

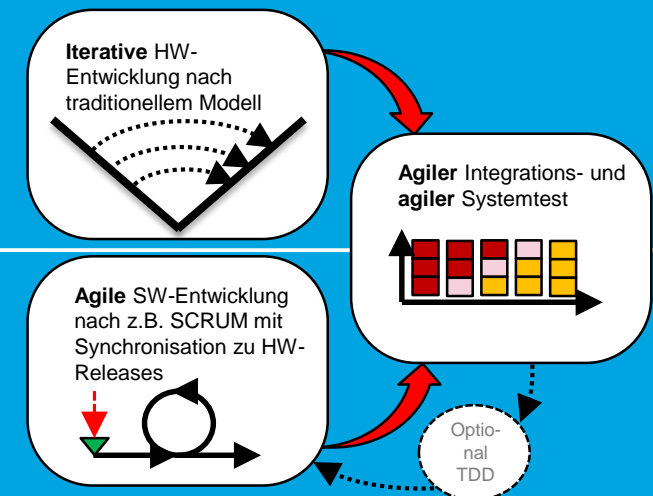
Agile Methoden richtig einführen

Die agile Unternehmenstransformation

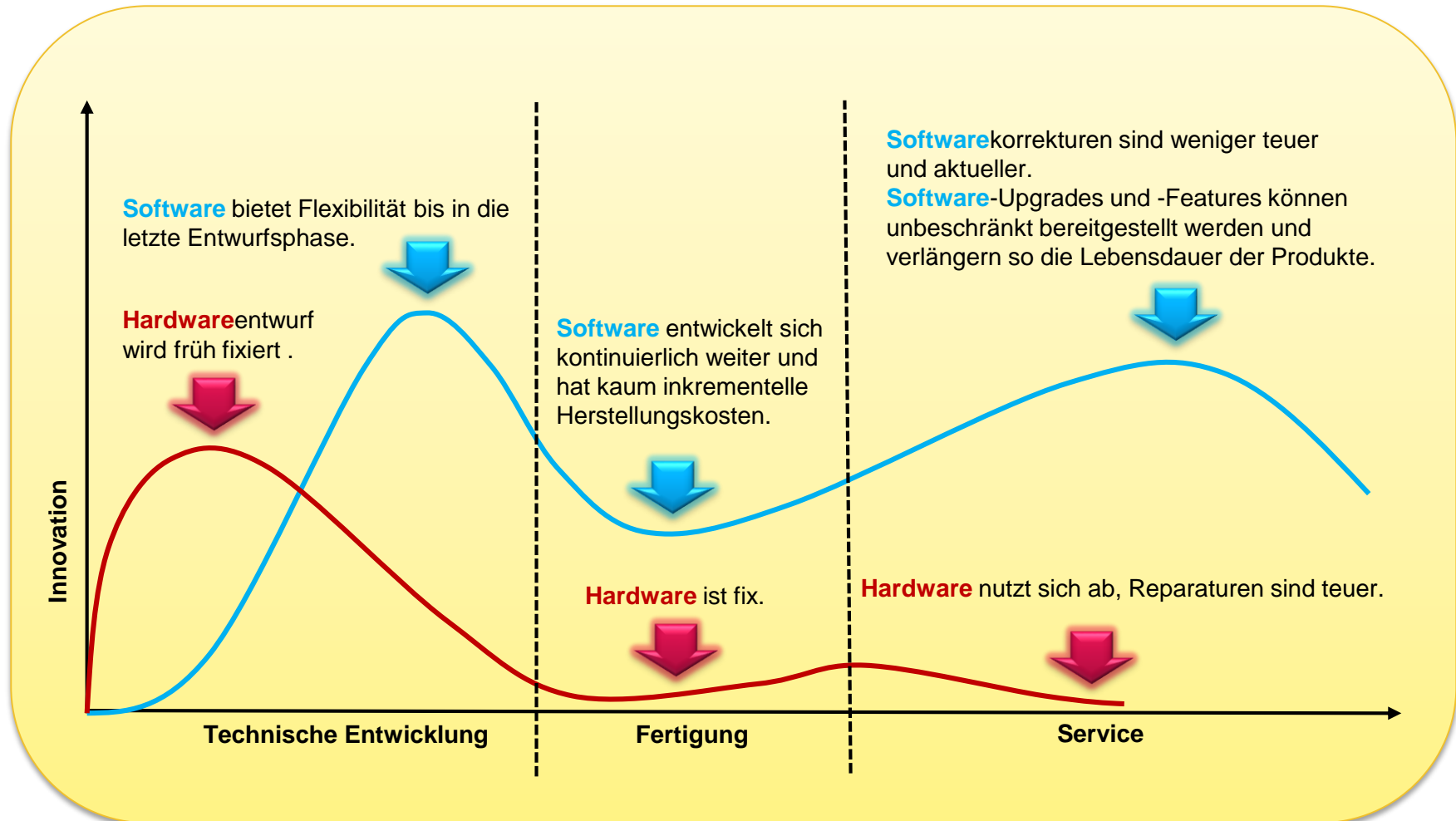
Dipl.-Ing. Univ.

Remo Markgraf

MicroConsult GmbH

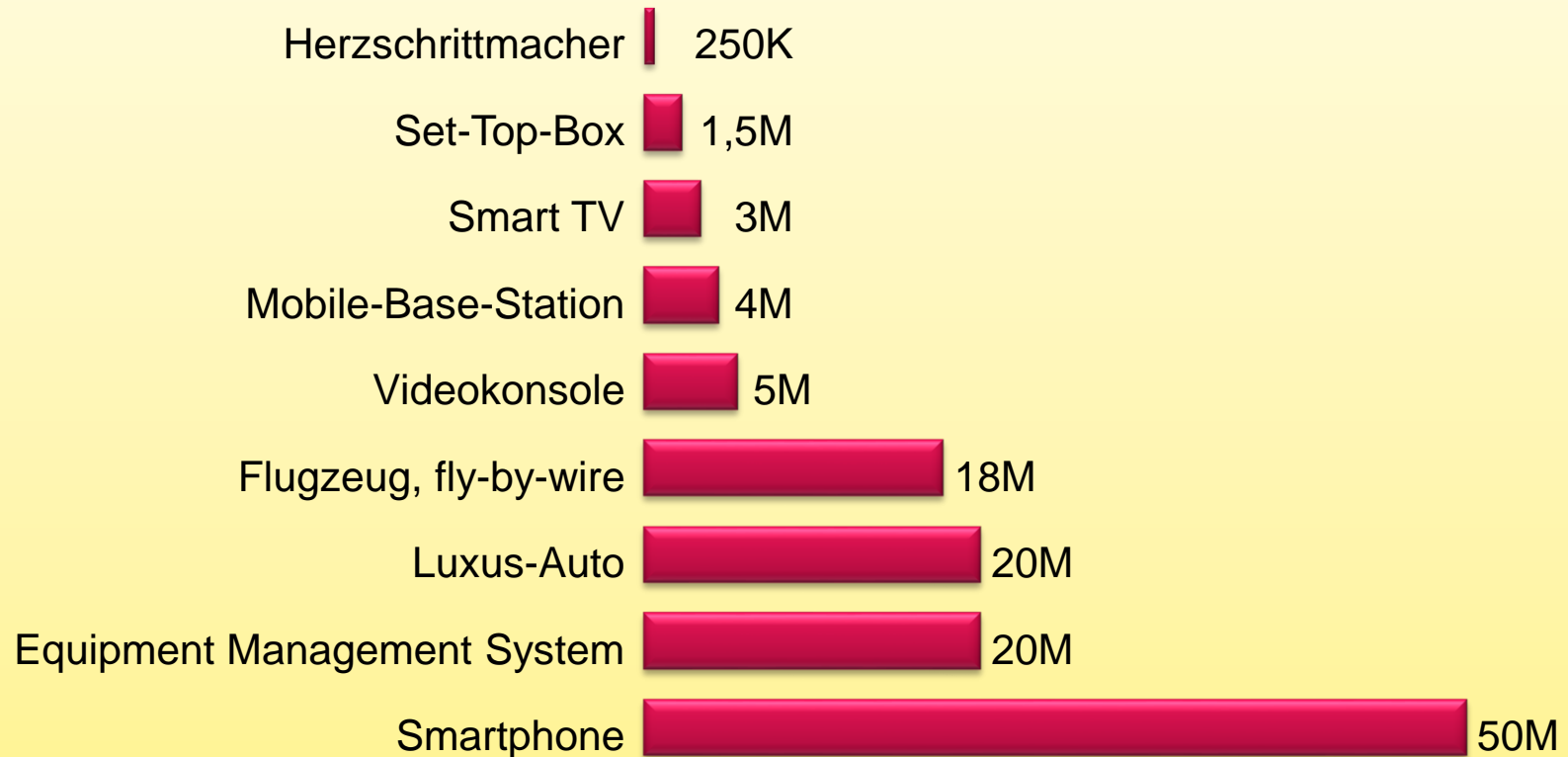


- Software ermöglicht kontinuierliche Innovation



Quelle: Intelligente softwareintensive Systeme, Systems Engineering für komplexe vernetzte Produkte, ELEKTRONIKPRAXIS, Vogel Verlag, Parametric Technology GmbH, Esslingen am Neckar, Feb :2014

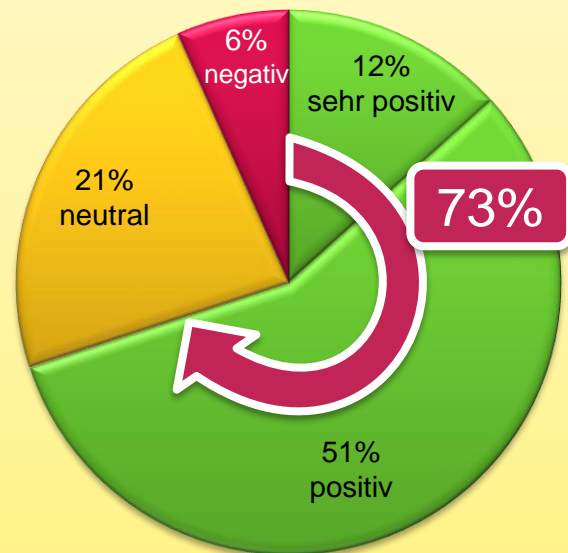
- Embedded-Software wird immer komplexer



Quelle: Aritomo Shinozaki, Walter Sun, Accelerating embedded software development via agile techniques. The nine strategies that lead to successful embedded software development, PricewaterhouseCoopers, June 2013

