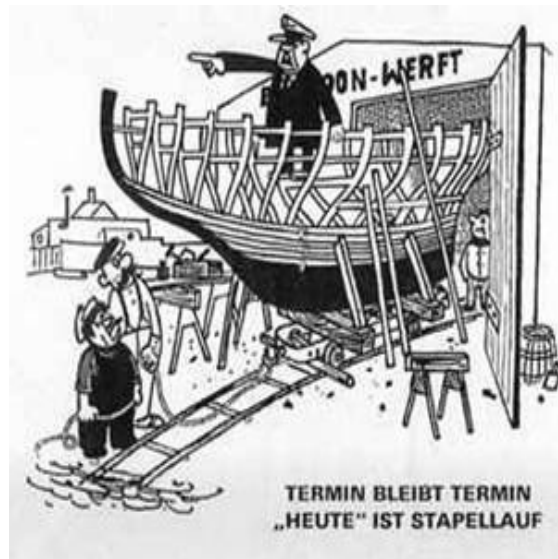


Moving to an Agile Culture

There is no silver bullet



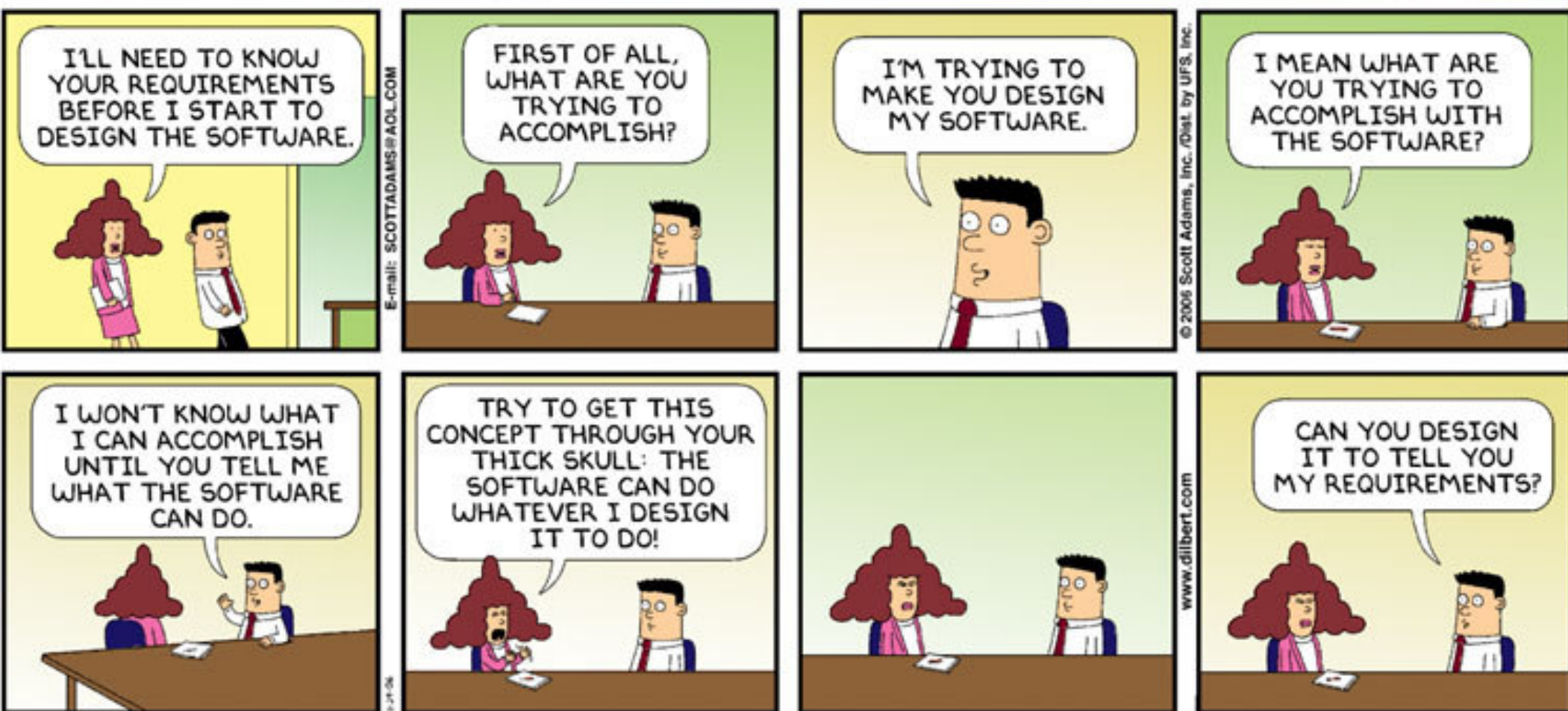
Was bedeutet Nicht-Agilität ?

Was bedeutet Agilität ?

Der Weg zur Agilität

Software Entwicklung ist ein kreativer Prozess

Der Fehlglaube an die Industrialisierung der Software Erstellung



Industrialisierung der Software Erstellung

Ein typisches Szenario



- Definition eines Produktionsprozesses
- Detaillierte Handlungsanweisungen für die im Produktionsprozess definierten Rollen
- Beschränkung der Freiheitsgrade für Projekte aufgrund von Standardisierungsbestrebungen
- Einführung von Kontrollmechanismen zur Überwachung des definierten Produktionsprozesses

Beispiele...



