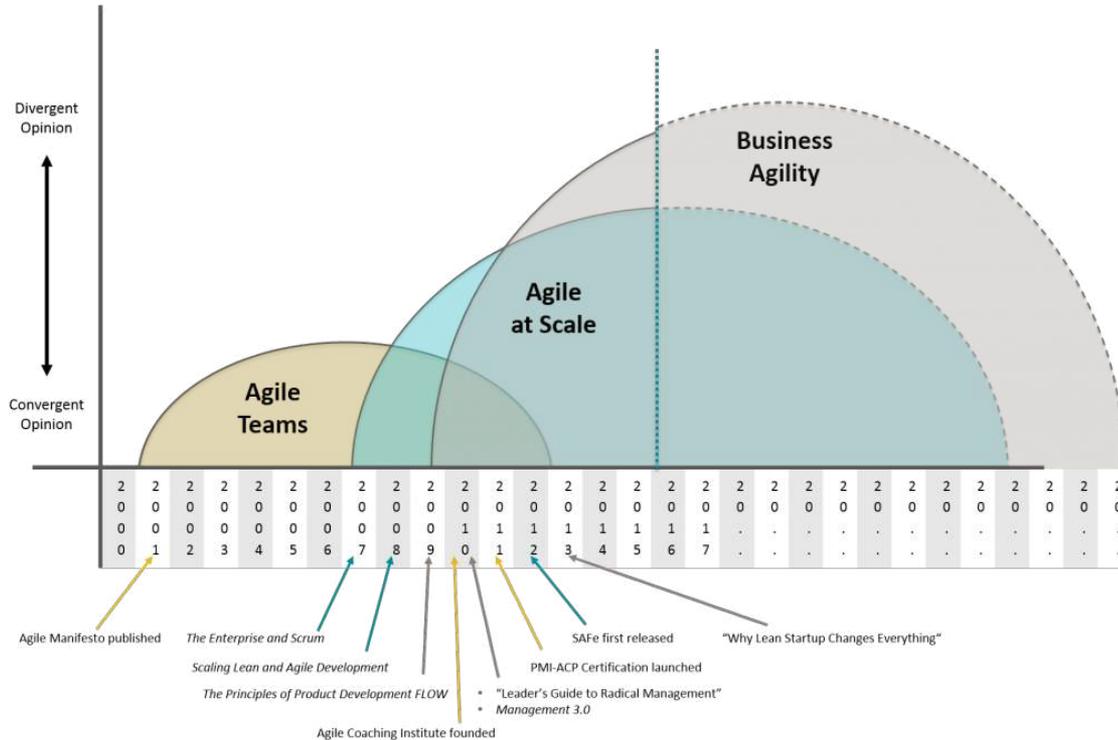




waka waka...



Die 3. Welle der Agilität



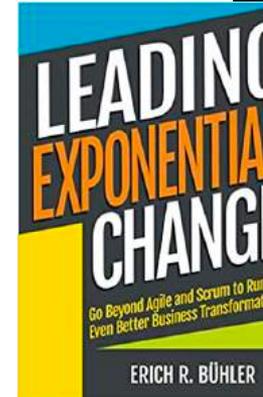
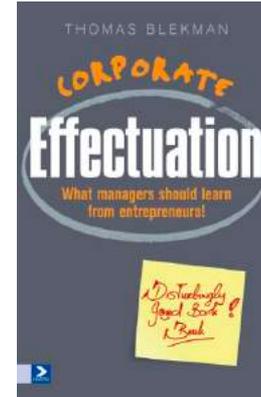
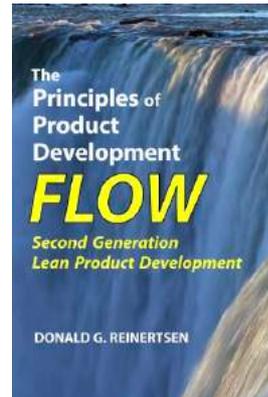
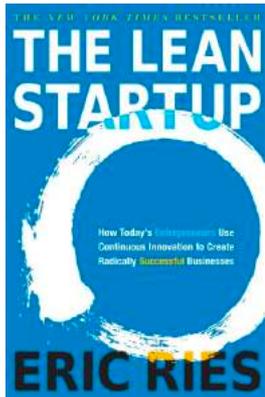
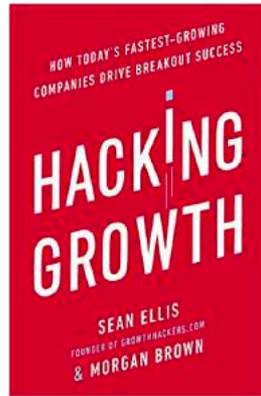
Flexibilität
 Dynamik
 Reaktionsfähigkeit
 Geschwindigkeit

...