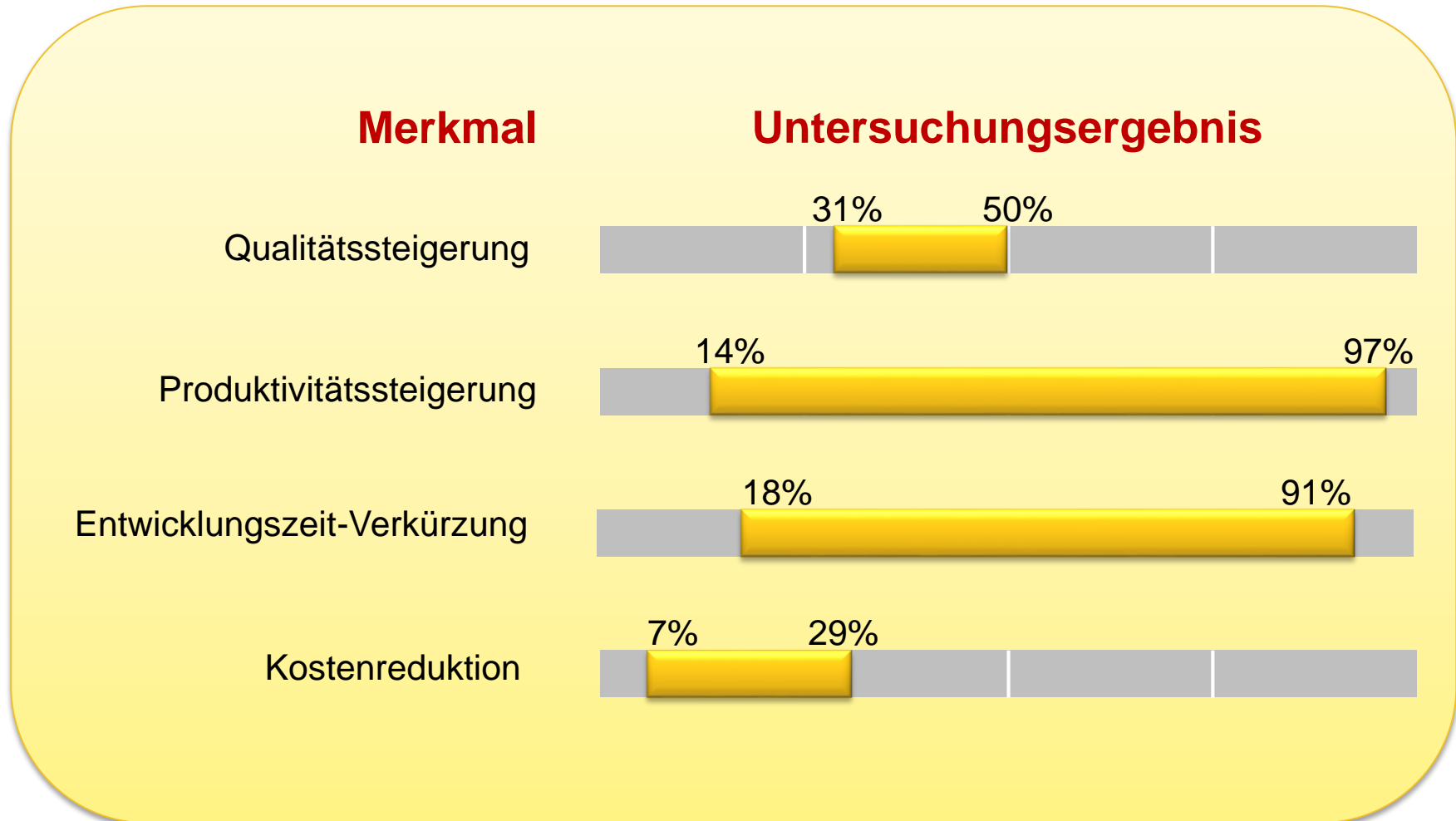
• Ergebnisse der AGILE-ITEA Untersuchung

- Domäne: Agile Softwareentwicklung für Embedded-Systeme
- 68 Pilotprojekte innerhalb von 2 Jahren
- 1800 Entwickler in 17 verschiedenen Unternehmen
- 73% beurteilen die agile Arbeitsweise positiv oder sehr positiv



Quelle: Itea innovation report 2006, 29.12.2008 (C) VTT, Pekka Abrahamsson

- Vorteile der agilen Entwicklung von Embedded-Software



Quelle: Aritomo Shinozaki, Walter Sun, Accelerating embedded software development via agile techniques. The nine strategies that lead to successful embedded software development, PricewaterhouseCoopers, June 2013

Übergang von der Massenfertigung schlechter und teurer Software zu innovativen und wirtschaftlich erfolgreichen Lösungen



- 2001 in den verschneiten Bergen von Utah
- 17 Menschen legen den Grundstein für das Agile Manifest*
- Heute: 3379 Buchtreffer bei amazon.com zum Suchbegriff „Agile Software“

*) <http://agilemanifesto.org>

- Agile wird durch 4 Werte und 12 Prinzipien definiert.
- Alle involvierten Bereiche unterstützen diese Werte und Prinzipien.
- Das bedeutet, man ist agil und weniger, man entwickelt agil.

Manifest für Agile Softwareentwicklung

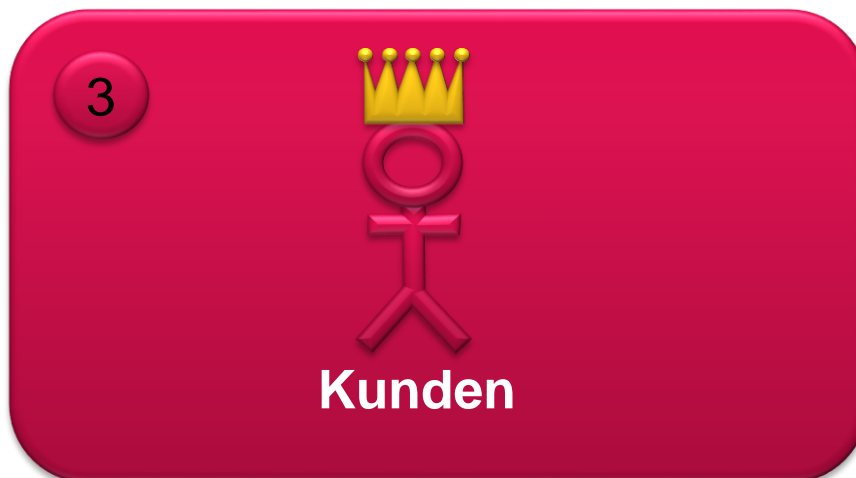
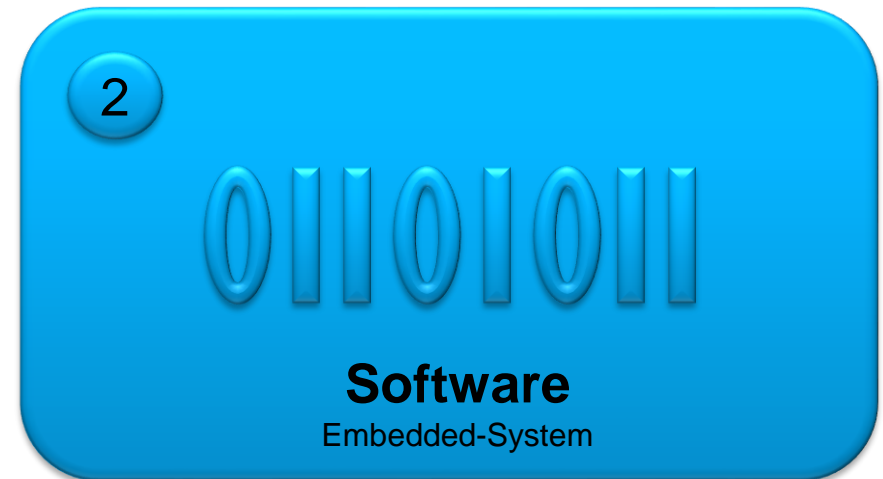
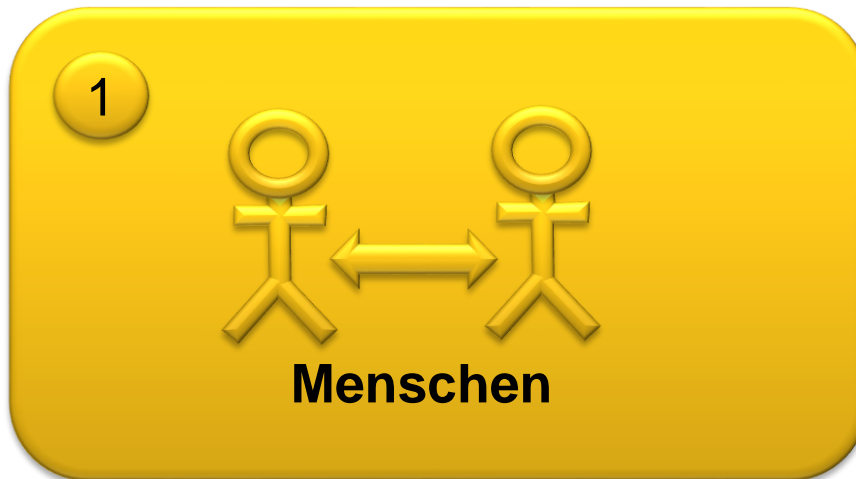
Wir erschließen bessere Wege, Software zu entwickeln,
indem wir es selbst tun und anderen dabei helfen.
Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:

- 1 Individuen und Interaktionen mehr als Prozesse und Werkzeuge
- 2 Funktionierende Software mehr als umfassende Dokumentation
- 3 Zusammenarbeit mit dem Kunden mehr als Vertragsverhandlung
- 4 Reagieren auf Veränderung mehr als das Befolgen eines Plans

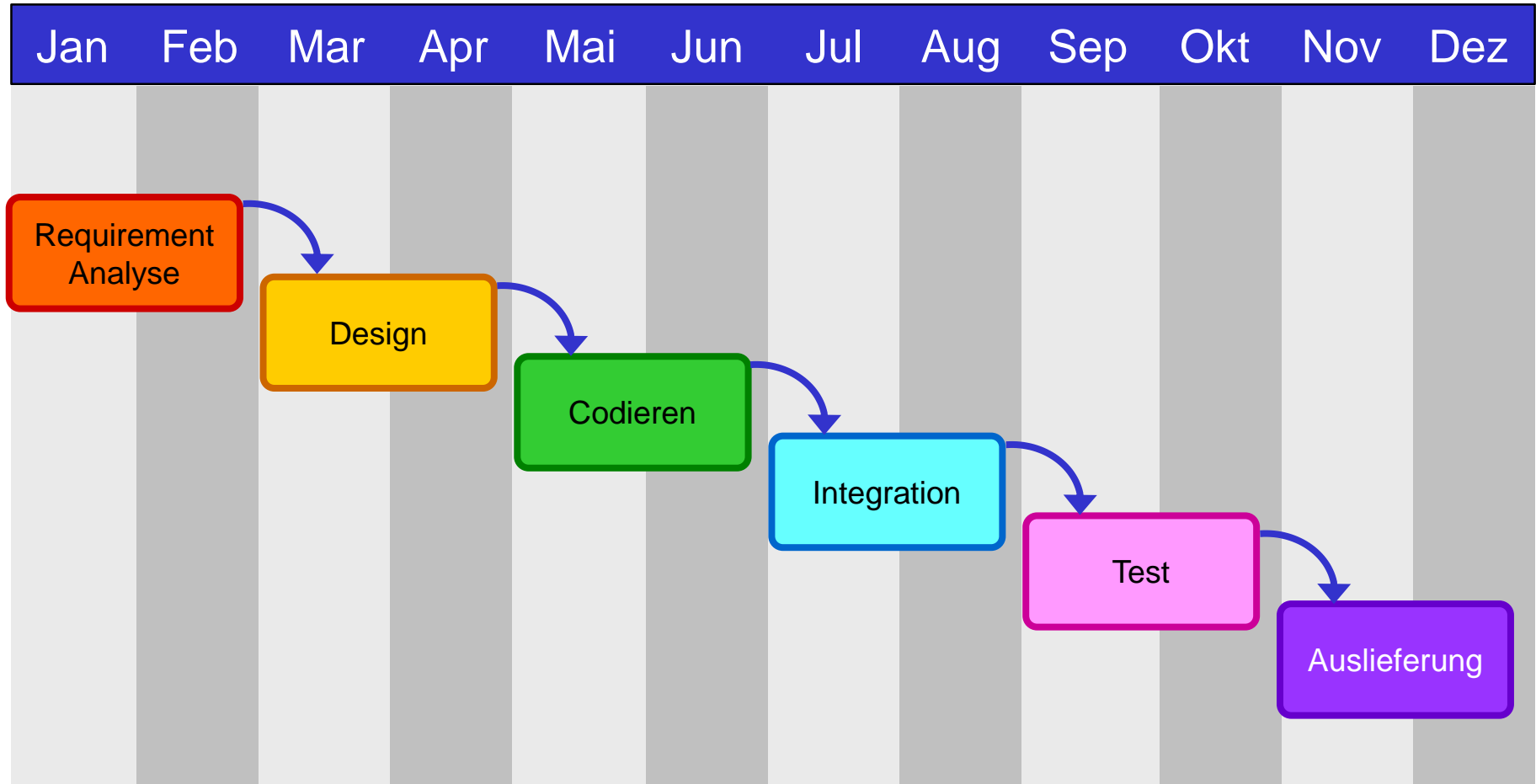
Das heißt, obwohl wir die Werte auf der rechten Seite wichtig finden,
schätzen wir die Werte auf der linken Seite höher ein.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