Moving to an Agile Culture
Folie 4
19. November 2009

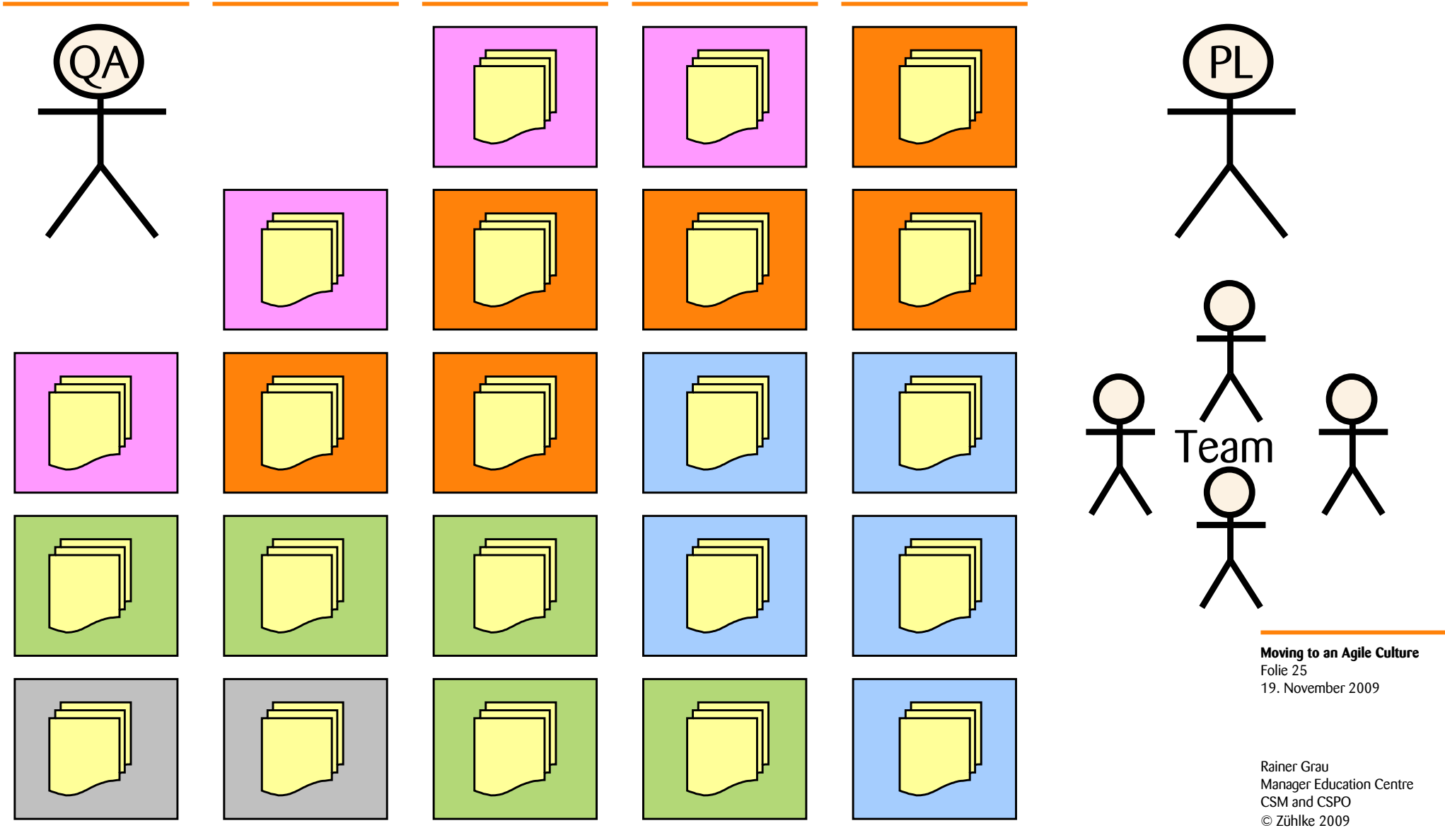
Rainer Grau
Manager Education Centre
CSM and CSPO
© Zühlke 2009

Identifikation von „Nicht-Agilität“



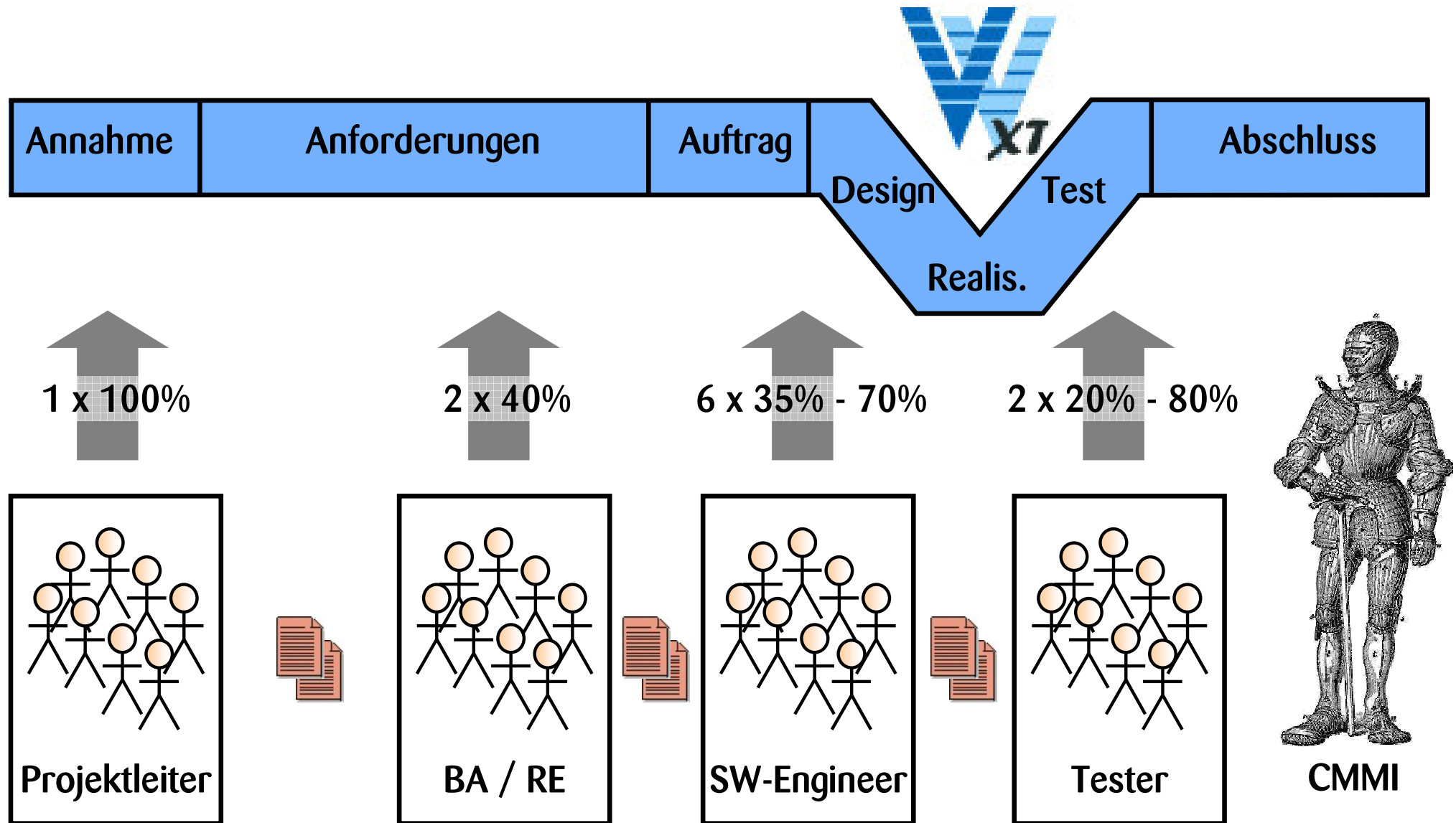
- Bewahrung von Wissen ausschliesslich in Dokumenten
- Sicherung des Vorgehens über exzessive Prozessdefinition
- Abbildung von Rollen in Jobprofilen → Pool Organisation
- Kommunikation der Rollen über Dokumente
- (Irr-)Glaube an die „Austauschbarkeit“ von Ressourcen, sprich Personen
 - 3 Stück Architekt, 1 Stück RE, 8 Stück Entwickler...
 - 2 Stück Manager ...?
- Etablieren von Kontrollmechanismen
 - Verifikation von Prozessvorgehen und Dokumentenqualität
 - Ausschliesslich über eigene Q-Organisation
 - Anhand von formalen Reviews und (weiteren) Dokumenten