© Charlie Rudd



Business Agility überall

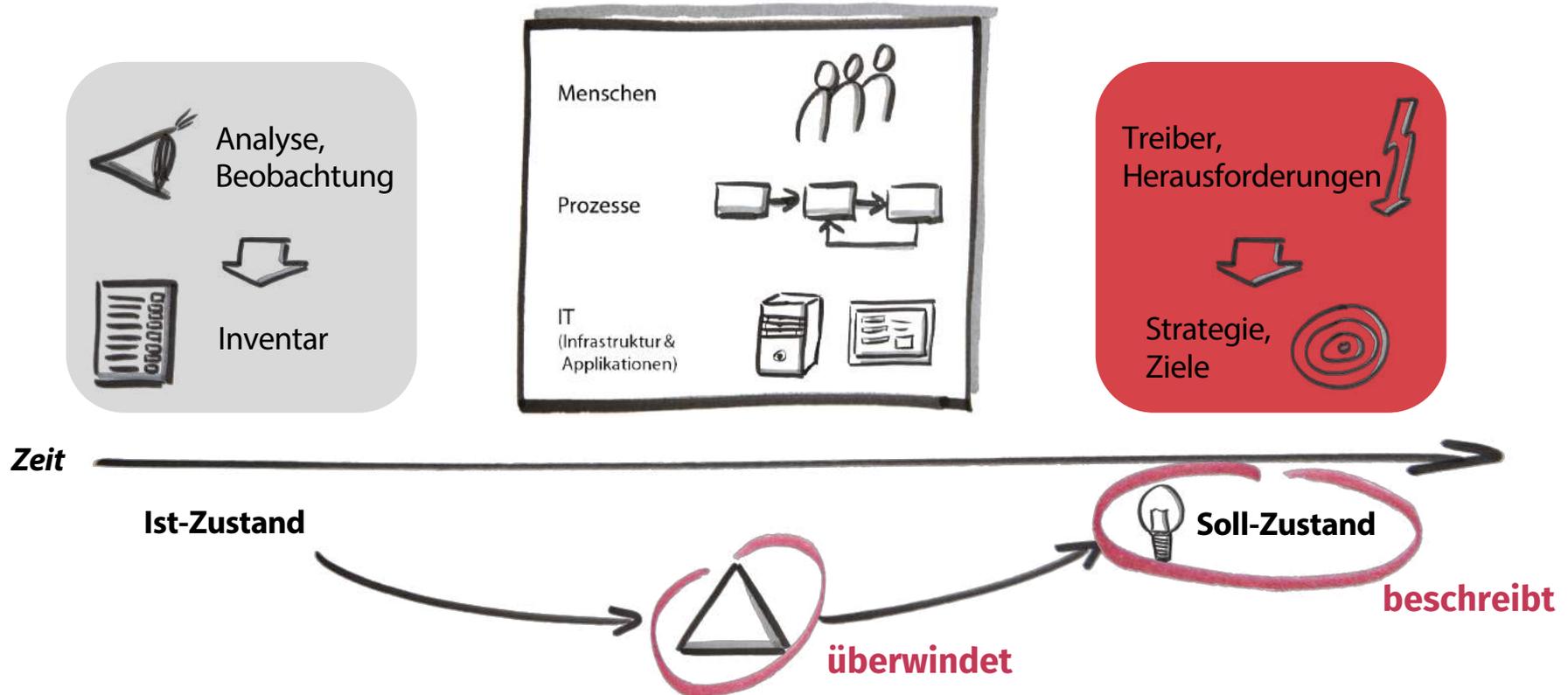


A dark, atmospheric image featuring Darth Vader in the center, holding a lightsaber. The background is a blurred, futuristic environment with glowing lights. On the left side, there is a red, multi-pointed geometric graphic. The text is overlaid on the image.

Enterprise Architektur

Schliesse Dich mir an,
zusammen können wir...

Was macht Enterprise Architektur?



Was hat Enterprise Architektur zu bieten?

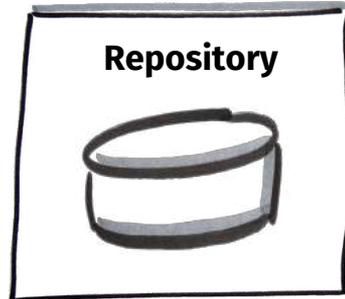
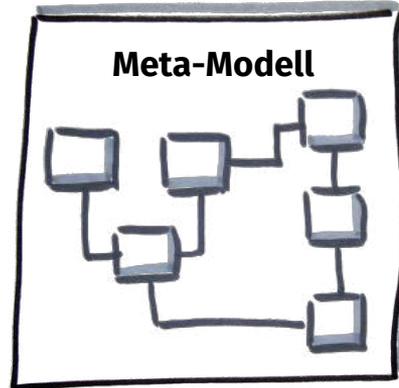
- Eine übergreifende **Architekturvision**
- Abgestimmt mit **geschäftlichen Zielen**
- **Überblick** was technologisch im Unternehmen gemacht wird
- **Transparenz** zu gewachsene Lösungslandschaften
- **Austauschplattformen** zu architektonischen Treibern, Zielen, Lösungen, ...
- Etablierung wichtiger **Prinzipien**
- **Weiterentwicklung** von **Fähigkeiten** und Skills im Unternehmen
- **Offene, gut strukturierte Information** zu Lösungen und Systemen
- Unterstützung von **Compliance Themen** (Risiko, Daten, Standards)



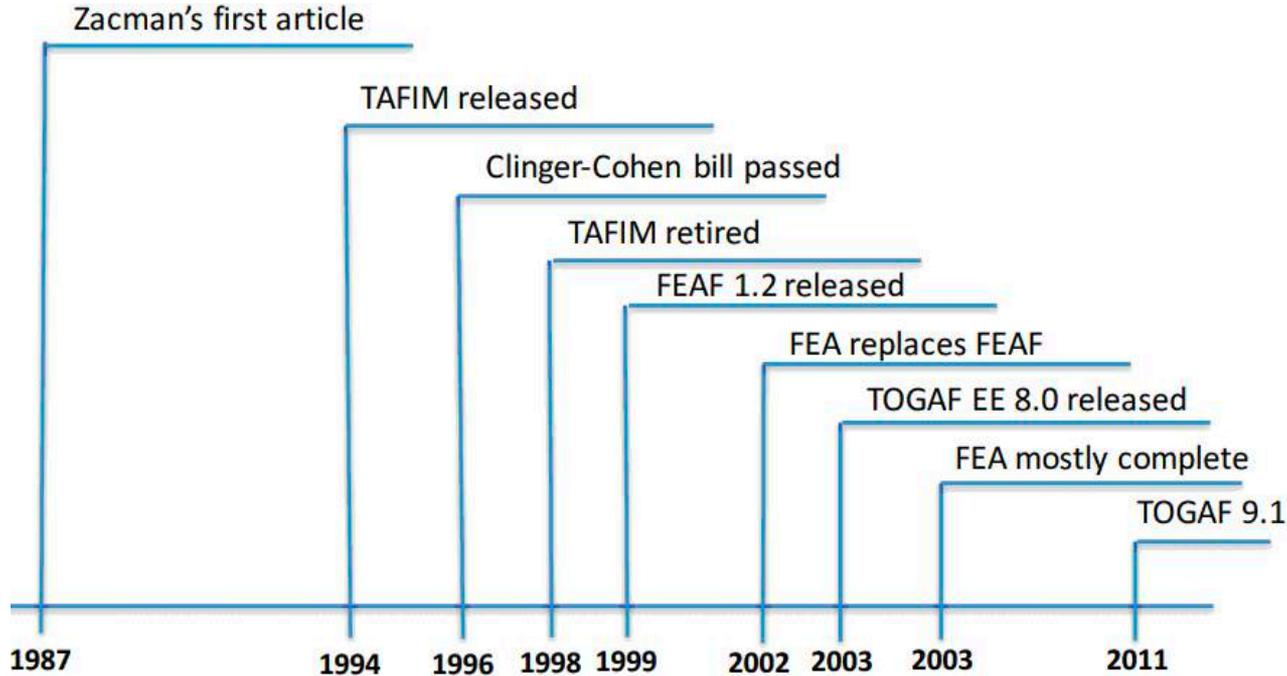
Aber so 90er...



EA - Beschreibung von Ist und Soll...



EA Frameworks



*aus Arnab Chattopadhyaya's
Enterprise Security Architecture*



Was wir häufig sehen

- **Keine Enterprise Architektur**, wenig Richtung, implizite Entscheidungen
- Eine **entkoppelte Enterprise Architektur**: Zahn- und wirkungslos
- Eine **klassische Enterprise Architektur**: einengend und behindernd

Was wir selten sehen:

- **Agile Prinzipien** in Enterprise Architektur
- Enterprise Architektur als **Katalysator für Business Agility**



Die häufigsten Widersprüche

Business Agility

Eigenverantwortung und Freiheit

Kontext und Ziele für „**Informed Captains**“

Diversität als Treiber für Innovation

Lernen und Feedback als Fokus

Entscheidungen lokal durch „Macher“

Wenig Ballast für Veränderung

Ständige Veränderung und Kurskorrektur

...



