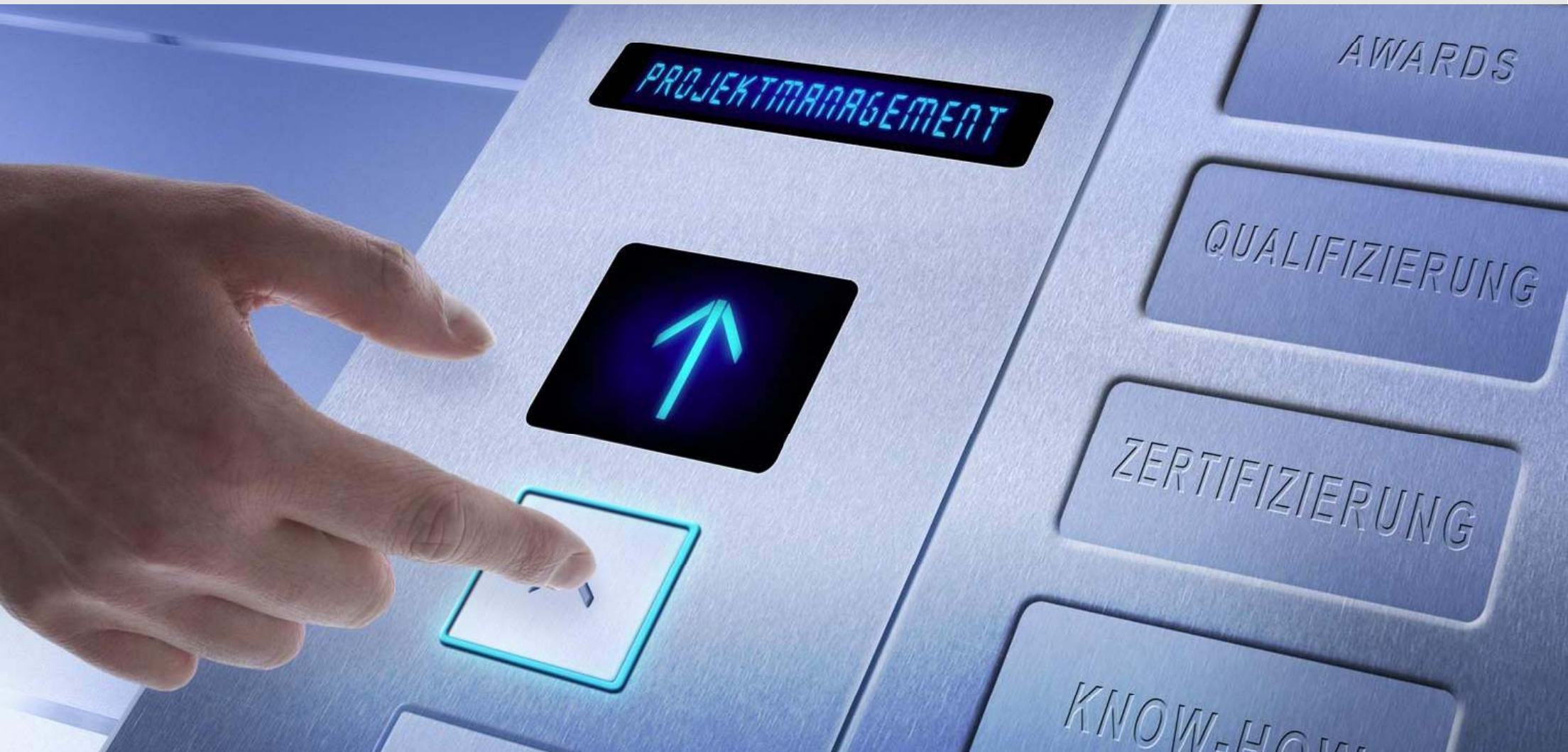


GPM Region Düsseldorf/Rhein-Ruhr

PM-Tag 2011 am 16.09.2011 mit dem Vortrag :
Wo bleiben traditionelle Rollen im Scrum?