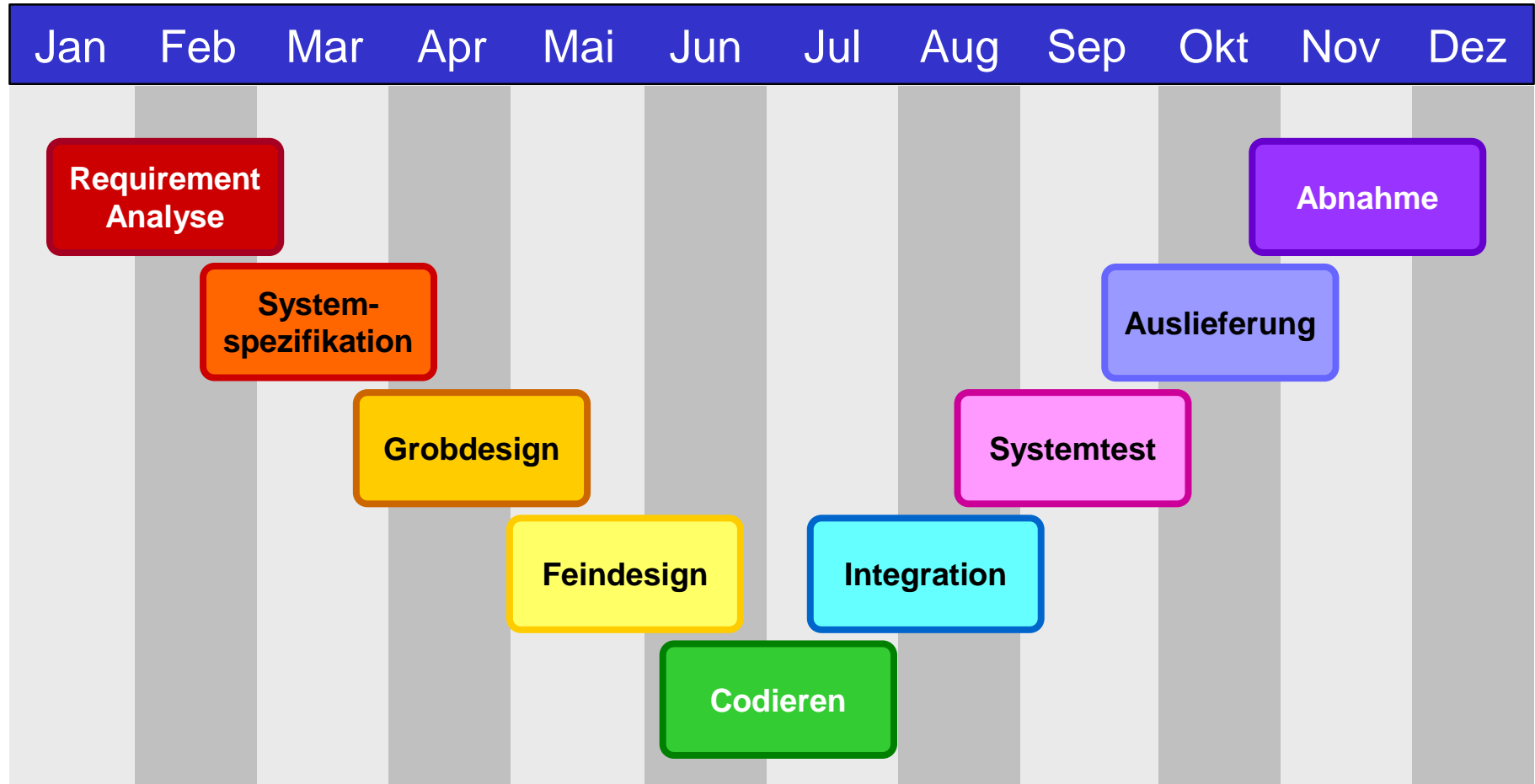
- Zusammenfassend: Agile wird durch 4 Werte definiert:



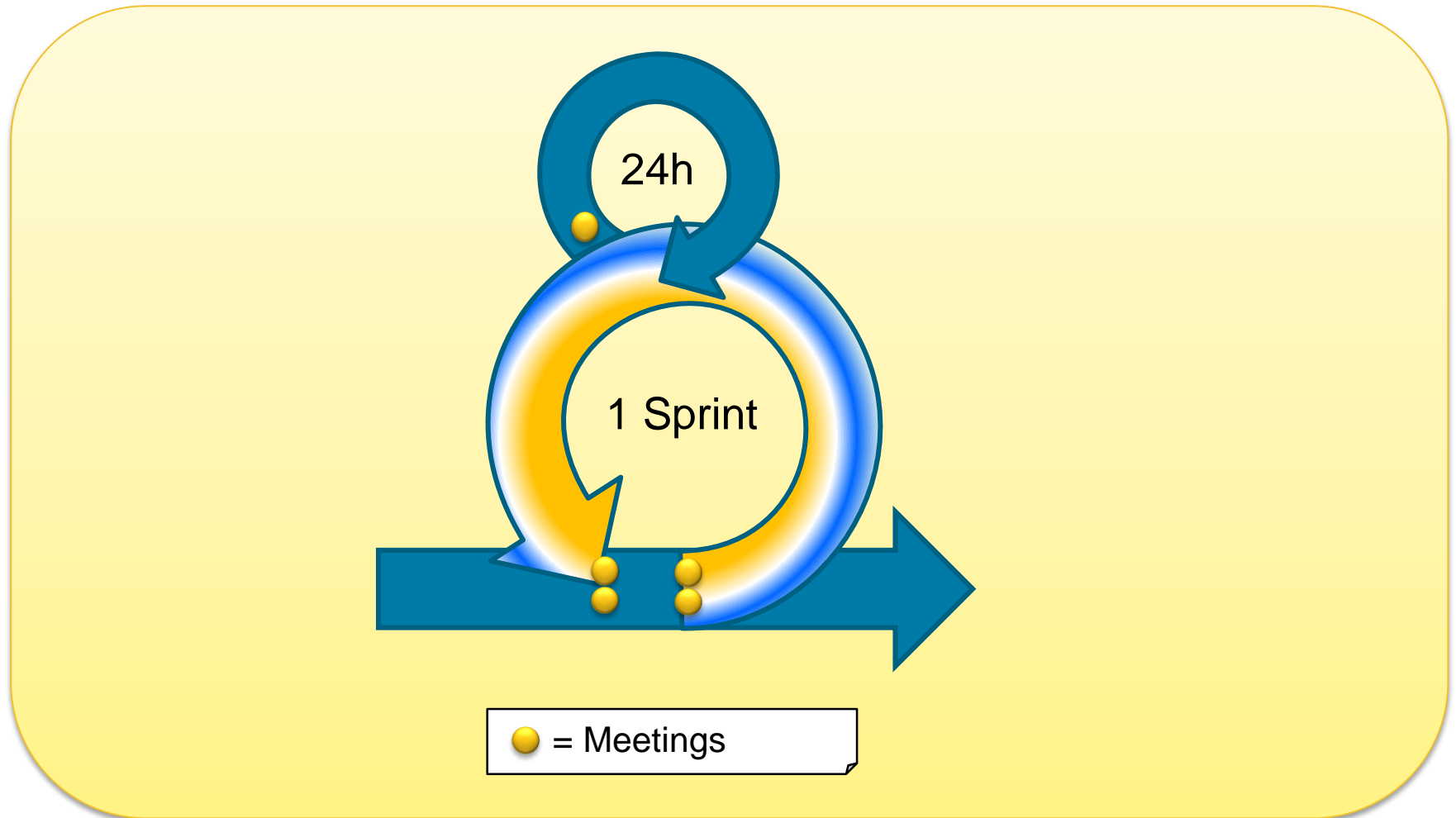
Traditionelles Wasserfall-Modell



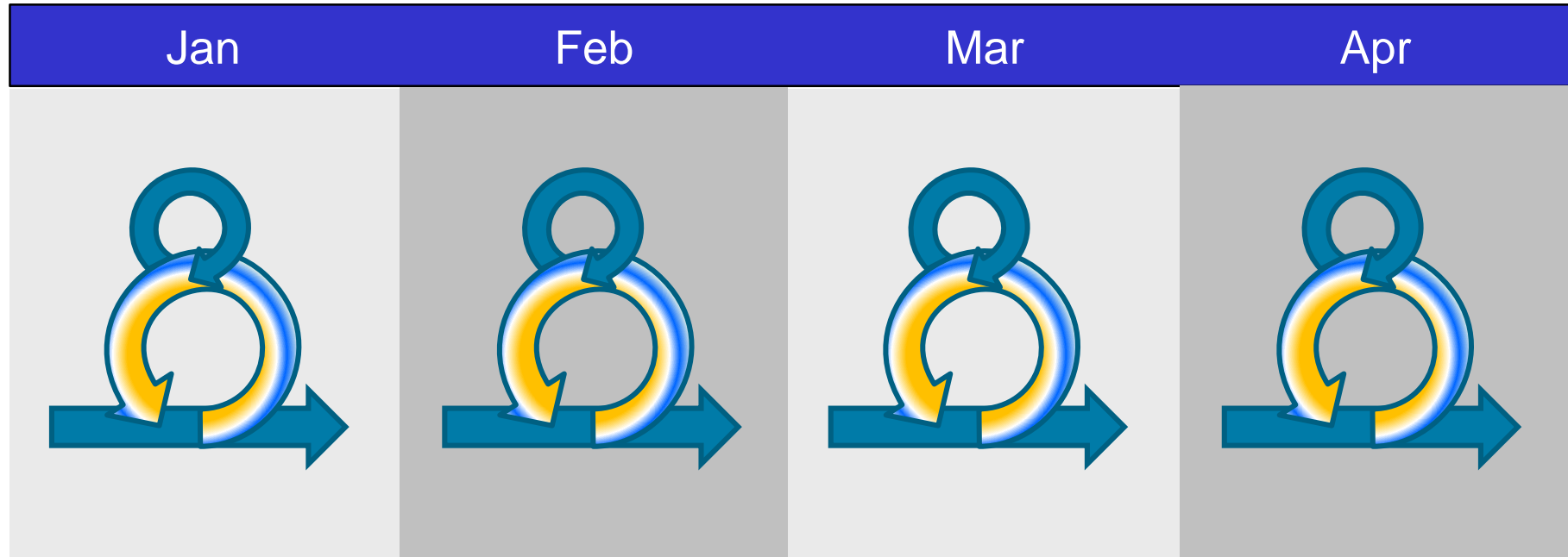
V-Modell XT



Sprint-Modell

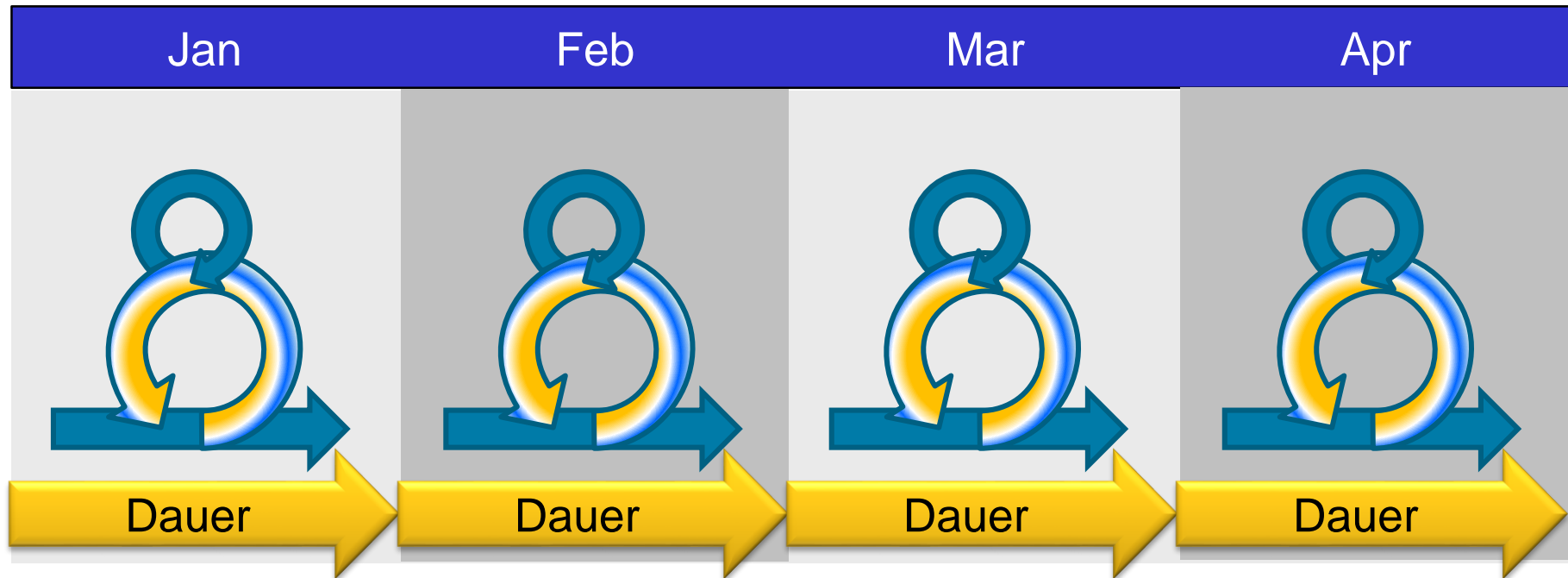


Sprint-Modell



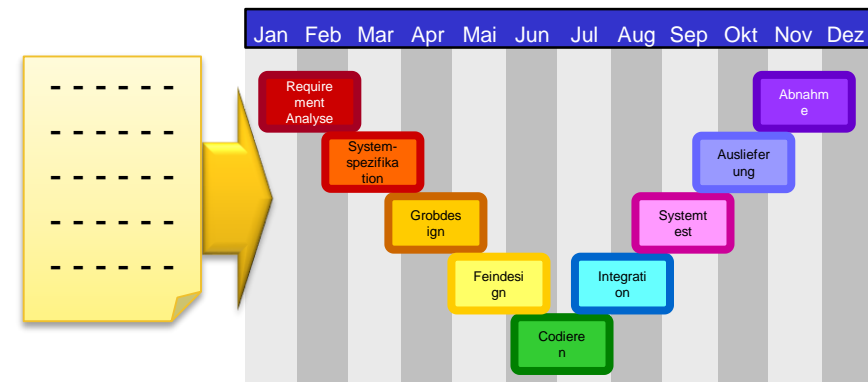
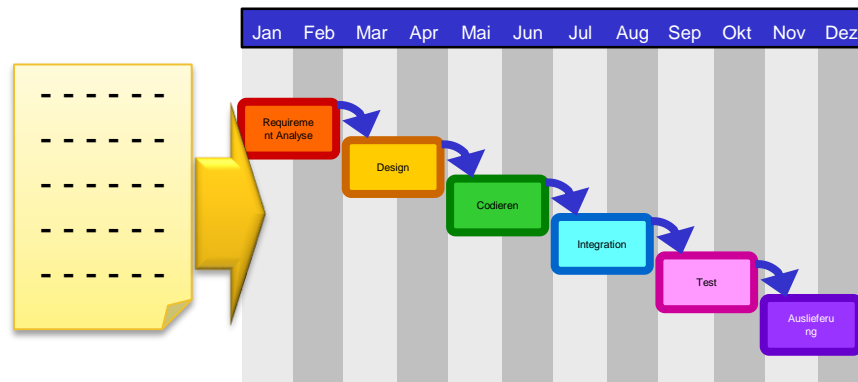
Das Sprint-Modell besteht aus inkrementellen Sprints.

Sprint-Modell



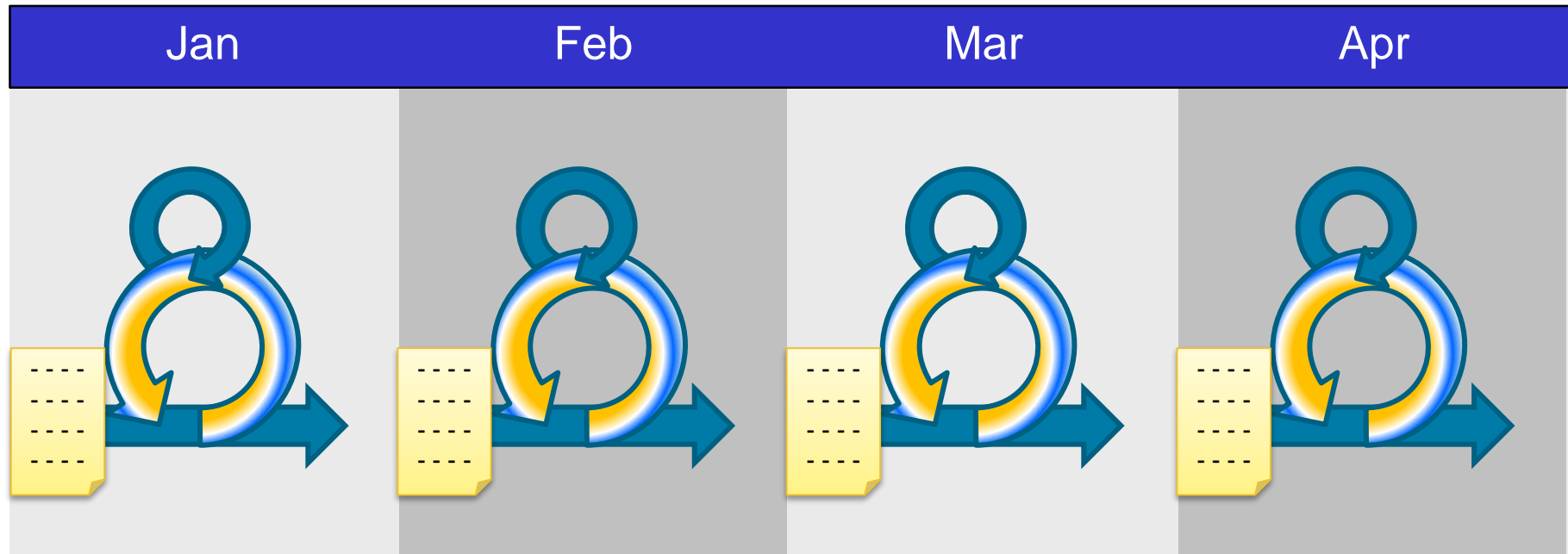
Jeder Sprint dauert gleich lang 1...4 Wochen
(time boxed)

Traditionelle Prozesse



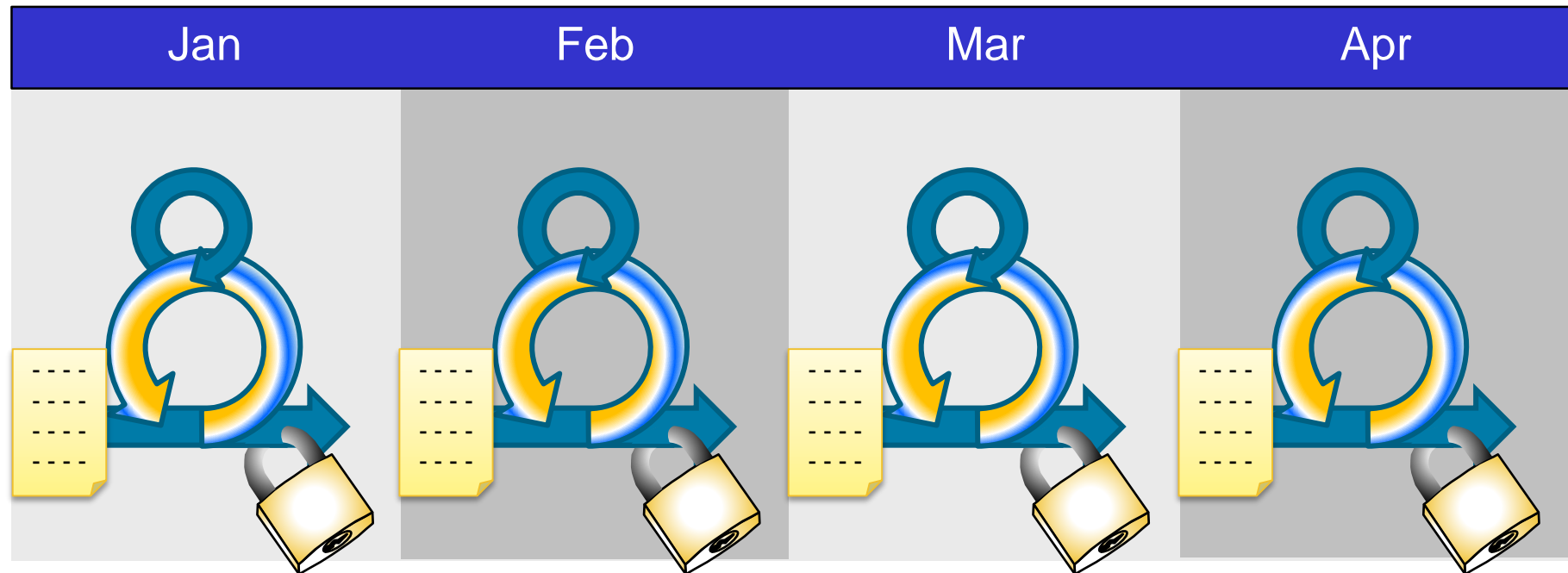
Im traditionellen Prozess wird der Funktionsumfang zu Beginn festgelegt.