Enterprise Architektur

Regeln, um inhaltlich zu leiten

Governance für Compliance

Einheitlichkeit als Ziel

Fehlervermeidung als Fokus

Zentrale Entscheidung durch Experten

Umfangreiche Dokumentation

Planung von Soll-Zustand und Weg

...



Wie kann EA für Business Agility sorgen?

1. EA muss sich **neue Ziele** setzen
2. EA muss **anders (inter-)agieren**
3. EA muss zwischen **effizient** und **innovativ** unterscheiden



Neue Ziele

Für moderne EA





Sechs Erfolgsfaktoren identifiziert

1-3

Organisatorisch an agilen Prinzipien ausrichten

4-6

Technisch/konzeptionell auf lose Kopplung und technische Exzellenz achten



Empirisches Vorgehen:

what's the
opposite of
empirical?



theoretical, conjectural,
hypothetical, theoretic,
speculative, unproved,
unobserved, metaphysical



 Thesaurus.plus

1

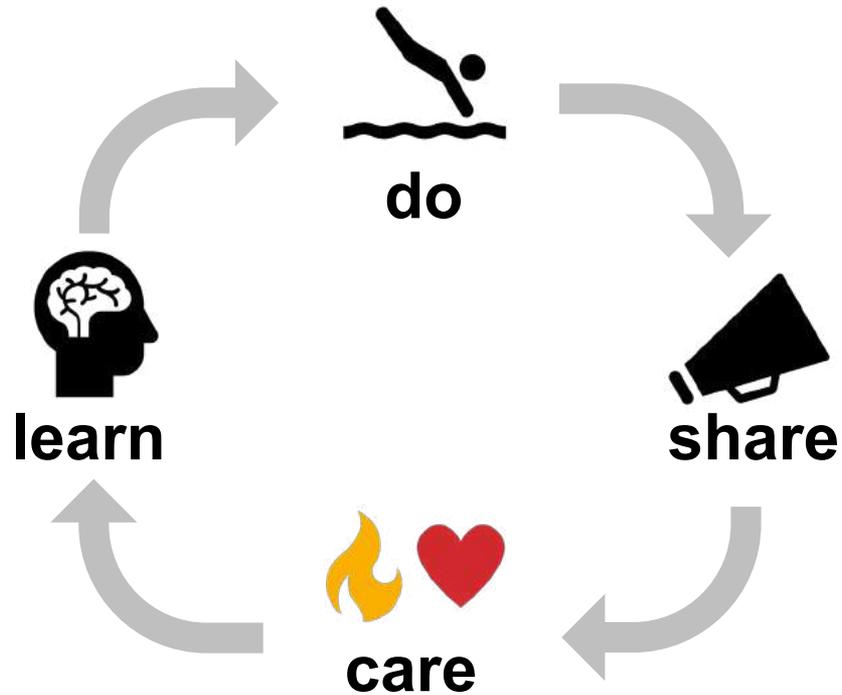


Empirisches Vorgehen – Fragen:

- Wie kommen wir zur **raschen Erprobung** von Ideen?
- Wie können wir (andere) Umsetzungsteams **für Ideen begeistern**?
- Sind **Technologiestandards hilfreich** und ordnend oder störend?
- Wo leben wir Standards-First statt **Eventual Consistency**?
- Wie **nutzen wir Erfolge** von Entwicklern/Teams und deren Lösungen?



Feedback und Transparenz



- Information Radiators
- Communities
- Techtalks
- Fachforen
- Tech-Radar
- Qualitätsziele
- Fitness Functions
- Liberating Structures



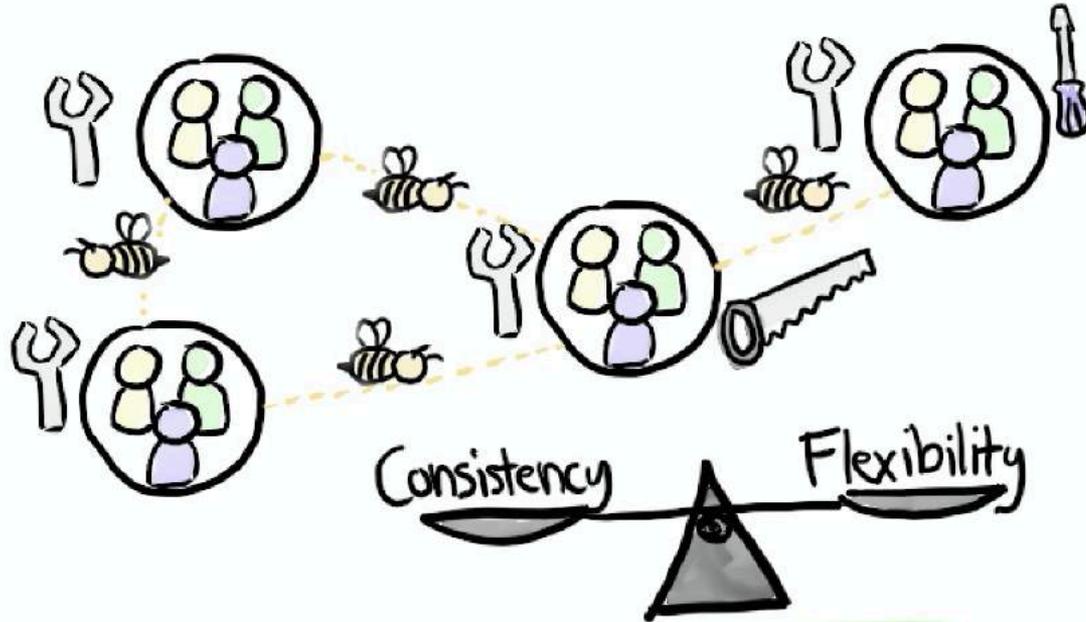
Feedback und Transparenz - Fragen

- Sind aktuelle Architekturideen von Systemen und deren **Evolution sichtbar**?
- Wie **schlank** sind indirekte Kommunikation und **Dokumentation**?
- Gibt es **aktive Communities** zu technologisch, strategischen Themen?
- Hat automatisiertes und **zeitnahes Feedback** für Umsetzer hohe Priorität?
- Sind **Peer-Reviews** auf allen Lösungsebenen üblich?
- Gibt es eine **Kultur** für direktes und ehrliches **Feedback**?