1. Foliensatz der Veranstaltung am 16.09.2011 bei der

IT.NRW

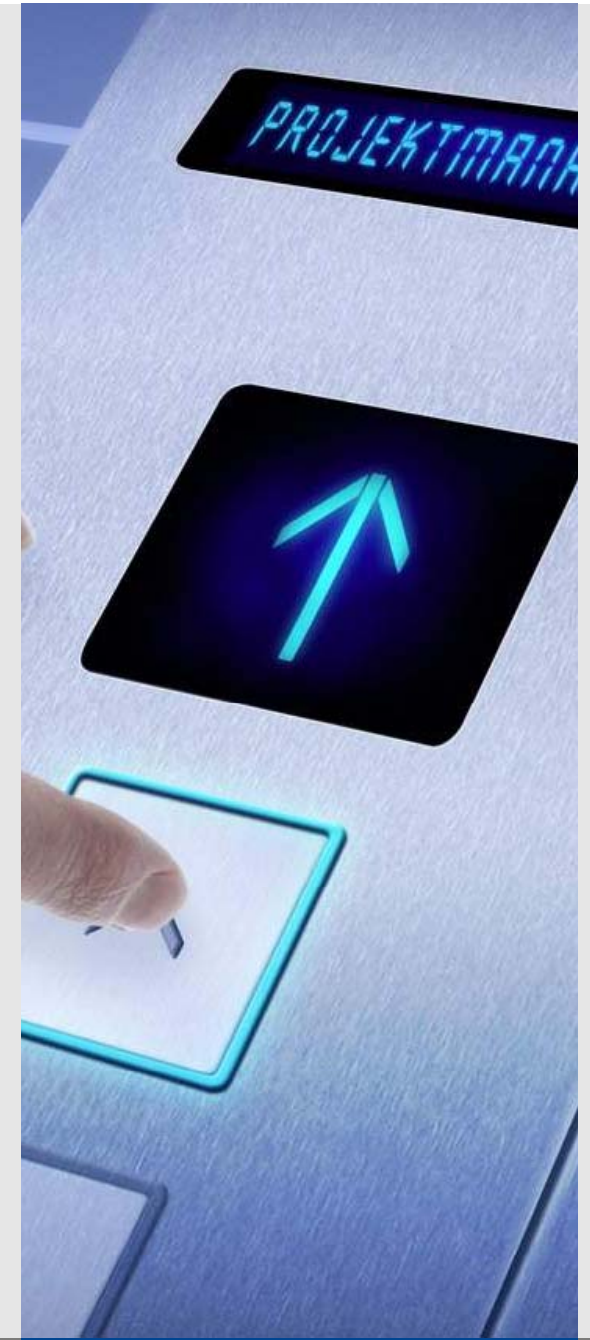
Mauerstr. 51

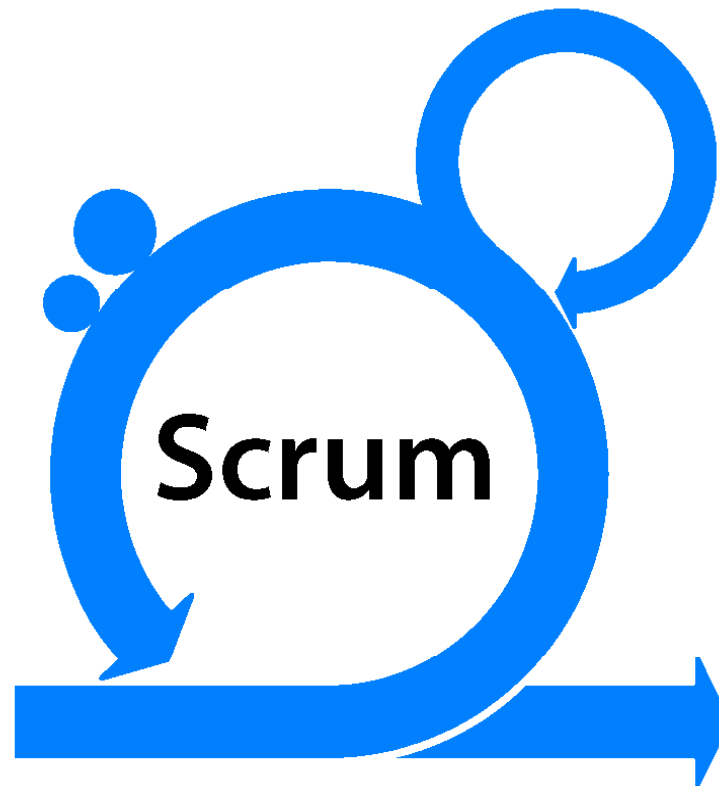
40476 Düsseldorf

2. Kontaktdetails des vortragenden Unternehmens

3. Kontaktdetails GPM/GPM Region Düsseldorf-Rhein/Ruhr

Hinweis: Der Inhalt dieser Präsentation muss nicht die Meinung der GPM Deutsche Gesellschaft für Projektmanagement e.V. darstellen. Alle Rechte und Pflichten obliegen dem Präsentierenden (Dr. Hofmann, Oose Innovative Informatik GmbH, Hamburg)





Scrum

„Scrum is a framework for developing complex products and systems. It is grounded in empirical process control theory.

Scrum employs an iterative, incremental approach to optimize predictability and control risk.

Within each iteration, Scrum employs self-organizing, crossfunctional Teams to optimize flexibility and productivity.“

(Scrum Guide, 2009)



Scrum beschreibt grundlegende Abläufe, Regeln und Rahmenbedingungen für die Entwicklung komplexer Produkte und Systeme.

Ein Schwerpunkt des Vorgehens liegt auf der kontinuierlichen Verbesserung der Prozesse auf Basis der gemachten Erfahrungen.

Das Scrum-Vorgehen ist iterativ-inkrementell.

Ein besonderer Augenmerk liegt auf der Teamorganisation.