Die nicht agile Form eines Projekt Setup

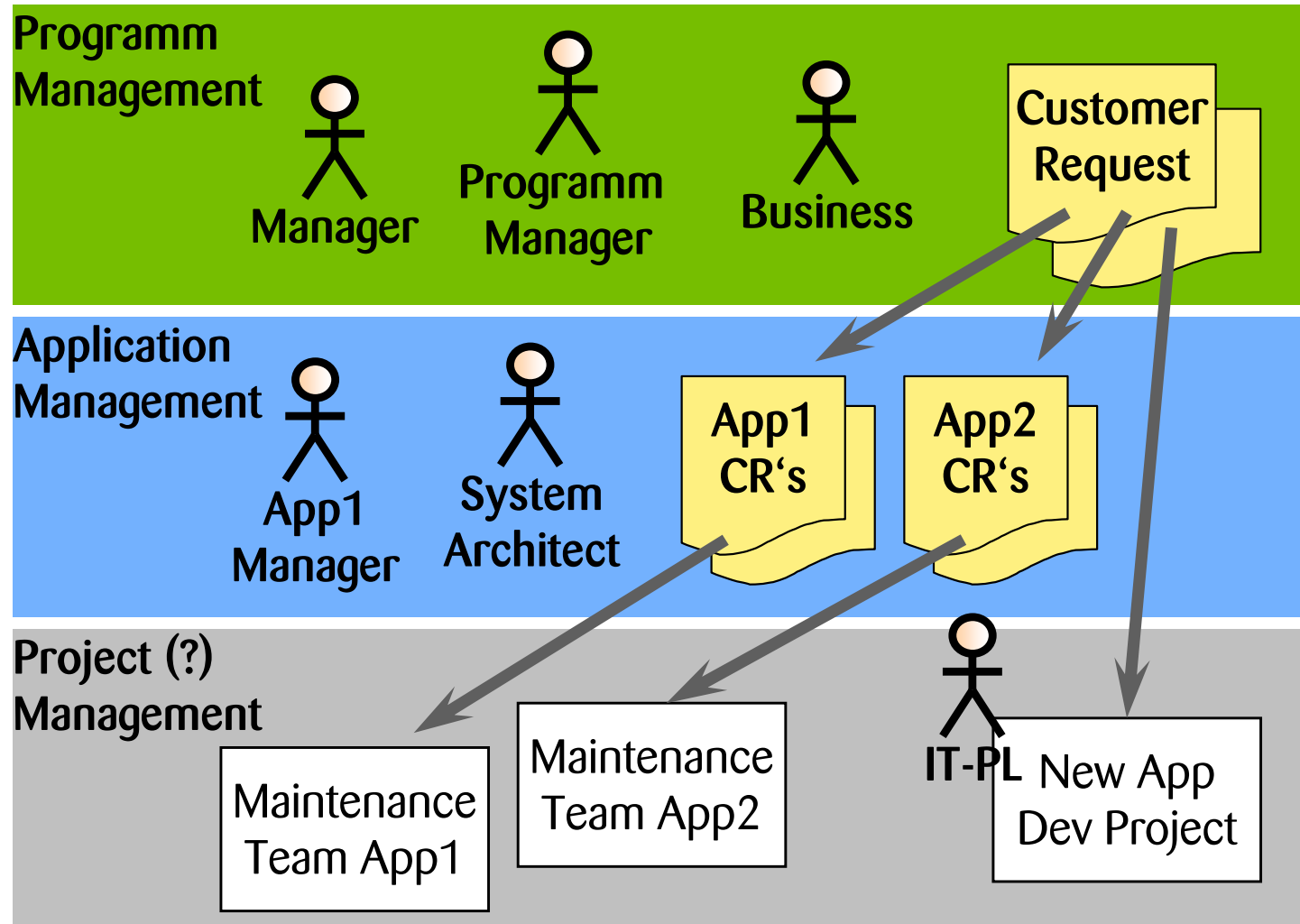


Szenario: Grossunternehmen Service Delivery mit eigener IT-Landschaft

zühlke
empowering ideas



Typischer Informationsfluss in einer nicht agilen Organisation



Agilität als Gegenbewegung zur Industrialisierung in der Software Erstellung

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

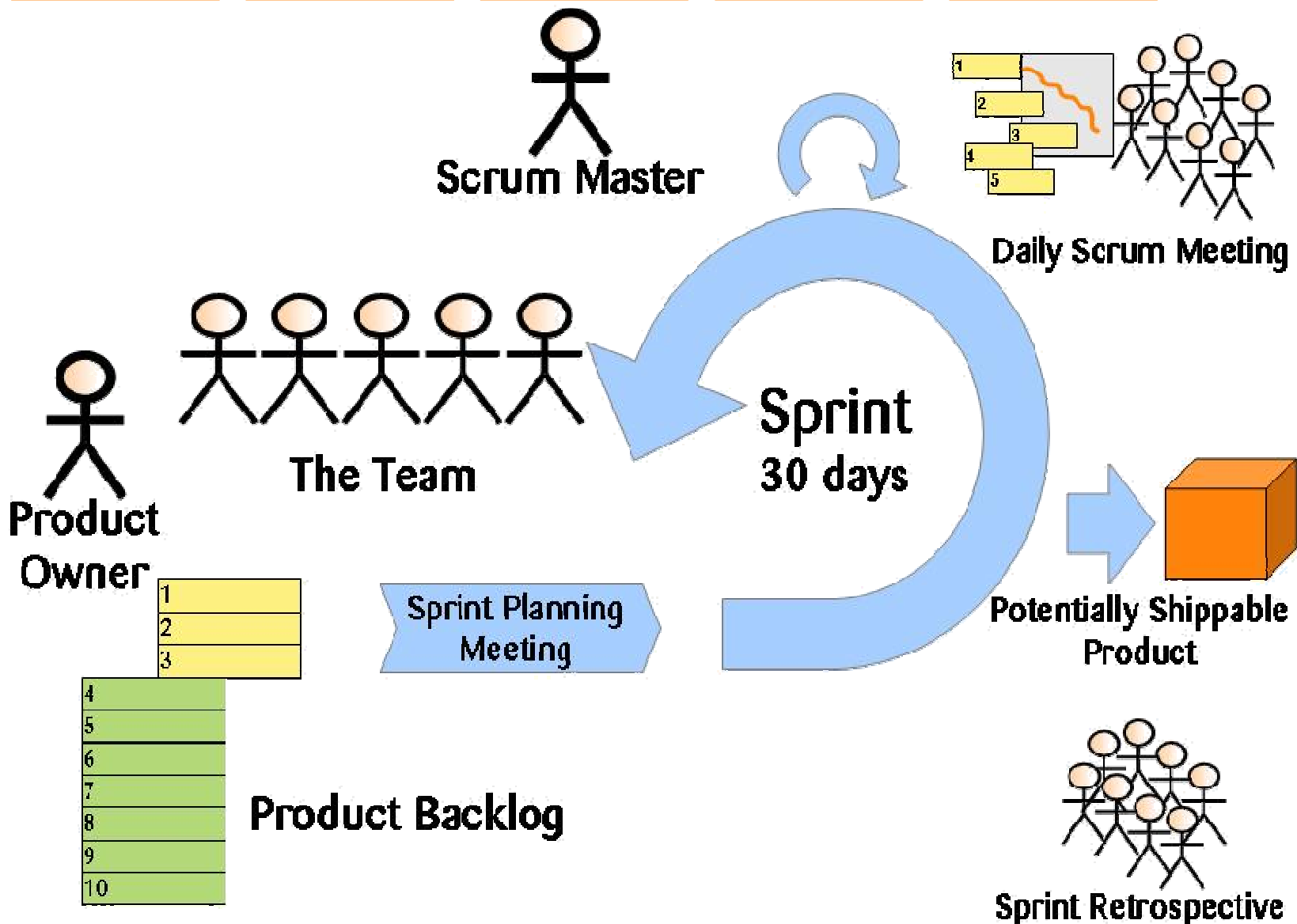
That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Agiles Management am Beispiel Scrum