Organisatorisch: Cross-Pollination

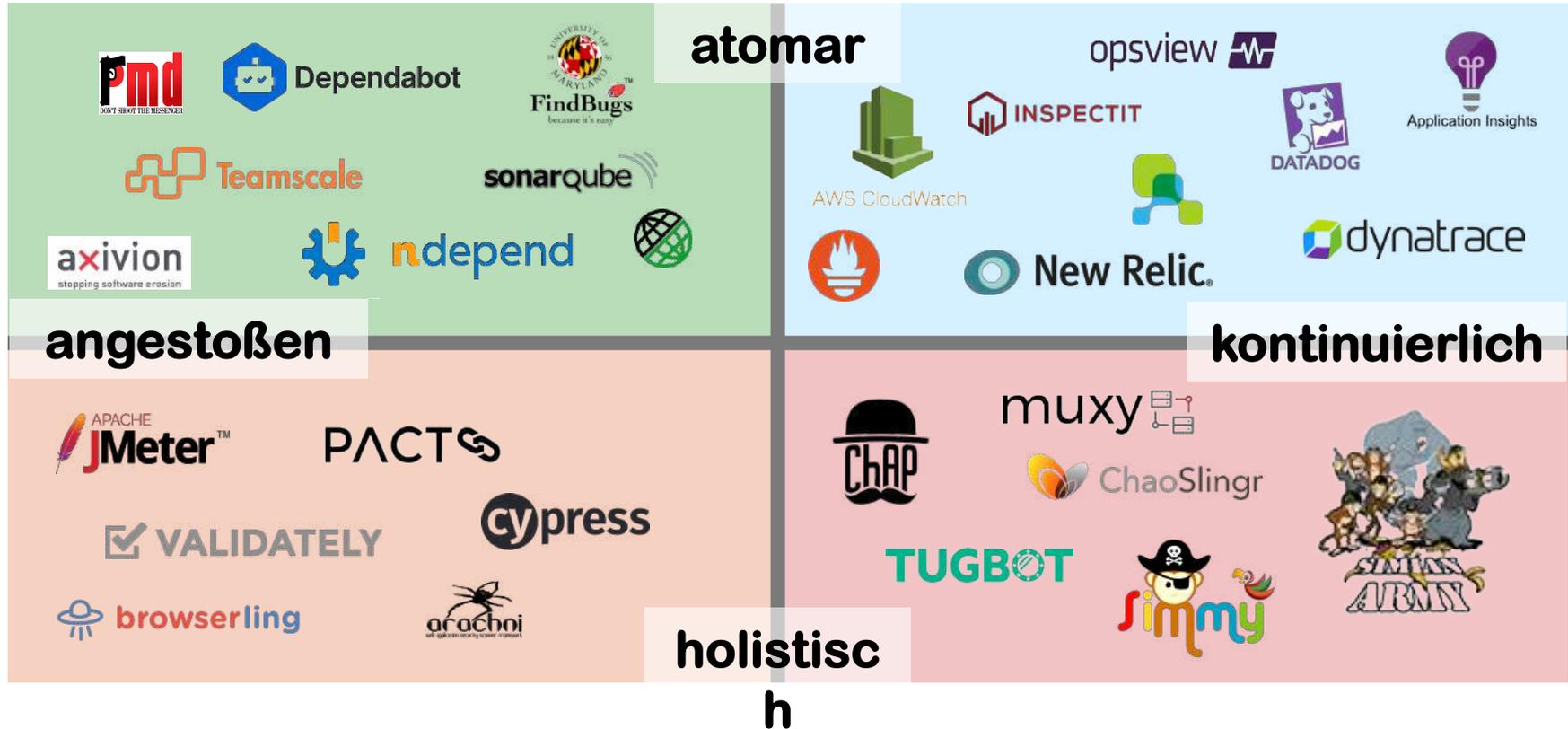
Cross-pollination > Standardization



Henrik Kniberg
(Spotify Labs Blog)



Technisch: Fitness Functions



Basis Fitness Functions

Qualifizier	Bezeichnung	Ziel	Beschreibung	Möglichkeiten / Alternativen / Abwägigungen	Idee für Quality Gate	Proz (Einschätzung embarc)	Voraussetz. / Tools / Frameworks	Was macht?	Vorgabe / Empfehlung / Vorgeh.	Zuweisung / Häufigkeit
Wartbarkeit (Testbarkeit)	Unit Test Coverage	Schnelle Übersicht über Unit Test Coverage	Unit Test Coverage pro Service		TestCoverage > X% TestCoverage = TestCoverage Produkt	Pro 1	Jira SonarQube zur Visualisierung	Jein Team für seine Services	Rot / Grün	Jein CI Build
Wartbarkeit	Statische Code Analyse	Schnelle Übersicht über Unit Test Coverage	Statische Code-Analyse pro Service / Metriken evtl. pro Service über Service unterschiedlich	Metriken evtl. pro Service individuell unterschiedlich Bspw. Abstr. Zahlensysteme (enum Werte), Large Classes/Files	XY Threshold > XY Threshold Punkten (= über nicht-schwarzer werden)	Pro 1	SonarQube FastLaps (CI/CD-integrated CI/CD)	Jein Team für seine Services	Rot / Grün / Blau / Grün	Jein CI Build
Wartbarkeit	Strukturierte Static Code-Analyse	Vorteil von ungenutzten Abhängigkeiten im Code	Evaluieren Abhängigkeitsgraphen pro Service zusammen mit einem Team (Meist Programmierer oder CI-Service) Abhängigkeiten werden separat definiert um eine Architekturfunktion (Kapselungen) und daraufhin zu überprüfen	Welche Metriken nutzt?	Bildet Brüche wenn Regel nicht eingehalten	Pro 2	SonarQube mit Architekt. Grundregeln Optional: SonarQube Integration (jeder für sich)	Jein Team für seine Services	Erklärung / Austausch mit Experten/Leads	Jein CI Build
Sicherheit	OWASP Dependency Check	OWASP Security Check von verwendeten Bibliotheken	Automatische Überprüfung der verwendeten 3rd Party Libraries auf bekannte Sicherheitslücken			Pro 2	OpenSourceVulnerability Scanner whowon plugin	Übergreifend	Rot / Grün	Jein CI Build (oder nightly wenn verfügbar)
Sicherheit	SAST Tests in Sonar	Static Application Security Testing (SAST) in Sonar	Automatische Überprüfung auf Code Ebene für sicherheitsrelevante Bereiche / Fehler			Pro 2	Sonar-Features verwenden (https://www.sonarcloud.org/features/security.html) evtl. auch direkt in CI/CD integriert	Übergreifend	Rot / Grün	Jein CI Build (oder nightly wenn verfügbar)
Wartbarkeit	Container Health Checks	Überprüfung der Service-Health innerhalb Teambereich für seine Deployments	Container Health Checks Triggern zwischen Services bzw. über externen Endpunkt. Services sind via 3rd Party Systemen			Pro 2.3	FACTI Bsp: https://github.com/elastic/containers-k8s/blob/master/docs/monitoring/monitoring.md	Jein Team für seine Services (Übergreifend / Teamverantwortl.)	Grün	Jein CI Build
Wartbarkeit	Dependency Updates	Automatische Updates der 3rd party Libraries		Warnung: Es gibt eine automatische Annotation Migration Tools die auch automatisch Feedback geben, dass die Library in Container ist		Pro 2.3	Bsp. Dependabot	Übergreifend implementiert?	Grün	
Wartbarkeit	3rd Party Libraries	Welche 3rd Party Libraries verwendet in der aktuellen Versionen				Pro 2	Ex: https://www.victoriametrics.com/	Übergreifend	Rot / Grün	