Sprint-Modell



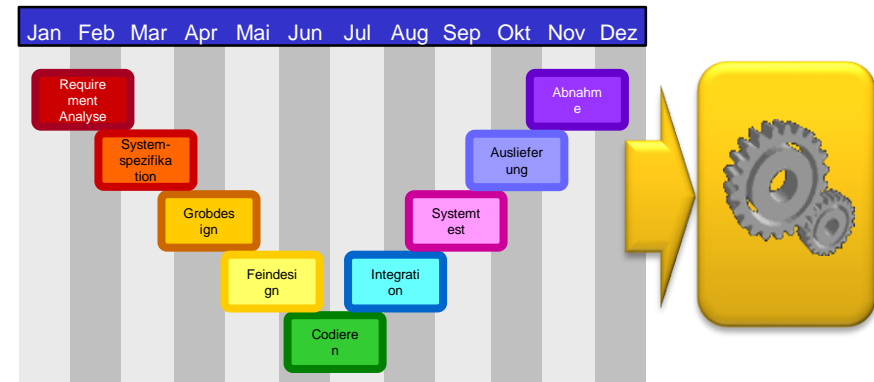
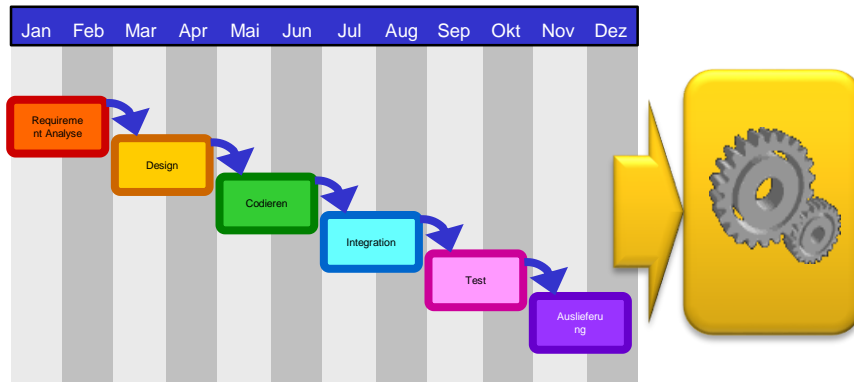
Der inkrementelle Funktionsumfang wird zu Beginn jedes Sprints neu festgelegt

Sprint-Modell



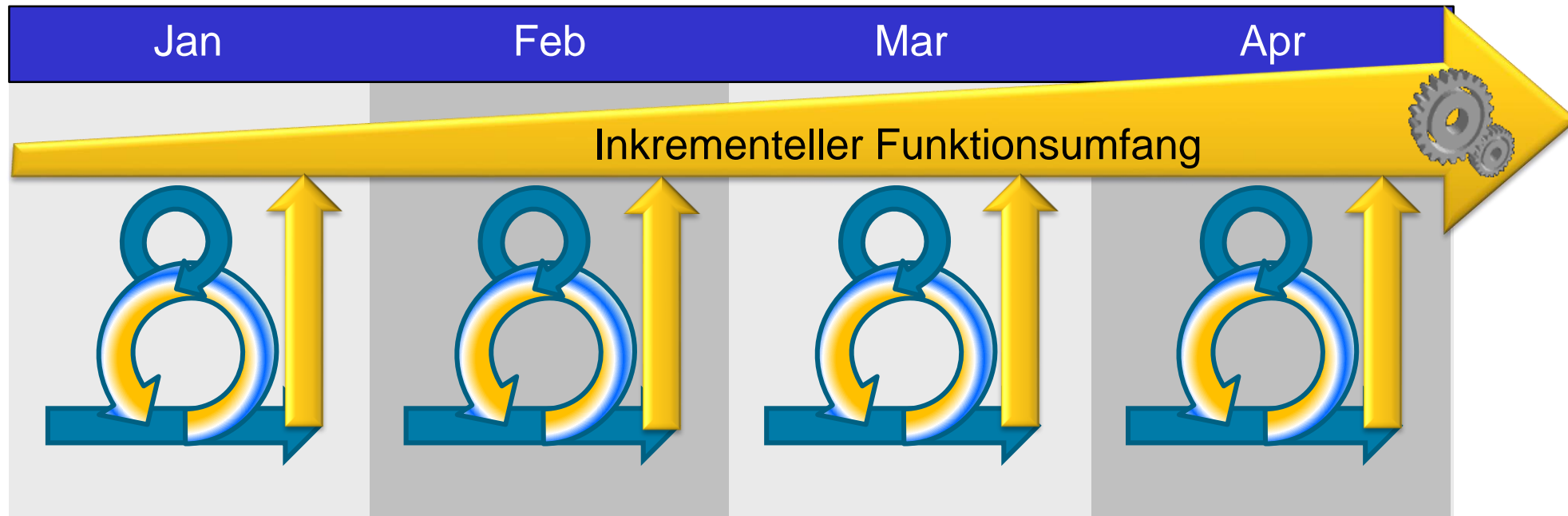
Der inkrementelle Funktionsumfang
bleibt innerhalb eines Sprints stabil

Traditionelle Prozesse

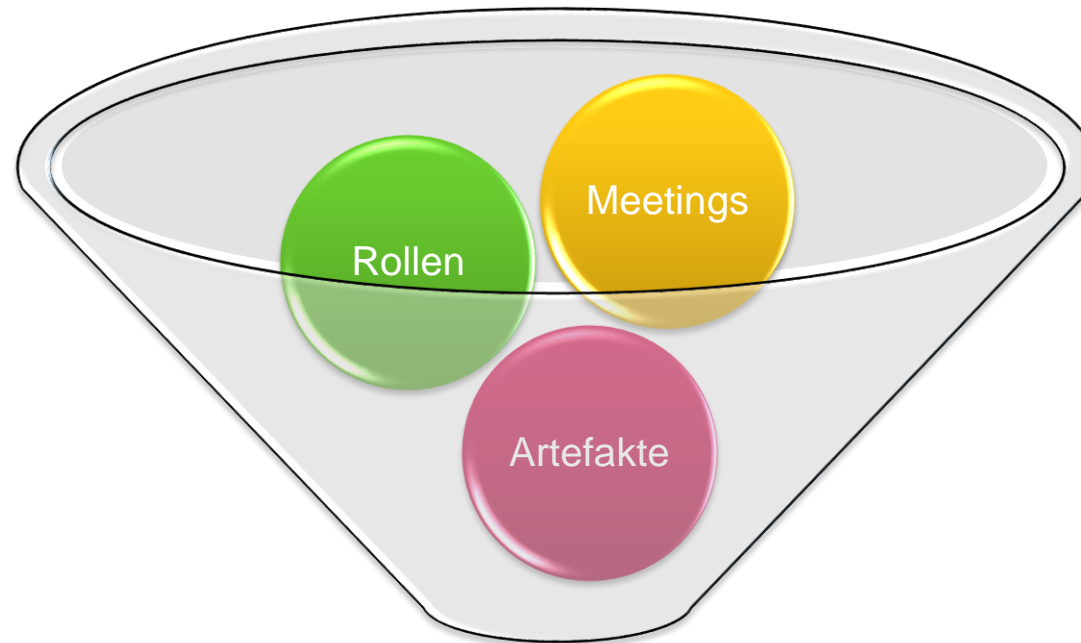


Im traditionellen Prozess steht das Ergebnis erst am Ende bereit.

Sprint-Modell



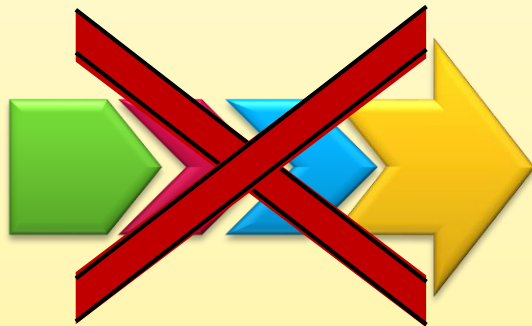
Der inkrementelle Funktionsumfang wächst mit jedem Sprint an



SCRUM

→ SCRUM besteht aus einem Set von Rollen, Meetings und Artefakten, die über Prinzipien und Regeln zusammenspielen.

SCRUM in der Entwicklung von Embedded-Systemen.



SCRUM ist **keine Entwicklungsmethode**, sondern ein **Framework** zur Risikosteuerung, Wertoptimierung und ständigen Neuplanung.

SCRUM in der Entwicklung von Embedded-Systemen.



SCRUM deckt Hindernisse und Dysfunktionen auf und erfordert daher den Willen, diese Funktionsstörungen im Unternehmen zu beseitigen.

SCRUM in der Entwicklung von Embedded-Systemen.



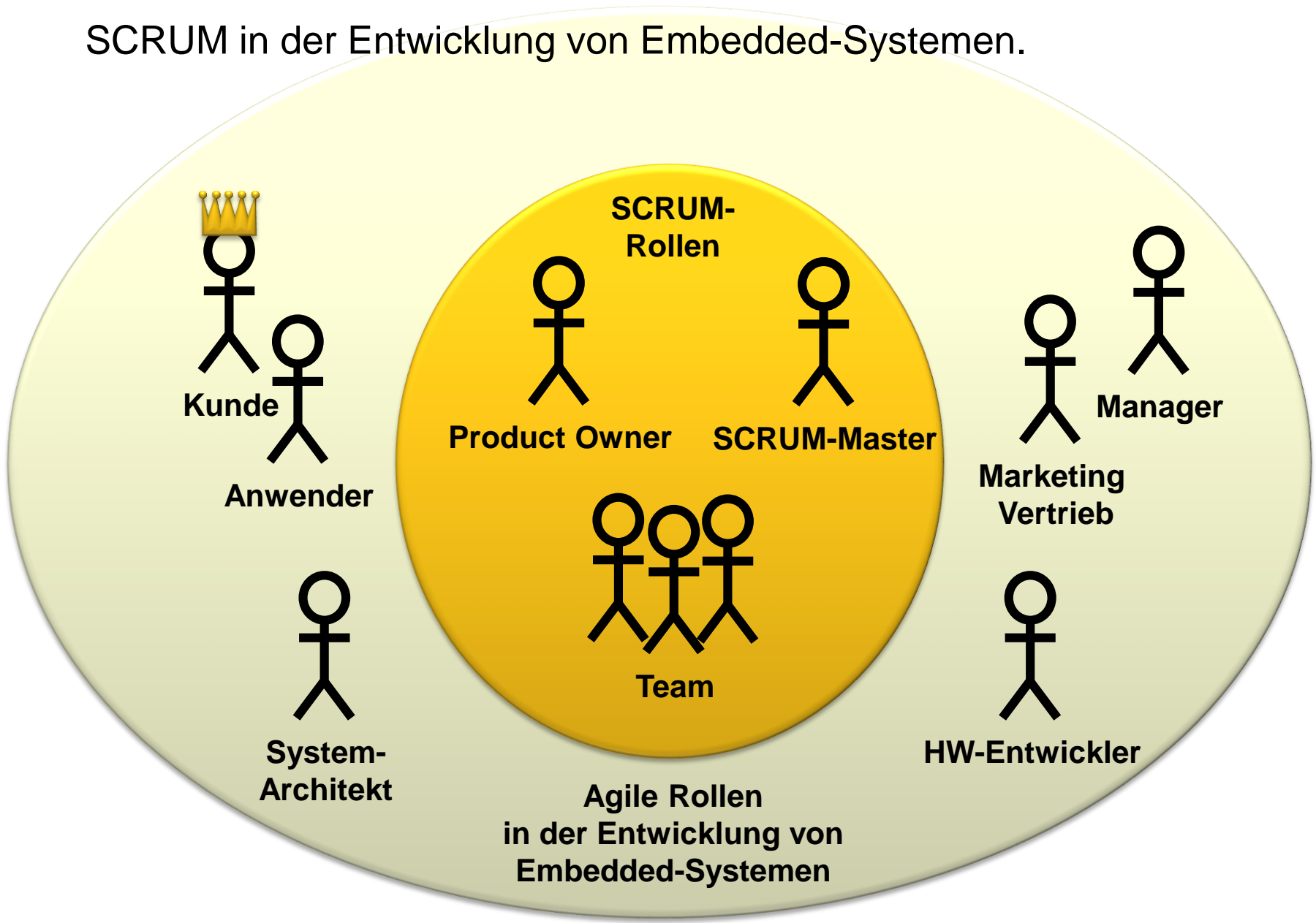
SCRUM setzt auf die Verantwortung des Einzelnen, es fördert dezentrale Entscheidungen.