Ergänzung um SW Engineering Praktiken

Beispiel Extreme Programming (XP)



Autor von XP ist wesentlich Kent Beck (Chrysler)

■ Definition

- Planung in den Abschnitten Release, Iteration, Tag
- Fokus auf Addressierung von Risiken und Kundennutzen
- Definiert über
 - 5 Werte,
 - 14 Prinzipien
 - Best Practices

■ Team

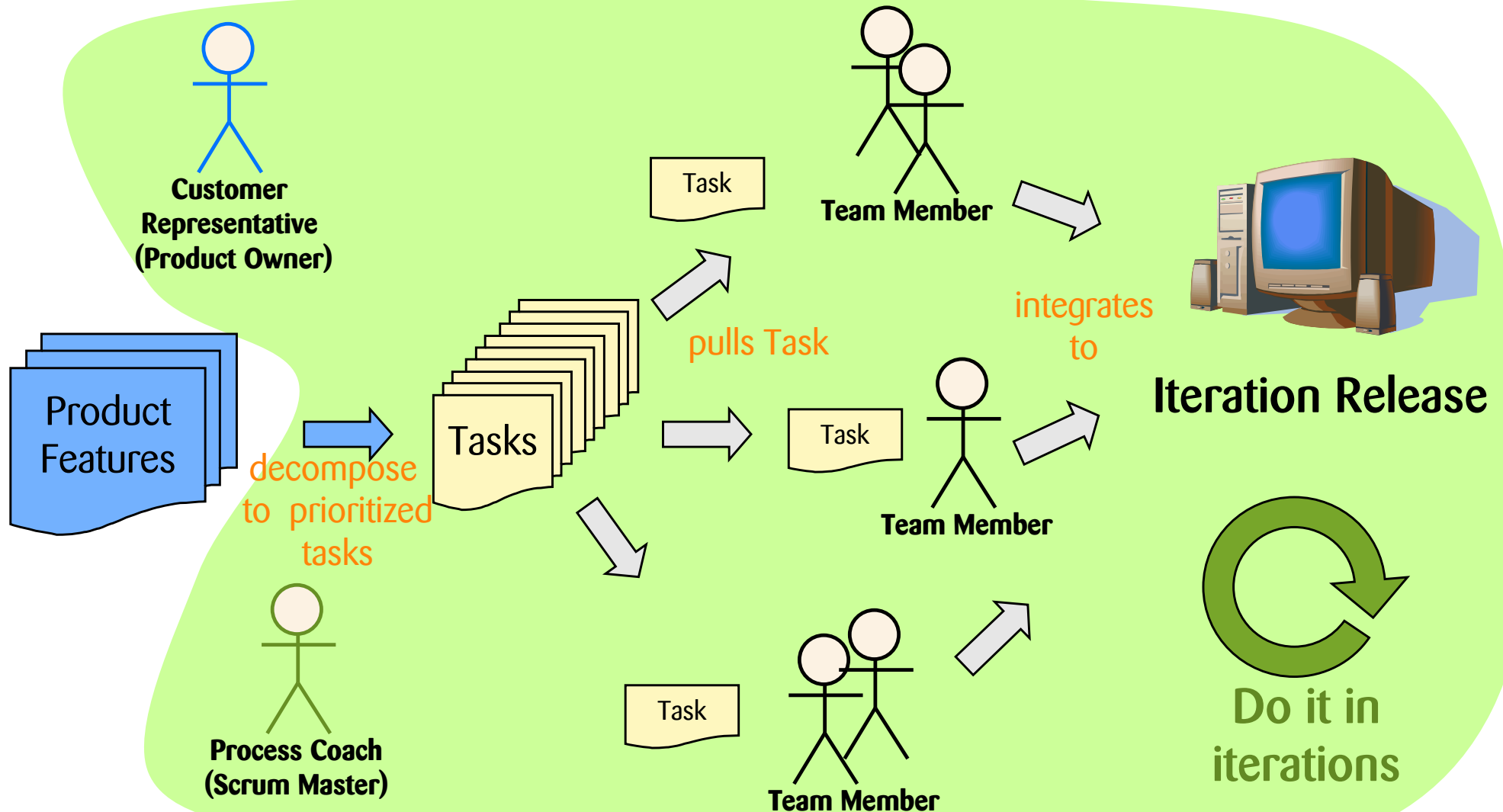
- Ähnlich einem Scrum Team, jedoch mit Projektmanager