Mittlere Ebene Fitness Functions

Wartbarkeit (Testbarkeit)	Integration Test Code Coverage	Schnelle Übersicht über Integration Test Coverage	Code Coverage für Integration Tests messen	Warnung: Es gibt integrierten Tests	TestCoverage > X% TestCoverage = TestCoverage Produkt	Pro 3	SonarQube Jira	Jein Team für seine Services	Grün	Nightly
Effizienz	Performance Tests	Performance Überblick über wichtige Use Cases	Performance Test für wichtige Integrations-Test Use Cases	Warnung: Es gibt CI/CD Integration Tests	Threshold > xy Latenz / %	Pro 1	SonarQube Jira evtl. Maven Integration in	Jein Team für seine Services	Warnung / Jein Team hat ein Minimum an solchen Tests	Nightly
Zuverlässigkeit	Chaos Engineering (Resilienz)	Service Health über Integrationstests messen	Service in Downstages "resilient" lassen & II-SAP, um über ein spezifisch Maß zu gehen (z.B. Speicher) möglich anzuzeigen (z.B. Resilienz)	3rd Party System für aus Betriebsbetrieber Welt aus Warnung: Es gibt integrierten Tests und Verfügbarkeit für die Abhängigkeiten		Pro 3	evtl. chaos-mesh/spring-bdd	Jein Team für seine Services	Grün	Nightly / in Woche
Benutzbarkeit / Effizienz / Funktionaler Umfang (SEO)	Google Lighthouse (Resilienz)	Google Lighthouse Nightly Test	Google Lighthouse Test für wichtige Strategien (Resilienz)	Ordnung ausgeführt auf einem Build-Agent Lighthouse Test Build-Agent z.B. AWS und GitHub mit einer Testumgebung (z.B. SonarQube)	Lighthouse-Metriken > XY Lighthouse-Metriken > XY Lighthouse-Metriken Produkt	Pro 2	GoogleLighthouse	Jein Team für seine Services	Erklärung / Jein Team hat ein Minimum an solchen Tests	Nightly

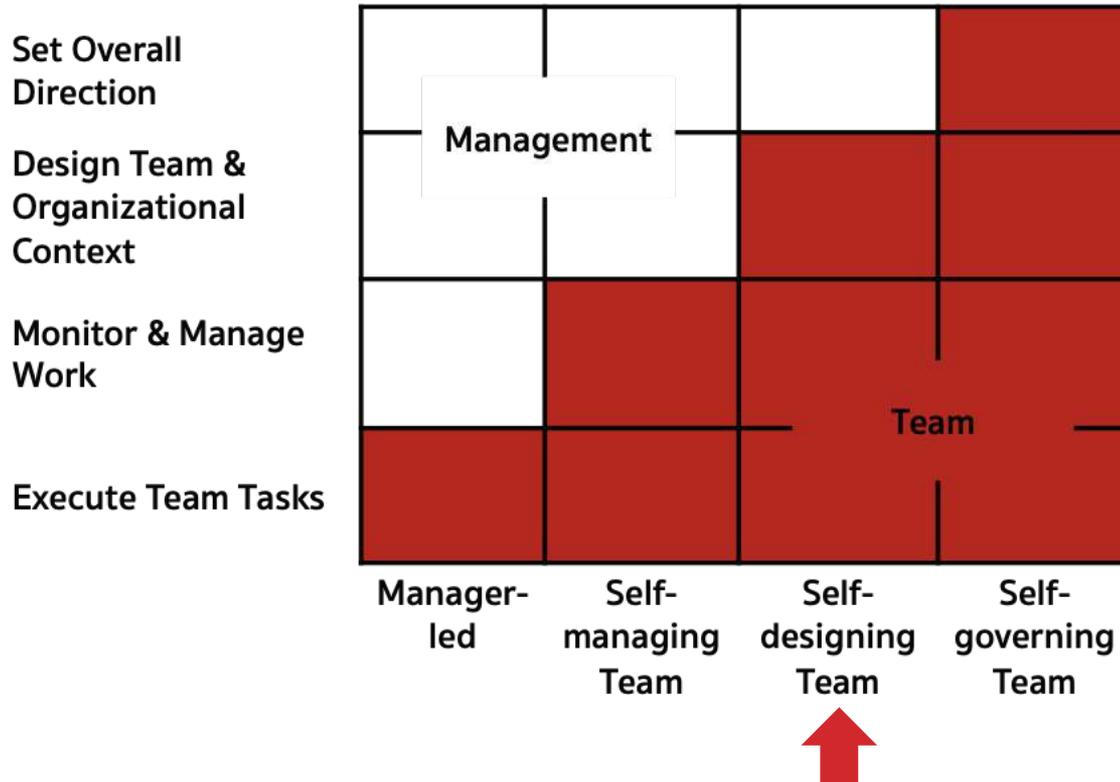
Pyramidenspitze

Benutzbarkeit / Effizienz / Funktionaler Umfang (SEO)	Google Lighthouse PROD	Google Lighthouse PROD Test	Google Lighthouse Test für wichtige Strategien (PROD)	Lighthouse Test über internen B. AWS und GitHub auf PROD-Systeme zu Bei Unsicherheit eines Greenwishes sind einchecken (Bsp. AWS, ...) notwendig		Pro 3	GoogleLighthouse	Übergreifend / Das Release/Produktionsteam	Grün	1x / Stunde (?) für PROD System
Zuverlässigkeit	Health / Zuverlässigkeit / Metriken	Health / Zuverlässigkeit / Metriken	Health / Zuverlässigkeit / Metriken	Health / Zuverlässigkeit / Metriken	Health / Zuverlässigkeit / Metriken	Pro 3		Übergreifend / Das Release/Produktionsteam	Grün	Continuous
Benutzbarkeit	Visual Regression in Testing	Visual Regression in Testing	Visual Regression Tests für wichtige Strategien			Pro 2		Jein Team für seine Web-Systeme	Grün	Nightly / in Woche

Monitoring & Alerting



Verantwortung in Umsetzungsteams



Richard Hackman's Authority Model

3



Verantwortung in Umsetzungsteams - Fragen

- Sind **wenige Teams**/Abteilungen am **Lebenszyklus** einer Lösung beteiligt?
- Sind **erfolgreiche Softwareprodukte** der wichtigste **Maßstab** für „Erfolg“?
- Sind **strategische** und qualitative **Ziele** auf Umsetzungsebene präsent?
- Zeigen Umsetzungsteams auch ohne starke Führung **Motivation**?
- Wie **klar** ist der **Verantwortungsbereich** von Umsetzungsteams?