SCRUM fördert die Motivation, die Menschen bewegt, für ihre Führungskraft zu arbeiten.

➔ Richtig eingesetzt und angepasst, bietet SCRUM Lösungen im Umgang mit steigender Komplexität und verkürzten Produktzyklen

SCRUM in der Entwicklung von Embedded-Systemen.

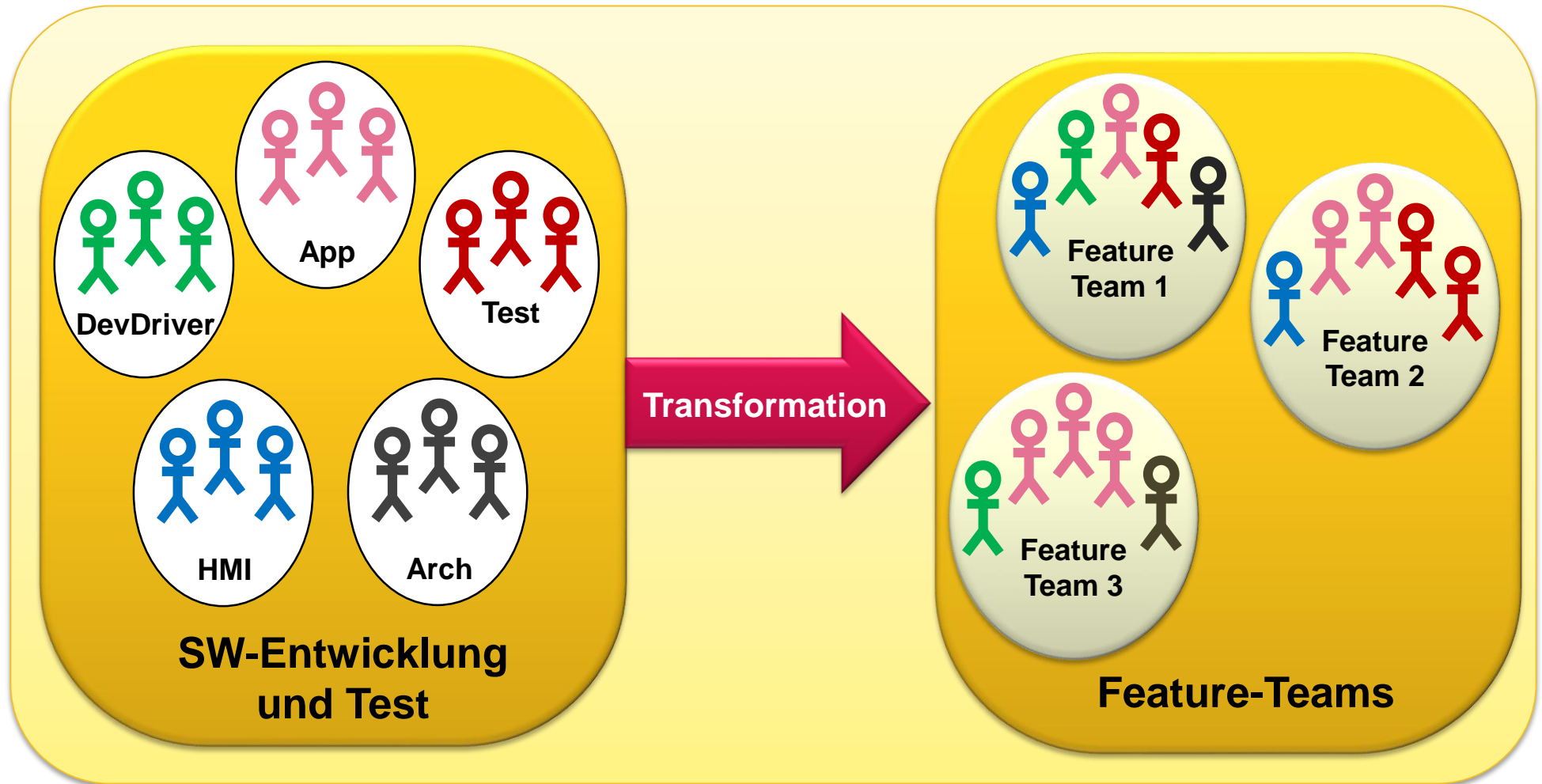


Die Rollen des SCRUM-Teams im agilen Prozess

Das wichtigste Ziel des SCRUM-Teams ist,
eigenverantwortlich das Produkt zu **liefern**.

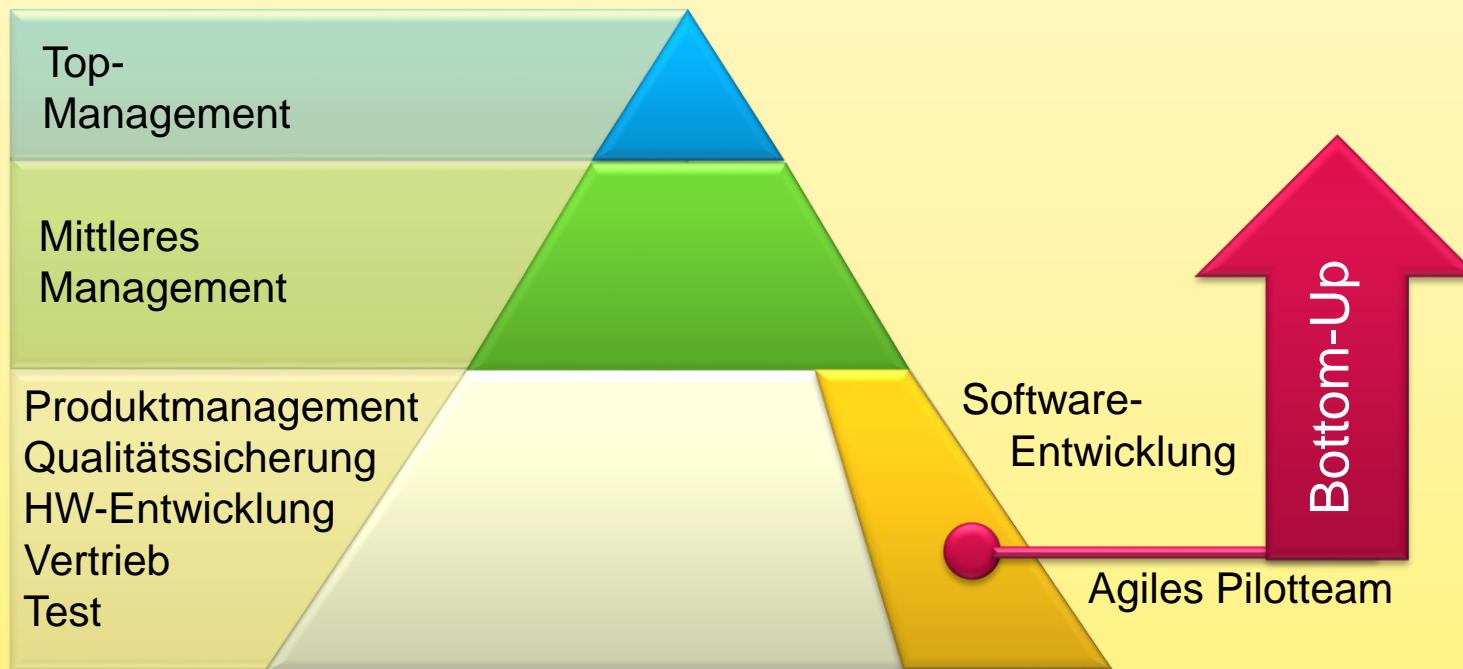


Transformation von Teams aus Spezialisten zu Teams, die einen Mehrwert für den Kunden liefern.