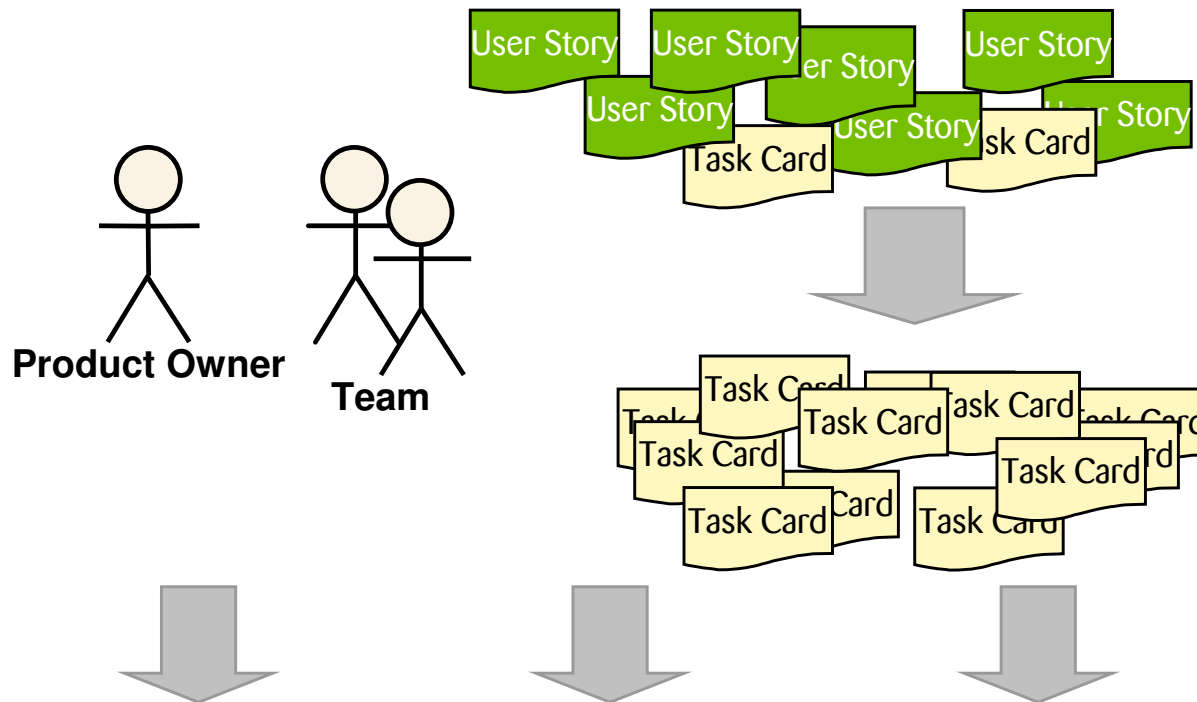
■ Best Practices

- Kommunikation
- Mut
- Kollektives Eigentum
- Pair-Programming
- Integration
- Testgetriebene Entwicklung
- Kundeneinbeziehung
- Refactoring
- Keine Überstunden
- Iterationen
- Stand-up-Meeting
- Dokumentation
- Metapher
- Team
- Standards
- Qualität



Aufstellung eines agilen Teams (allgemeine Terminologie)





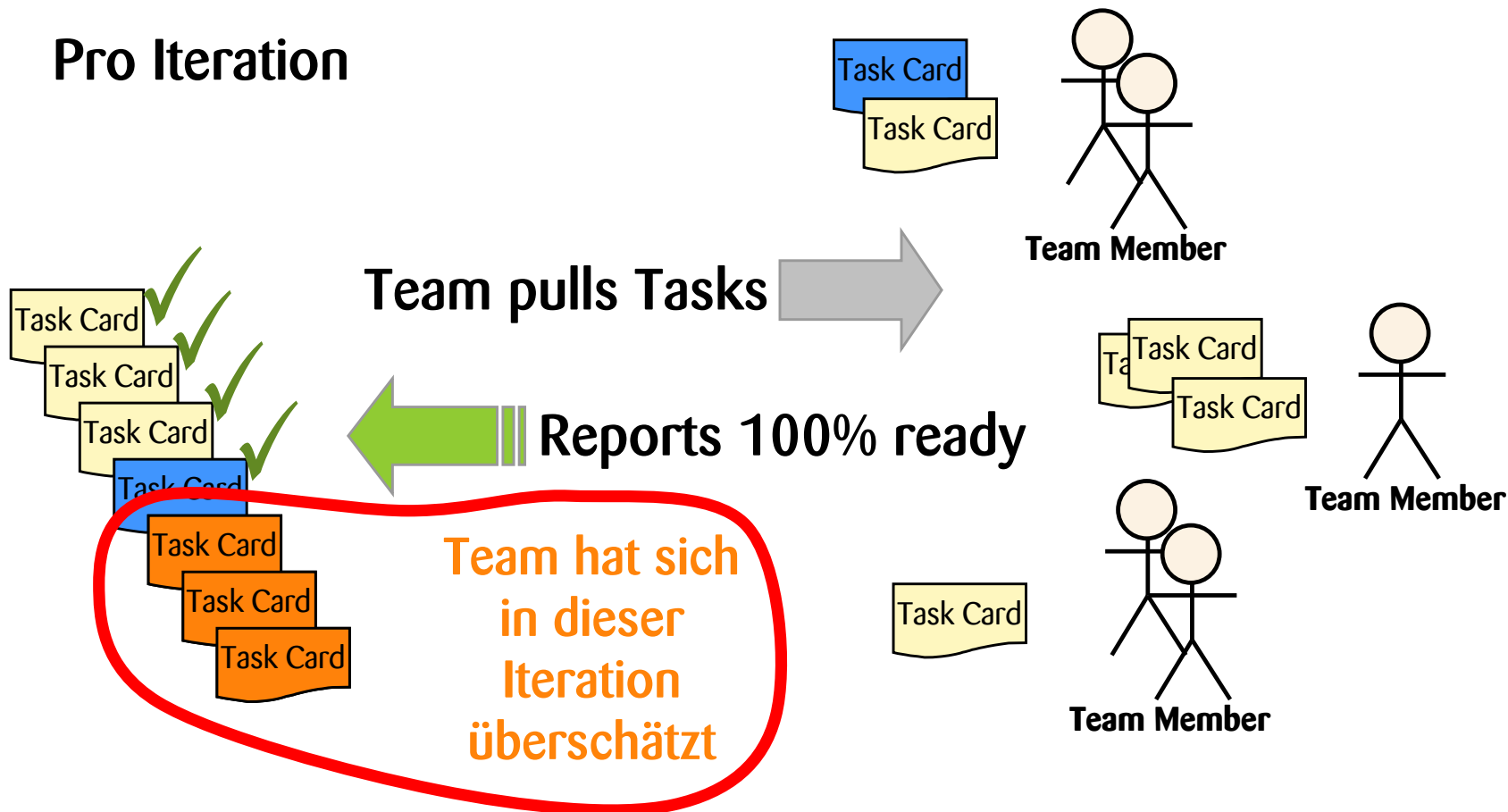
**Definition von Scope
und Priorität des Produkts
(Ebene Product Backlog)**