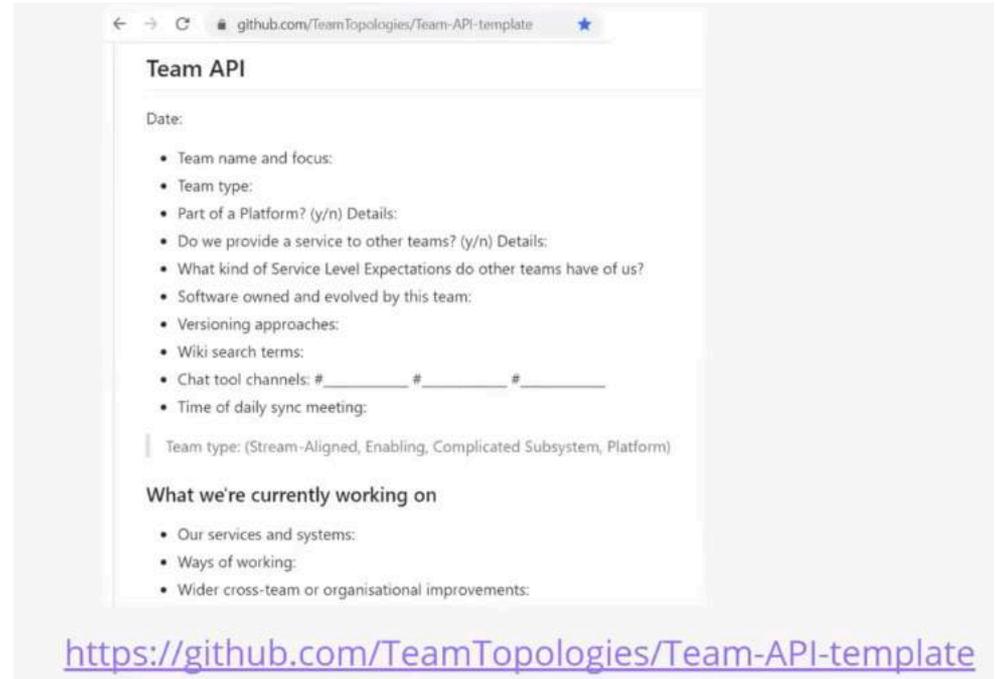
3



Team APIs

- Wie arbeitet das Team?
- Wie kann/soll man mit dem Team interagieren?

„For effective team-first ownership of software, teams need to continuously define, advertise, test and evolve their Team API“

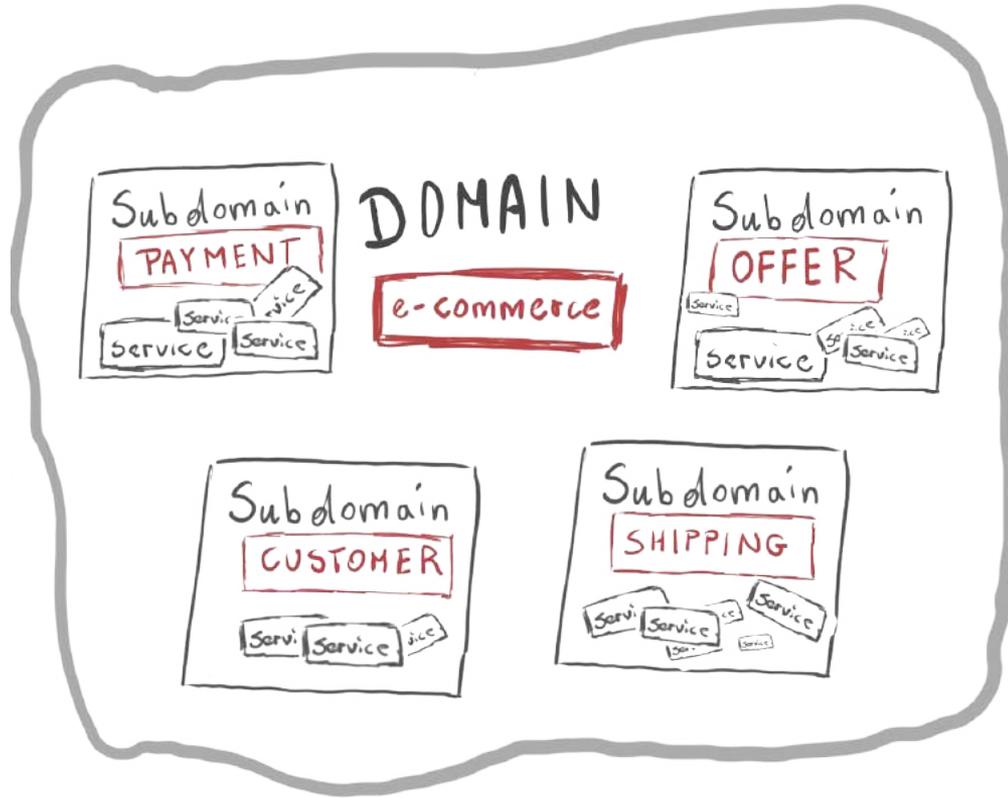


Team Dependency Tracking

Team name	Depends on Team	Type (blocking/slowing/ok)	Cross-streams (Yes/No)	Short description of dependency (artifacts, approvals, other)
Analytics	DBA	slowing	NA	sometimes we need data exports that can take several days because DBA team is busy
Self Heal	Mobile	slowing	NA	mainly synchronize changes to data being used for recommendations, but even small changes take time because Mobile team is constantly overloaded
Self Heal	Ops	blocking	NA	for deploying changes
Self Heal	DBA	blocking	NA	for reviewing schema changes
Self Heal	InfoSec	blocking	NA	for changing firewall settings and other configurations
QA	InfoSec	blocking	NA	when we suspect there might a security violation but it takes very long to get an answer and be able to create the final test report
DBA	Ops	ok	NA	provisioning more storage
IoT	Networking	slowing	NA	changes in networking rules can take a few days to become active
IoT	InfoSec	blocking	NA	sometimes new features are ready but takes weeks to validate security
Sensors	Networking	ok	NA	
Sensors	Mobile	slowing	NA	can't update firmware until mobile apps use the new version
Support	Ops	blocking	NA	we need to wait for Ops team for infra related tickets
Support	Mobile	blocking	NA	we need to wait for Mbile team for mobile related tickets
Support	Self Heal	blocking	NA	we need to wait for Self Heal team for recommendations related tickets



Vertikalität



- Domänenorientierung
- Klare Produktgrenzen
- Tiefe Entkopplung
- Private Datenmodelle
- Interaktionsregeln
- ...