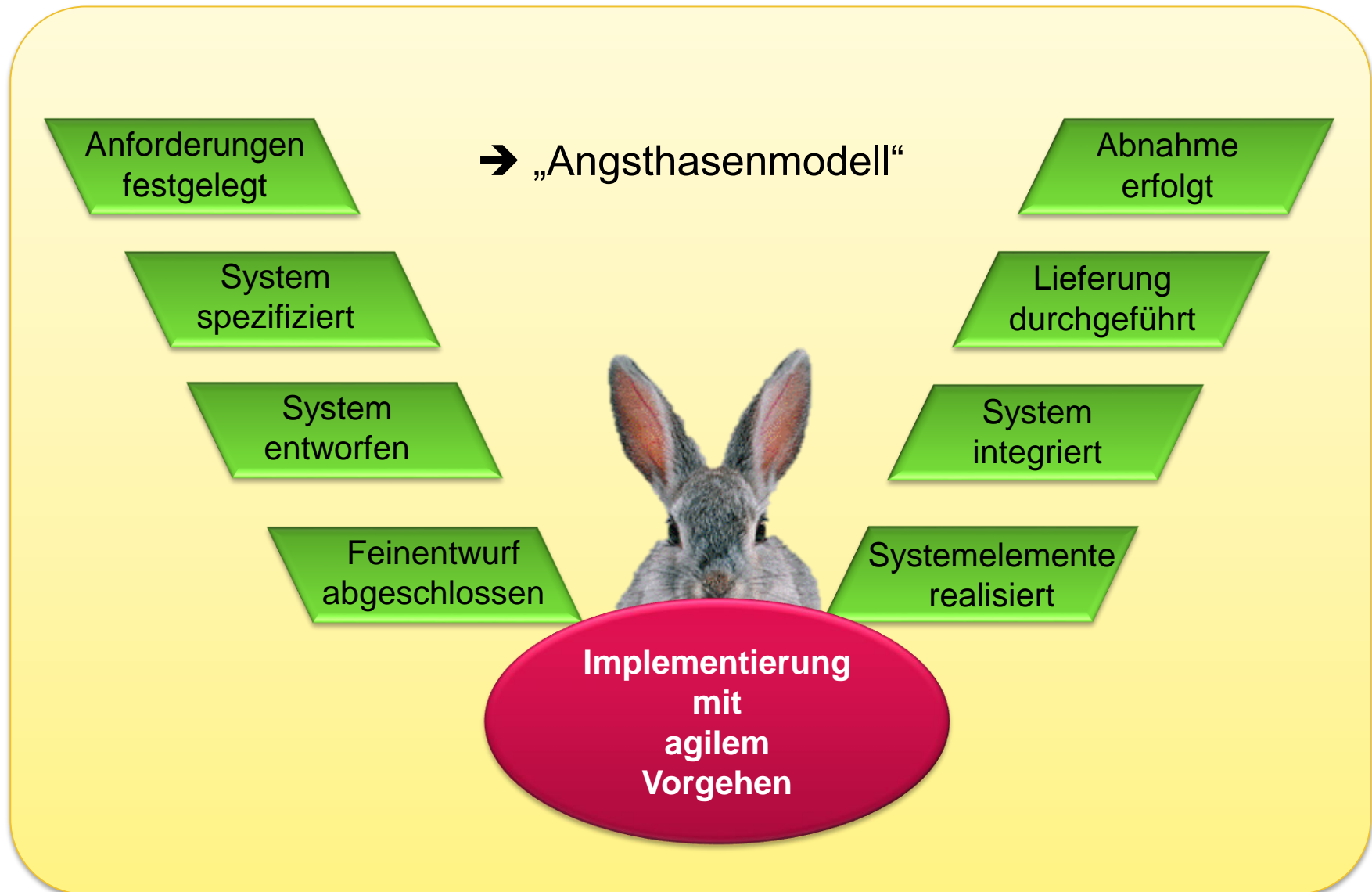


Der übliche Ansatz zum Start agiler Entwicklung

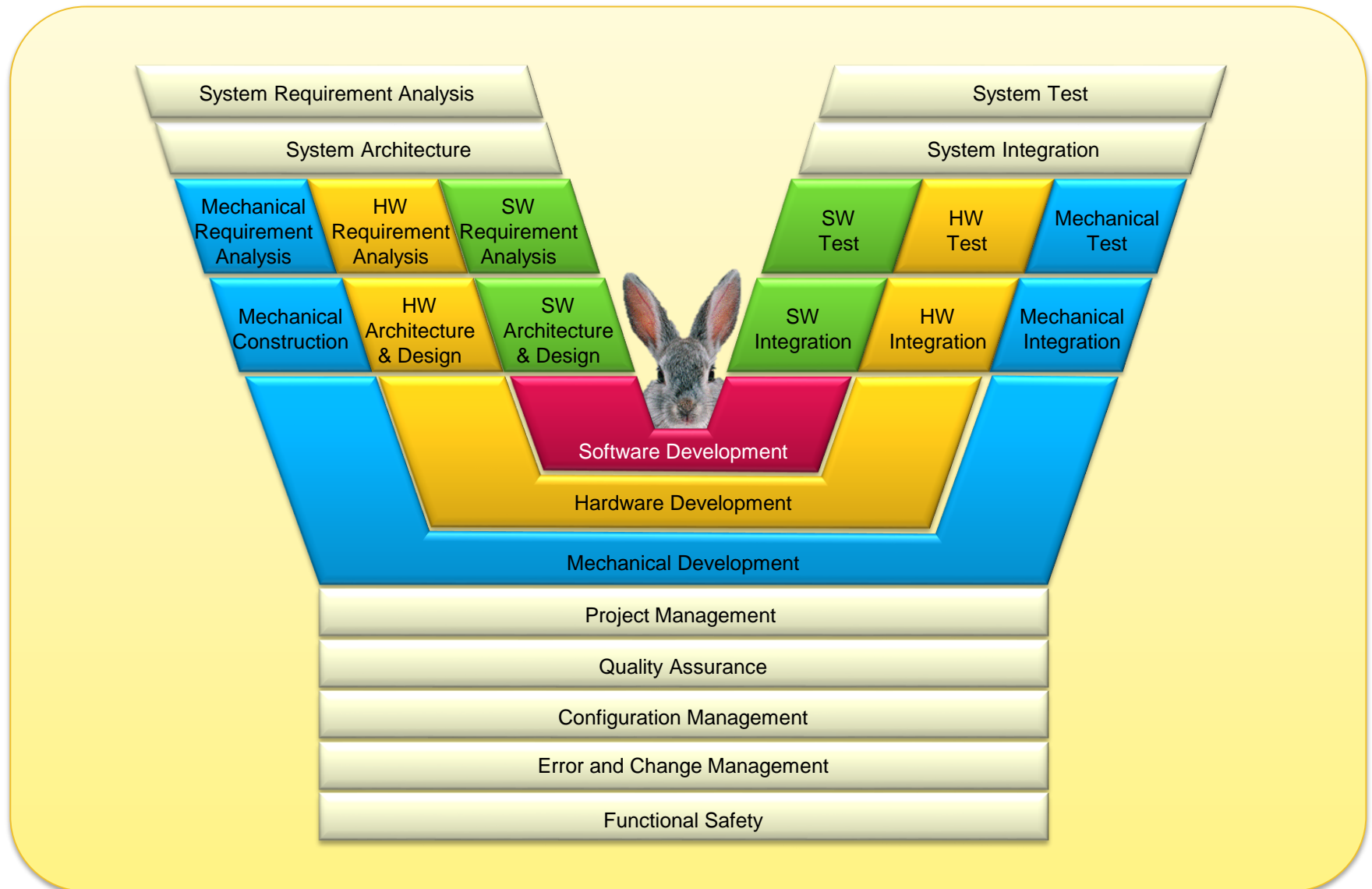
- Risiko minimieren
- Laufende Projekte nicht stören
- Möglichst nur lokale Änderungen
- Nur ein Software-Entwicklungsteam betreffend
- In einem kleinen Pilotprojekt Erfahrung sammeln



Agiles Vorgehen im Bereich der Implementierung im V-Modell



Agiles Vorgehen im Bereich der Software-Implementierung im V-Modell



Ergebnis des Bottom-Up-Ansatzes

Ansatz:

- Traditionelles Produkt- und Projektmanagement
- Agile Entwicklung
- Traditioneller Test

Ergebnisse:

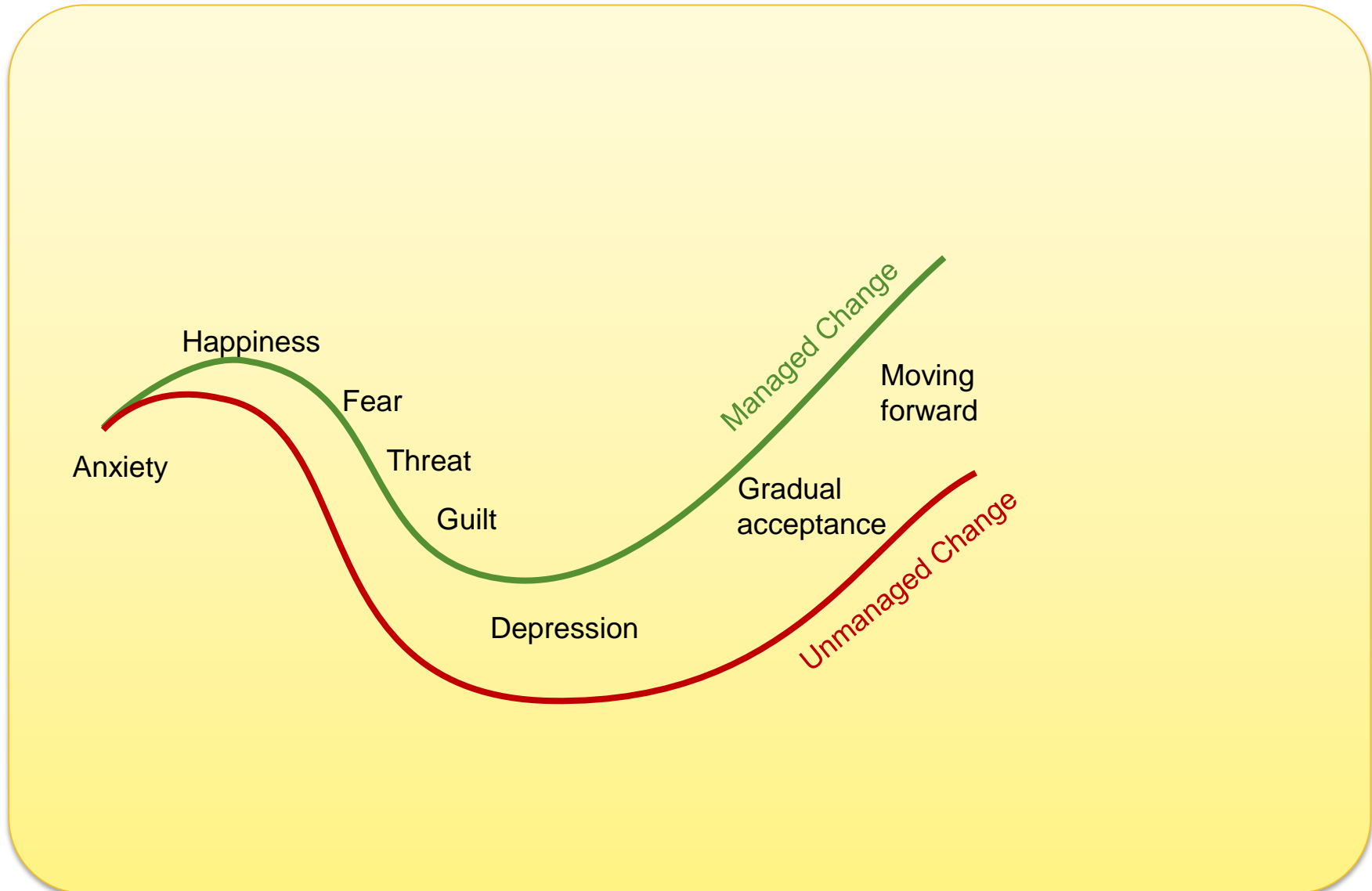
- Keine selbstorganisierenden Teams
- Kein Lean-Ansatz = Ausprobieren-Machen-Dokumentieren-Zyklus
- Kluft zwischen Entwicklung und Test
- Reine Papier-Sprints zu Beginn jedes Projektes und Releases

Konsequenz:

- Kaum agile Vorteile
- Viele Probleme



Agile = viel Neues, viel Lernen, Abweichen vom gewohnten Weg



Quelle: Change Management Kurve nach 'Process of Transition' von JM Fisher.

Ergebnis des Top-Down-Ansatzes

Ansatz:

- Produkt- und Projektmanagement involviert
- Projektgesteuerte Transformation des Entwicklungsprozesses
- Klare Vision