**Herunterbrechen auf
sinnvollen Aufgabenumfang**

**Definition des Scopes
einer einzelnen Iteration
(Ebene Iterations-Backlog)**

Kontrolle des Projektfortschritts über (Scrum) Burndown Chart

Pro Iteration



– aufgrund von Bugs, Architektur und technischen Gründen



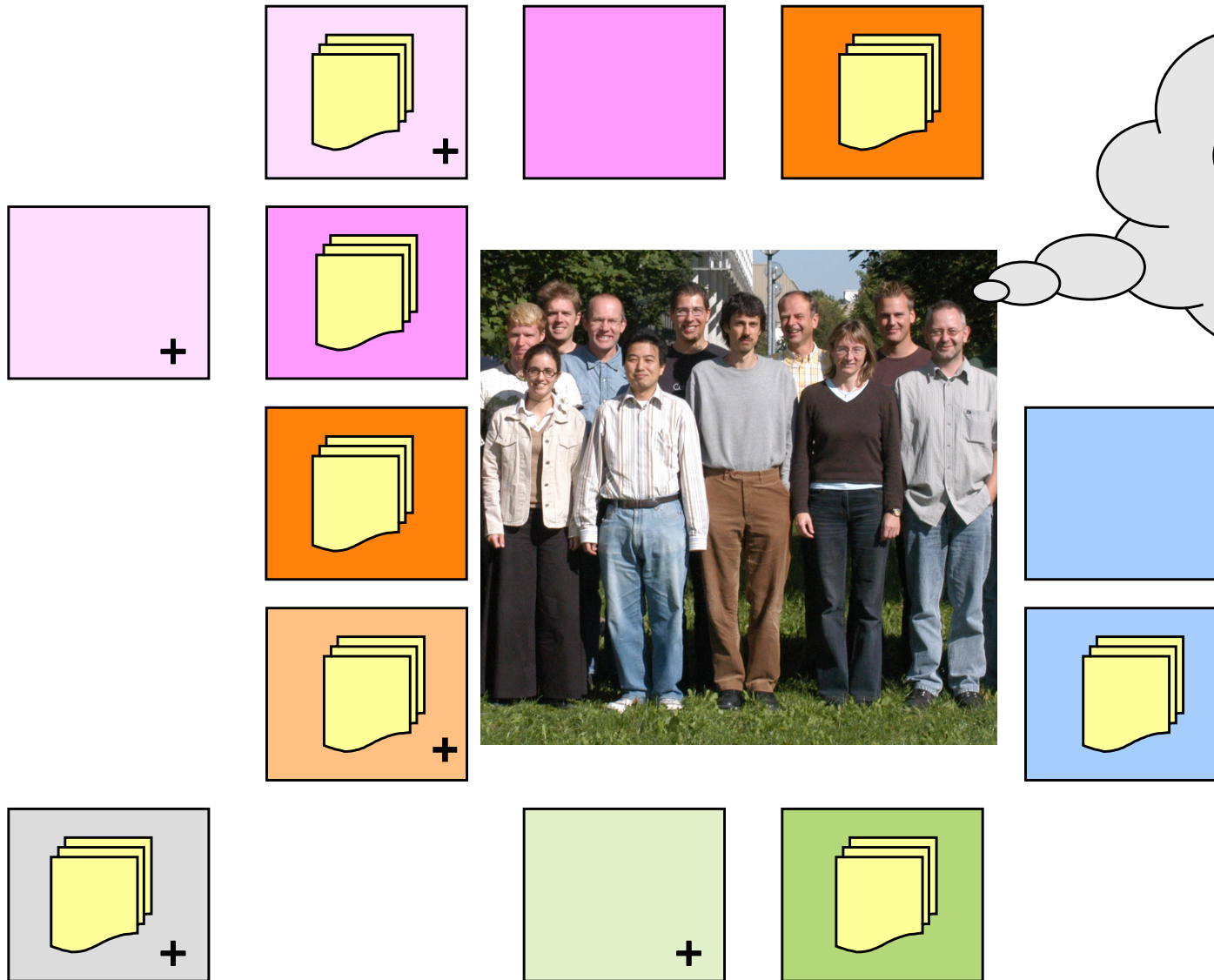
– nicht zu 100% abgearbeitete Tasks

Identifikation von „Agilität“



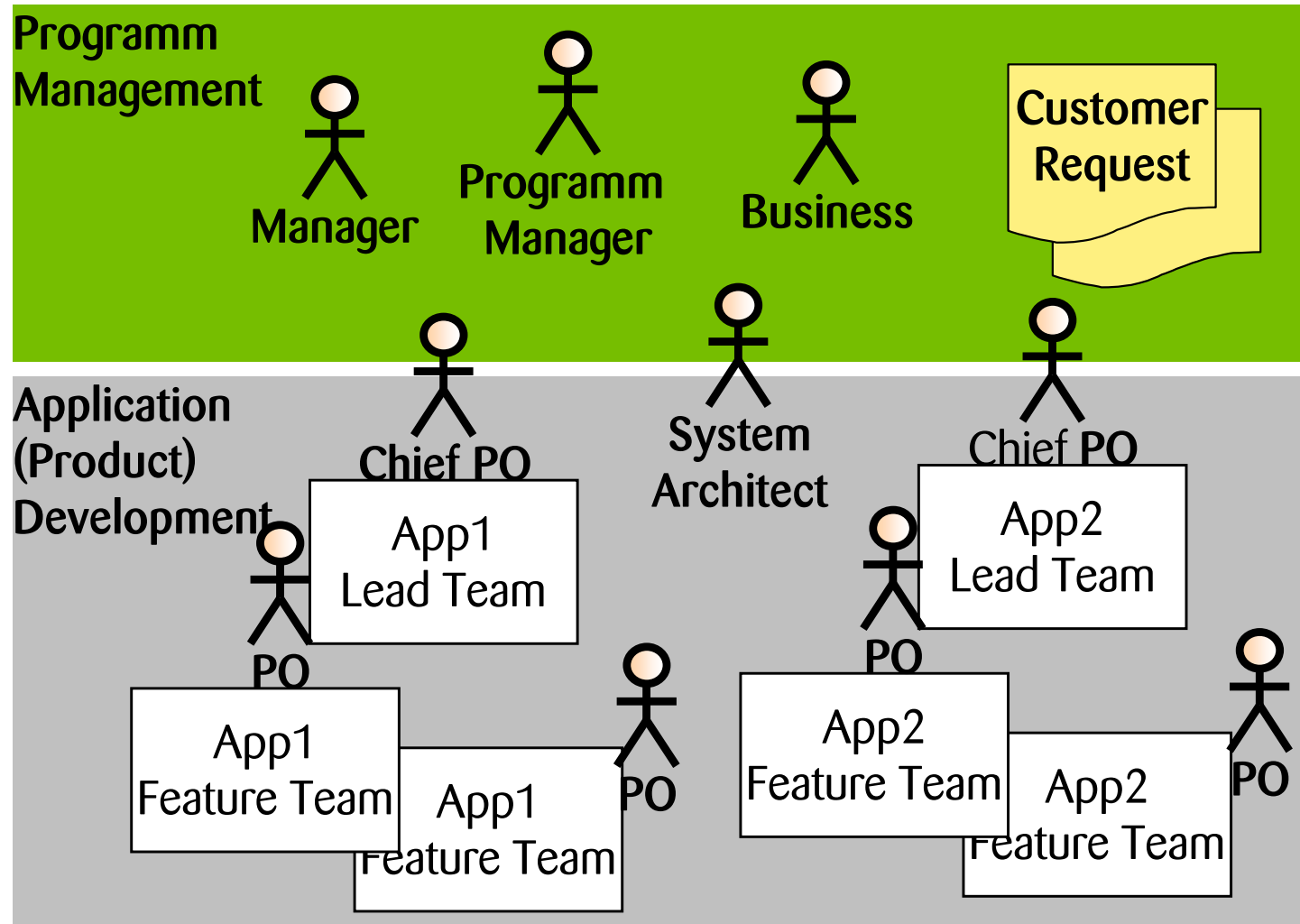
-
- Aktivierung von Wissen im Team (Spirit)
 - Multidisziplinäre Mitarbeiter statt Spezialistenpools
 - Beständiges situationsbezogenes Optimieren des (vorgeschlagenen) Vorgehens
 - Direkte Kommunikation im Team und mit Kunde
 - Bewahrung nur des nutzbringenden Wissens in Dokumenten
 - Vom Team nachgewiesene Selbstkontrolle
 - Anhand von Policies (Beispiel: Definition von Done-Kriterien)