4

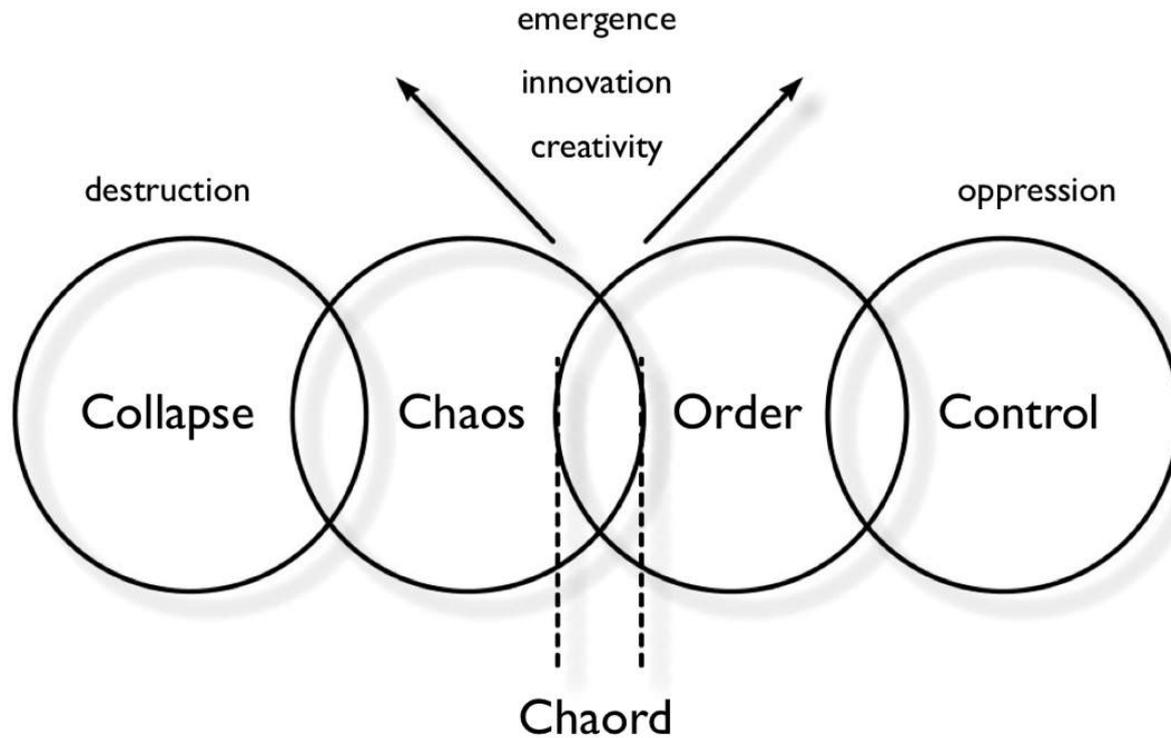


Vertikalität - Fragen

- Wie gut mappt die **Domäne** auf einzelne Applikationen/**Produkte**?
- Sind technische **Abhängigkeiten** zwischen Produkten **gering**?
- Sind übergreifende **technologische Aspekte wenig einschränkend**?
- Sind die **Interaktionsprinzipien** zwischen Applikationen **klar** definiert?
- Ist die **Ursache** für **Verzug oder techn. Probleme** immer in **einem Umsetzungsteams** selbst zu finden?



Architektonische Emergenz



© Simon Robinson From *Holonomics: Business Where People and Planet Matter*, Floris Books, 2014

5



Architektonische Emergenz - Fragen

- Gibt es mehr **Vorschläge und Defaults** als harte Regeln?
- Würden faule Leute zentralen **Architekturideen folgen** oder sie umgehen?
- Gibt es verstandene, und breit akzeptierte **Architekturprinzipien**?
- Sind **zentrale Architekturrollen** mehr „**Kurator**“ als „**Terminator**“?
- Werden **Innovationspotentiale** auch **bottom-up** getrieben?
- Werden **Fehlschläge** offen und gern **kommuniziert**?

5



Technische Exzellenz

Benötigte „Talentdichte“

Business Komplexität

„Talentdichte“

benötigte Prozesse

Prozesse und Regeln

gute Leute

6

Quelle: Reed Hastings
(Netflix CEO)



Technische Exzellenz - Fragen

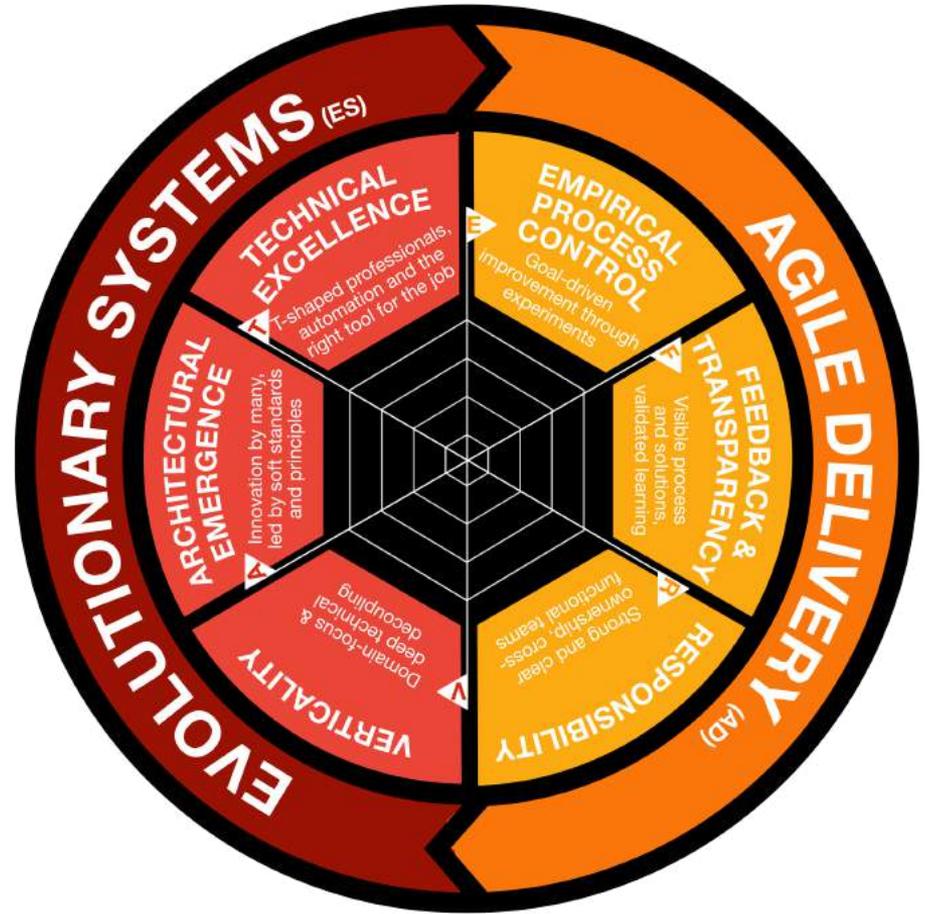
- Sind **persönliche Weiterentwicklung** und entsprechende Freiräume normal?
- Ist **Wissen und Können wichtiger** als Position und Politik?
- Sind **Qualitätsbewusstsein** und Produktfokus als Haltung **breit** gestreut?
- Haben **Trends und Erkenntnisse** eine große Bühne?
- Gibt es wenige „**undone-departments**“?

6



Das ADES-Framework

- **Wichtige Eigenschaften** von agilen Organisationen
- **6 Lerngebiete.** Enthalten Beispiel-Techniken für Experimente
- **Iterative Optimierung** der Lerngebiete
- Wiederholte **Selbsteinschätzung** (Spinnennetz)
- **KEIN** Maturity Modell
- **KEIN** Skalierungs-Framework



<http://ades-framework.org/>



Neues Verhalten

Was steckt hinter den Zielen?