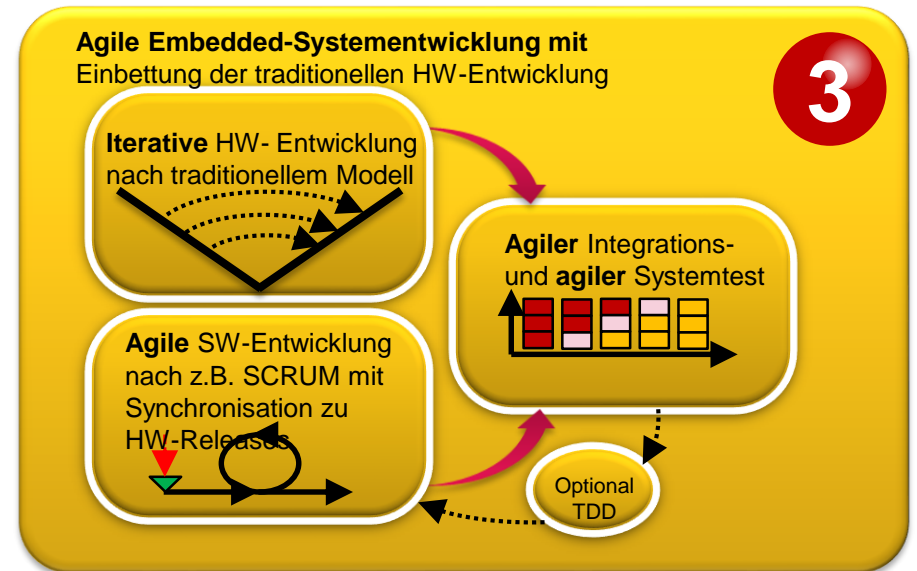
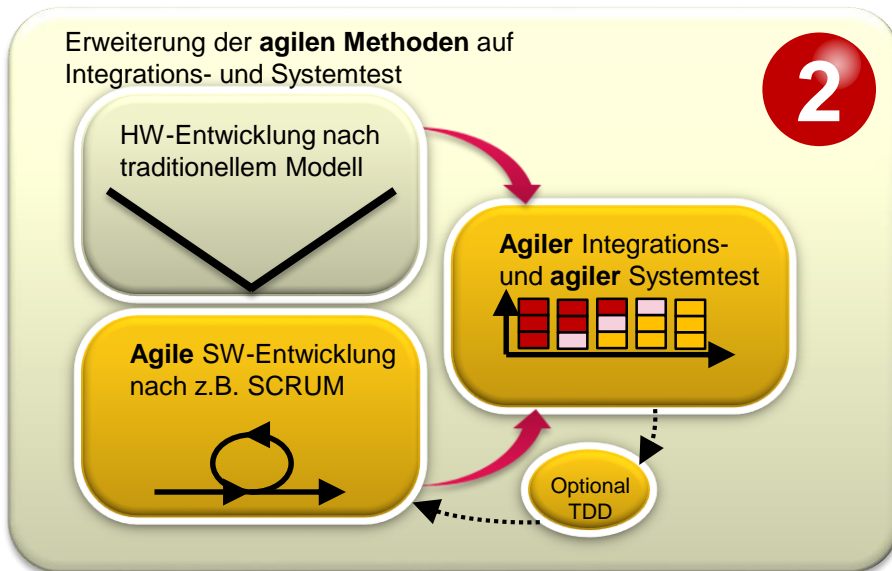
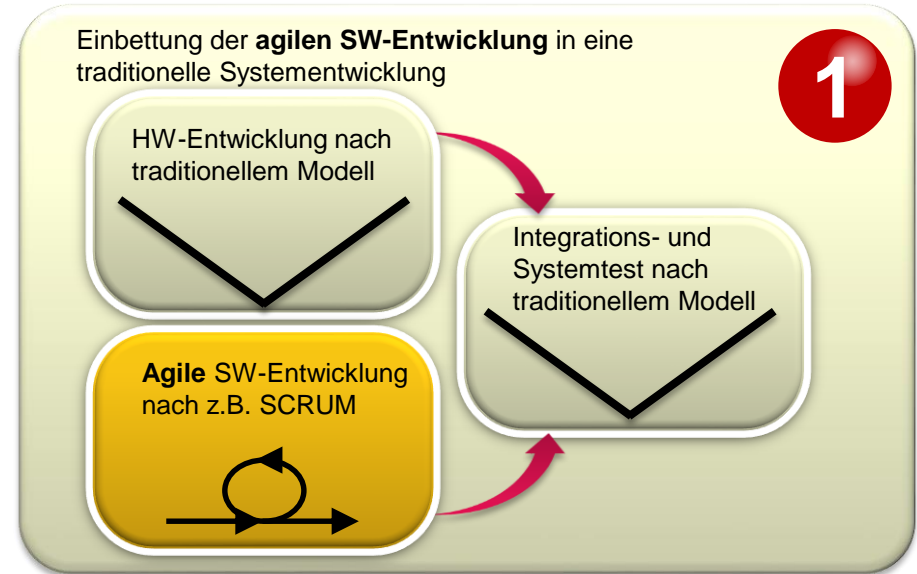
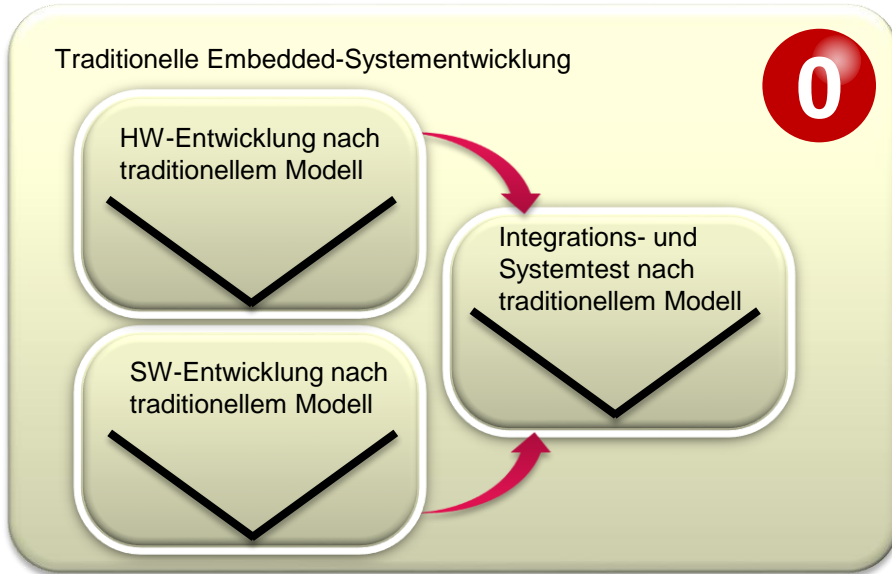
Ergebnisse:

- Selbstorganisierende, hochmotivierte Teams
- Lean and Agile => Innovations-Boost
- Schnelle Reaktion auf Marktveränderungen und Kundenwünsche

Konsequenz:

- Agile Vorteile
- „Trotzdem“ Probleme





Woran ich den falschen Weg erkennen kann:



Woran ich den falschen Weg erkennen kann:

Ich kümmere mich nicht darum, was agile Entwicklung in meiner Organisation bewirkt.

Ich halte so viele Rollen und Aktivitäten wie möglich unberührt.

Ich betrachte ausschließlich die Entwicklung.

Woran ich den falschen Weg erkennen kann:

Ich kümmere mich nicht darum, was agile Entwicklung in meiner Organisation bewirkt.

Ich halte so viele Rollen und Aktivitäten wie möglich unberührt.



Ich betrachte ausschließlich die Entwicklung.

Woran ich den falschen Weg erkennen kann:

Ich entwickle ohne Prozessanpassungen streng nach SCRUM-Lehrbuch.

Ich entwickle keinen maßgeschneiderten Entwicklungsprozess.

Ich lasse alle existierenden Verbesserungsprojekte und Prozesse außer Acht.

Woran ich den falschen Weg erkennen kann:



Woran ich den falschen Weg erkennen kann:

Ich plane nicht mehr als 2-4 Wochen voraus.

Ich betreibe 3-4 Monate „agile Analyse“, damit ich nicht falsch loslege

Alle müssen SCRUM anwenden, ohne Ausnahme.

Die grobe System-Architektur kristallisiert sich während der Entwicklung heraus.

Woran ich den falschen Weg erkennen kann:

Ich plane nicht mehr als 2-4 Wochen voraus.

Ich betreibe 3-4 Monate „agile Analyse“, damit ich nicht falsch loslege



Alle müssen SCRUM anwenden, ohne Ausnahme.

Die System-
struktur kristallisiert
sich während der
Entwicklung heraus.

Woran ich den falschen Weg erkennen kann:

Meine Entwickler sind unerfahren und jung. Agile Entwicklung ist die perfekte Lösung.

Ich erlaube keine kulturellen Besonderheiten.

Ich fördere einzelne Mitarbeiter und nicht komplette Teams.

Selbstorganisierende Teams lösen alle Probleme alleine.

Woran ich den falschen Weg erkennen kann:

Meine Entwickler sind unerfahren und jung. Agile Entwicklung ist die perfekte Lösung für sie.

Ich erlaube keine kulturellen Besonderheiten.



Ich fördere einzelne Mitarbeiter und nicht komplette Teams.

Selbst Teams lösen Probleme alleine.

Woran ich den falschen Weg erkennen kann:

Ich bilde meine Leute nicht aus.

Ich gebe meinen Managern keinen Trainingsupport.

In agilen Projekten muss man alle Erfahrungen selbst sammeln.

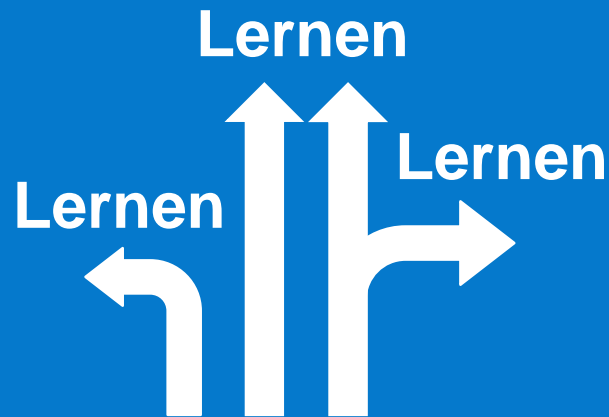
Ich suche keine Unterstützung bei externen Partnern und Consultants.

Woran ich den falschen Weg erkennen kann:

Ich bilde meine Leute nicht aus.

In agilen Projekten muss man alle Erfahrungen selbst sammeln.

Ich gebe meinen Managern keinen Trainingssupport



Suche keine Unterstützung bei externen Partnern und Consultants.

Danke



Für Interessierte:

<http://www.microconsult.de/um/agile-dev>

<http://www.microconsult.de/um/scrum>

<http://www.microconsult.de/um/agile-tdd>

MicroConsult bietet Trainings & Coaching zu agilen Methoden an:

Agile Entwicklung von Embedded-Systemen

Agile Entwicklung ist die erfolgreichste Methode, um immer schneller werdende Marktveränderungen und Kundenforderungen erfolgreich erfüllen zu können. Agile Entwicklung erfordert ein Umlernen, nicht nur in der Entwicklung, sondern auch im Produkt- und Projektmanagement, im Vertrieb und in der Unternehmensführung. Die Einführung der agilen Entwicklung von Embedded-Systemen beinhaltet im komplexen Zusammenspiel von Hardware und Software besondere Herausforderungen an die agile Unternehmenstransformation. In diesem Training werden Ihnen neben dem nötigen theoretischen Wissen auch praktische Umsetzungswege und Tools an die Hand gegeben, mit deren Hilfe Sie schrittweise und kontrolliert die agile "Lebensweise" in Ihr Unternehmen einführen.

Mehr auf: www.microconsult.de/um/agile-dev

SCRUM in der agilen Entwicklung von Embedded-Systemen

SCRUM ist der weitverbreitetste Vertreter der agilen Software-Entwicklungsmethoden. Die Einführung der agilen Entwicklung von Embedded-Systemen beinhaltet besondere Herausforderungen. Es genügt daher nicht, die SCRUM-Methodik aus der IT-Welt einfach auf Embedded-Software anzuwenden. Erst eine Umstellung des gesamten Entwicklungsprozesses inklusive des Systemtests auf agile Methoden sowie die enge Kopplung mit der Hardwareentwicklung führt zum gewünschten Erfolg. In diesem Training wird daher neben den SCRUM-Grundlagen speziell auf die Besonderheiten des komplexen Zusammenspiels von Hardware und Software und die Umstellung des gesamten Entwicklungsprozesses inklusive des Integrations- und Systemtests und des übergeordneten Systemprozesses auf agile Methoden eingegangen. Die verschiedenen Rollen und Meetings nach SCRUM werden als praktische Übungen durchgeführt, um Sicherheit im Umgang mit den ungewohnten Methoden zu erlangen.

Mehr auf: www.microconsult.de/um/scrum

Agiles Testen und Test Driven Development von Embedded-Systemen

Das agile Testen von Embedded-Systemen ist die konsequente Erweiterung der AGILE-Methodik auf das gesamte Embedded-System. Analog zum agilen Entwickeln bedeutet der agile Testansatz einen Paradigmenwechsel. Unter Test Driven Development werden Tests erstellt und durchgeführt, bevor die eigentliche Komponente entwickelt wurde. Die Tests wachsen inkrementell mit dem System. Durch die kontinuierliche Testautomatisierung und Testwiederholung werden Fehler sehr früh im Entwicklungsprozess entdeckt und sind leichter lokalisierbar. In diesem Training werden Ihnen neben dem nötigen theoretischen Wissen auch praktische Umsetzungswege vermittelt, mit deren Hilfe Sie agiles Testen und Test Driven Development in die Entstehung Ihrer Embedded-Systeme einführen.

Mehr auf: www.microconsult.de/um/agile-tdd