Die agile Form eines Projekt Setup

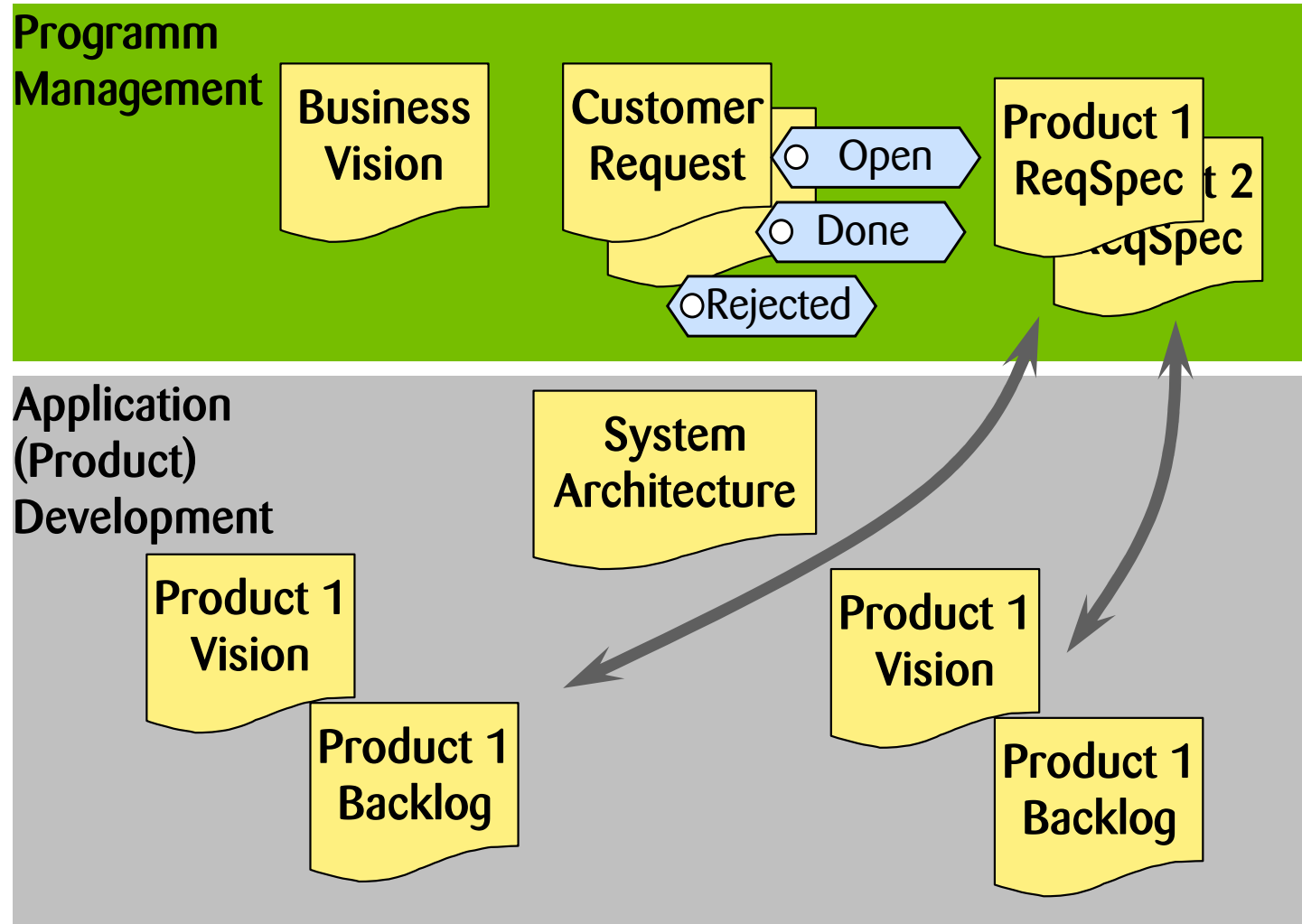


Gemeinsame
Vision

Typischer Informationsfluss in einer agilen Organisation



Minimal sinnvolle Dokumentation



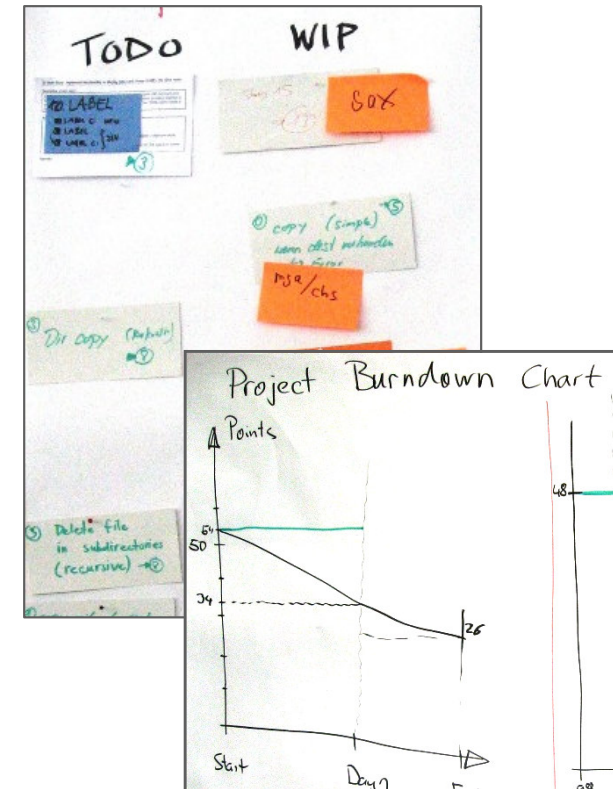
Die drei Stufen zu einer agilen Kultur

- Integrieren agiler Best Practices im Projekt
 - Professionelles Software Engineering
 - Befähigung des Teams
 - Veränderung der Projektkultur
- Agile Prozesselemente einsetzen
 - Veränderungen auf operativer Ebene
 - Nicht zwingend Veränderung der Unternehmenskultur
- Lean Management
 - Veränderungen auf Management Ebene
 - Veränderung der Unternehmenskultur