New Style Enterprise Architecture

- Only constraint what's necessary
- Focus on low coupling (technical and organizational)
- Lead with context and goals
- Favour marketing over governing rules
- No exclusive information
- Support good people's ideas
- Learn from everybody
- Go for empirical data... always





Einordnung

Business Agility & EAM

Dynamik/Innovation vs. Effizienz/Standards



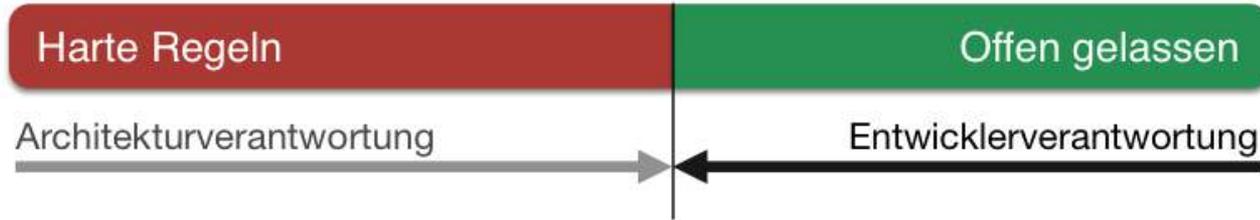
CAPABILITY STATES

--	No change is required.
X	No longer needed.
L	Low level of change is required.
M	Medium level of change is required.
H	High level of change is required.
N	Capability does not yet exist.

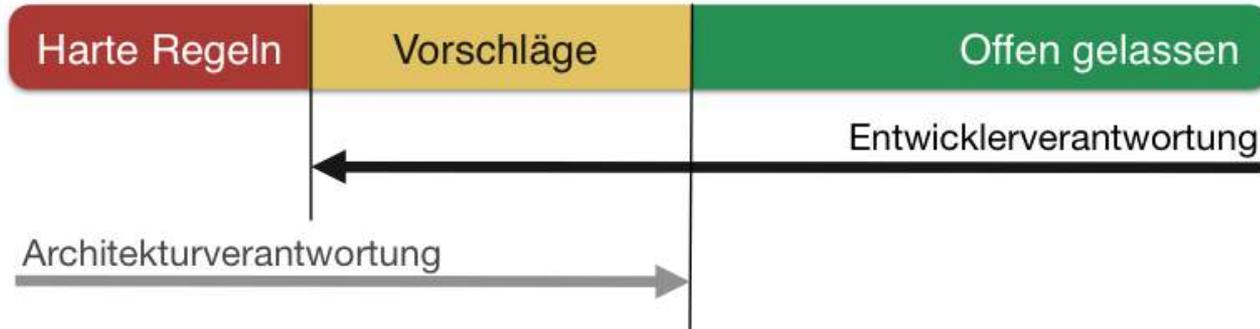
- Capability Maps **unterscheiden** zwischen Capabilities. z.B.:
 - Core/Supporting/Generic
 - Change need (siehe links)
 - Lifecycle (Plan bis EOL)
 - Functional Fit
 - ...
- Wo ist **Einheitlichkeit** wichtig?
- Wo **Innovation**?

Freiheiten und Kontext \neq Anarchie

Klassische Architekturvorgaben



Weiche Architekturvorgaben



Zusammenfassung

Was könnt ihr mitnehmen?



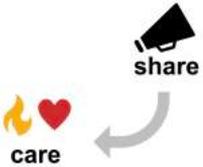
Die wichtigsten Aspekte



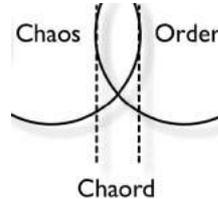
Fokus auf
Empirie

DOMAIN
e-commerce

Vertikalität &
Unabhängigkeit



Transparenz
& Feedback



Emergenz durch
„Chaord“



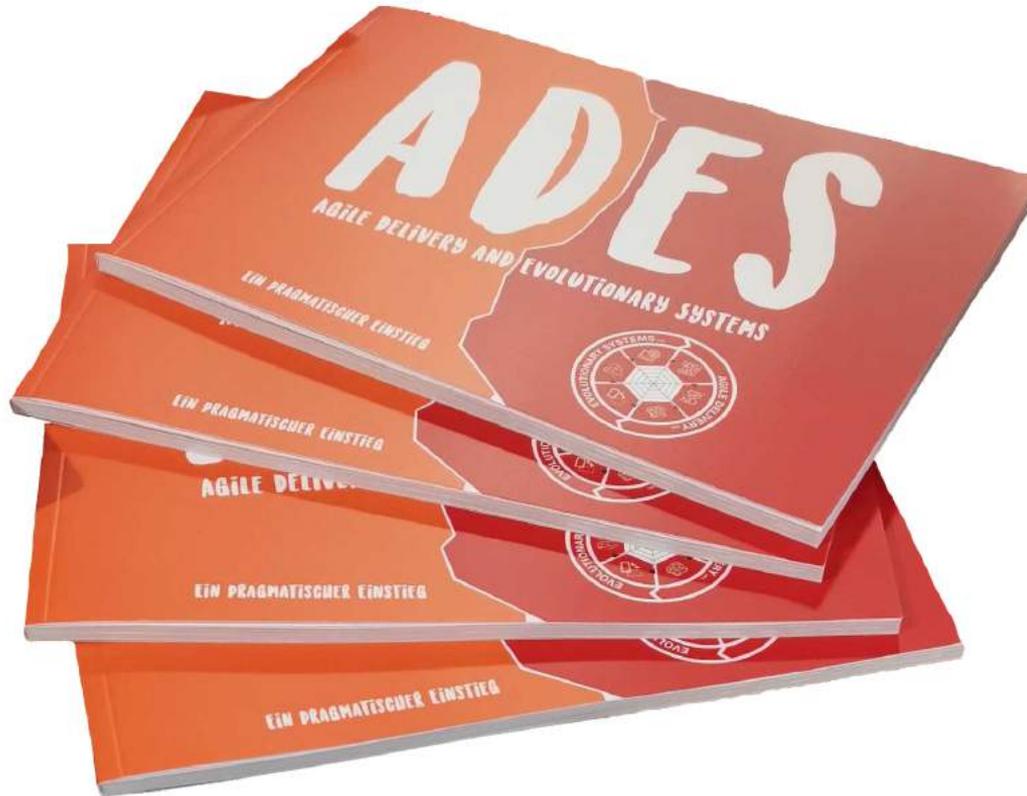
Verantwortung
bei Umsetzung

„Talentdichte“
benötigte Prozesse

Technische
Exzellenz



Agile Organisationsarchitektur



HOME

WHY ADES?

ADES FRAMEWORK

FAQ

CONTACT

<http://ades-framework.org/>



Danke.

Jegliche Fragen sind willkommen!

 stefan.toth@embarc.de

 @st_toth

 xing.to/sto


embarc 
Software Consulting GmbH