Integrieren agiler Best Practices im Projekt



- XP Best Practices als Vorbild nehmen
 - Continuous integration
 - Test First Ansatz in der Implementierung
 - Automatisches Testen auf Level Modul- und Systemtest
 - Feature orientiertes Arbeiten (Feature = Kundennutzen)
 - Aufwand zur Umsetzung eines Features < 3 Wochen
 - Done Kriterien zur Abnahme von Features definieren
- Optimierungen innerhalb des Teams
 - Pull Prinzip zur Aufgabenverteilung im Team
 - Visuelles Projektmanagement (Burndown, Post-its, ...)
 - Pairwise Programming
 - Bei komplexen Arbeiten
 - Als informellen Walkthrough
 - Für Know-how Sharing und zum Einlernen neuer Team Member
 - Daily Standups
- Ausprobieren und Experimentieren

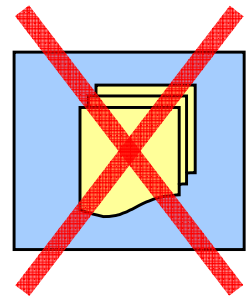
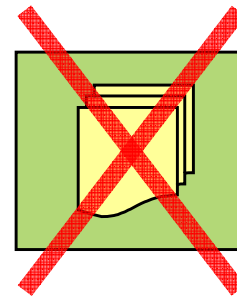
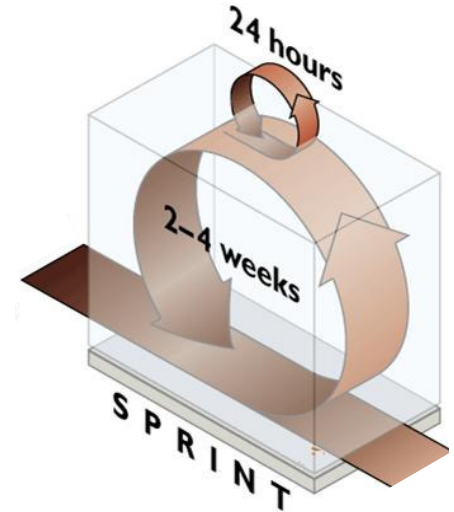


Moving to an Agile Culture
Folie 20
19. November 2009

Agile Prozesselemente einsetzen

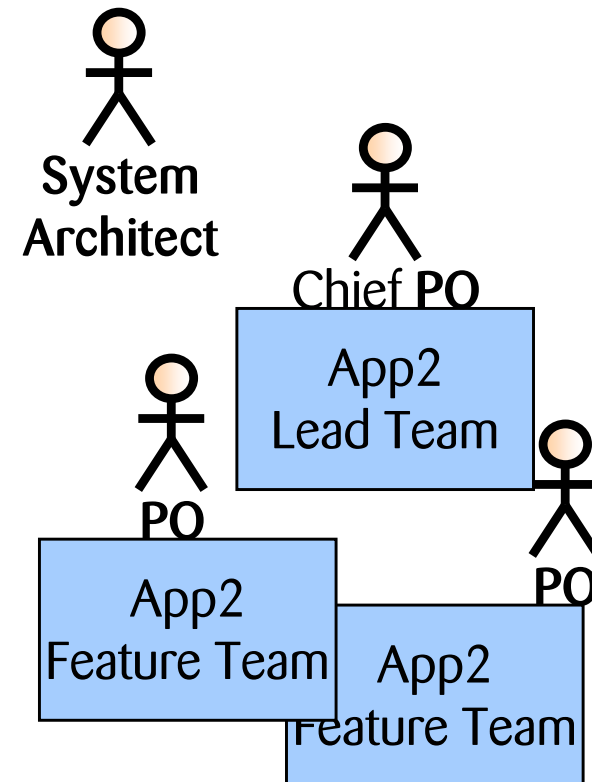


- Nur unbedingt notwendige Upfront-Arbeiten
 - Architektur
 - Spezifikation (Begriff der Vision)
- Iteratives Arbeiten
 - Team schätzt Aufwand (zur Feststellung der Velocity)
 - Done Kriterien zur Abnahme von Features
 - Iterationsergebnis vom Kunden / Benutzer validieren lassen
- Scrum Management Elemente einsetzen
 - Produkt und Iterationsbacklog
 - Priorisierung
- Streichen überflüssiger Dokumentation
- Vorhandene Spezialistenpools auflösen



Moving to an Agile Culture
Folie 21
19. November 2009

- Übertragen von Verantwortung an das Team
 - Scope, Budget, Time Nivellierung mit Team
 - Abstand zwischen Fachbereich und IT auflösen
- Vermeiden von Micromanagement
 - Steering Committee arbeitet auf Basis von Business Features
 - Zuweisung von Business Features an Teams mit Team Leads
 - Zerlegung von fachlichen Anforderung erfolgt durch das Team
- Product Owner Prinzip einführen
 - Team Lead wandelt sich zum Product Owner eines Teams
 - Ein Produkt, eine Applikation wird durch ein oder mehrere Teams weiter entwickelt
 - Ein Team übernimmt Lead für das Produkt / die Applikation
 - Product Owner der Lead Teams sind Teil des Steering Committes



The Nokia Test: Are you Agile?



<http://agileconsortium.blogspot.com/2007/12/nokia-test.html>

Are you Agile?

- Are your Iterations time-boxed and last no longer than 4 weeks?
- Is the software completely tested and working at the end of an iteration?
- Is the customer allowed to operate the software ?
- Can the iteration start before specification is complete?

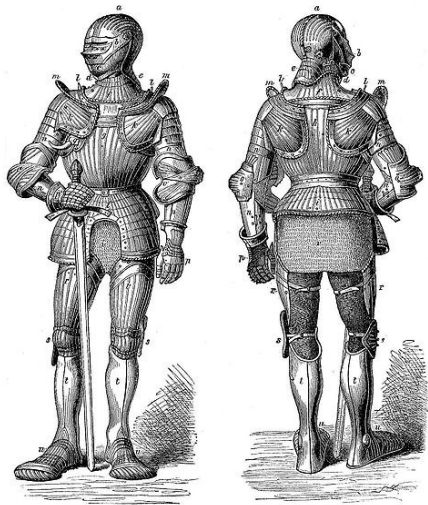
Do you Scrum?

- Does the team know who the product owner is?
- Is there a product backlog prioritized by business value?
- Does the product backlog have estimates created by the team?
- Does the team generate its burndown charts and knows its velocity?
- Are there no project managers (or anyone else) disrupting the work of the team within a sprint?

Der Nutzen einer agile Kultur

Ob agil – oder ob nicht agil – oder wie agil ist nicht das Wesentliche...

Das Wesentliche ist es am Markt erfolgreich zu sein



nicht agil



wir



Agilistas