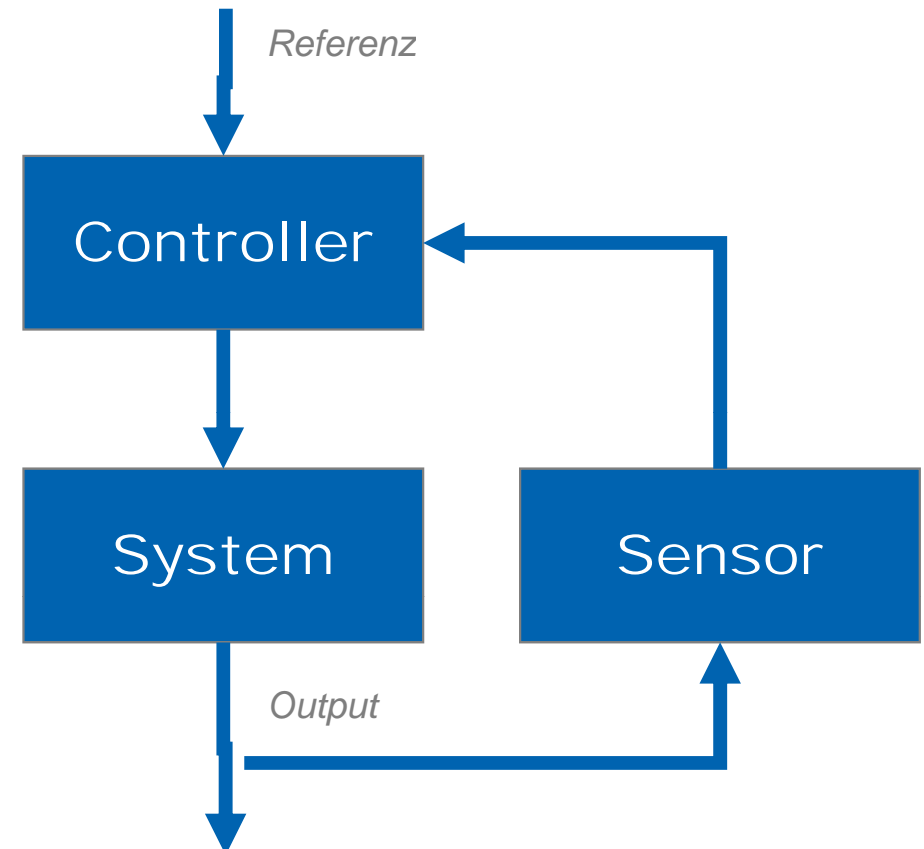
Entstehung und Entwicklung

- Die Entstehung von Scrum reicht zurück bis Anfang 1990
- Wesentliche Einflüsse bei der Entwicklung von Scrum entstammen Lean Production und dem Lean Product-Development
- Jeff Sutherland und Ken Schwaber haben die Entwicklung von Scrum wesentlich geprägt
- Heute steuern die Scrum Alliance und scrum.org die Vermarktung von Scrum und arbeitet an der Etablierung von Scrum-Zertifizierungen

Empirical Process Control Theory

Grundelemente

- **Transparenz** — der Prozess muss in seinen wesentlichen Aspekten einsehbar sein und von den Beteiligten verstanden werden
- **Untersuchung** — Regelmäßige Untersuchung der Prozesse durch erfahrene und kundige Menschen erlaubt Abweichungen zu entdecken.
- **Adaption** — werden Abweichungen entdeckt, müssen umgehend der Prozess oder das verwendete „Material“ angepasst werden.



Empirical Process Control Theory und Scrum

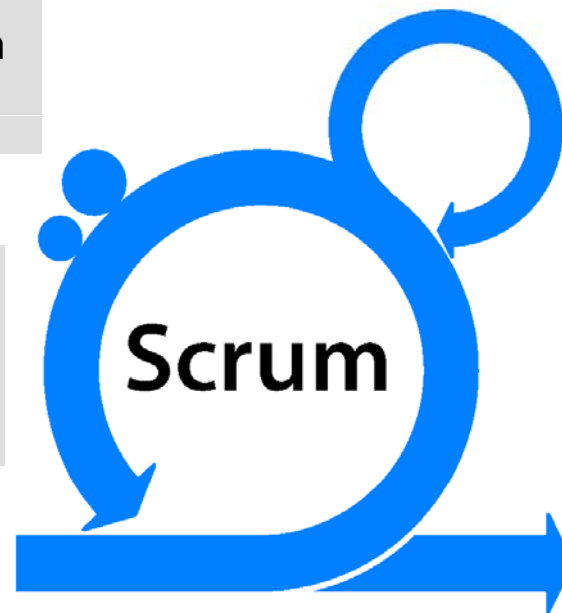
Scrum kennt 3 Untersuchungs und Adaptions-Punkte die unterschiedliche Stellschrauben beeinflussen:

Sprint Retrospektive

Umgang mit Hindernissen im Entwicklungsprozess

Sprint-Review/-Planung

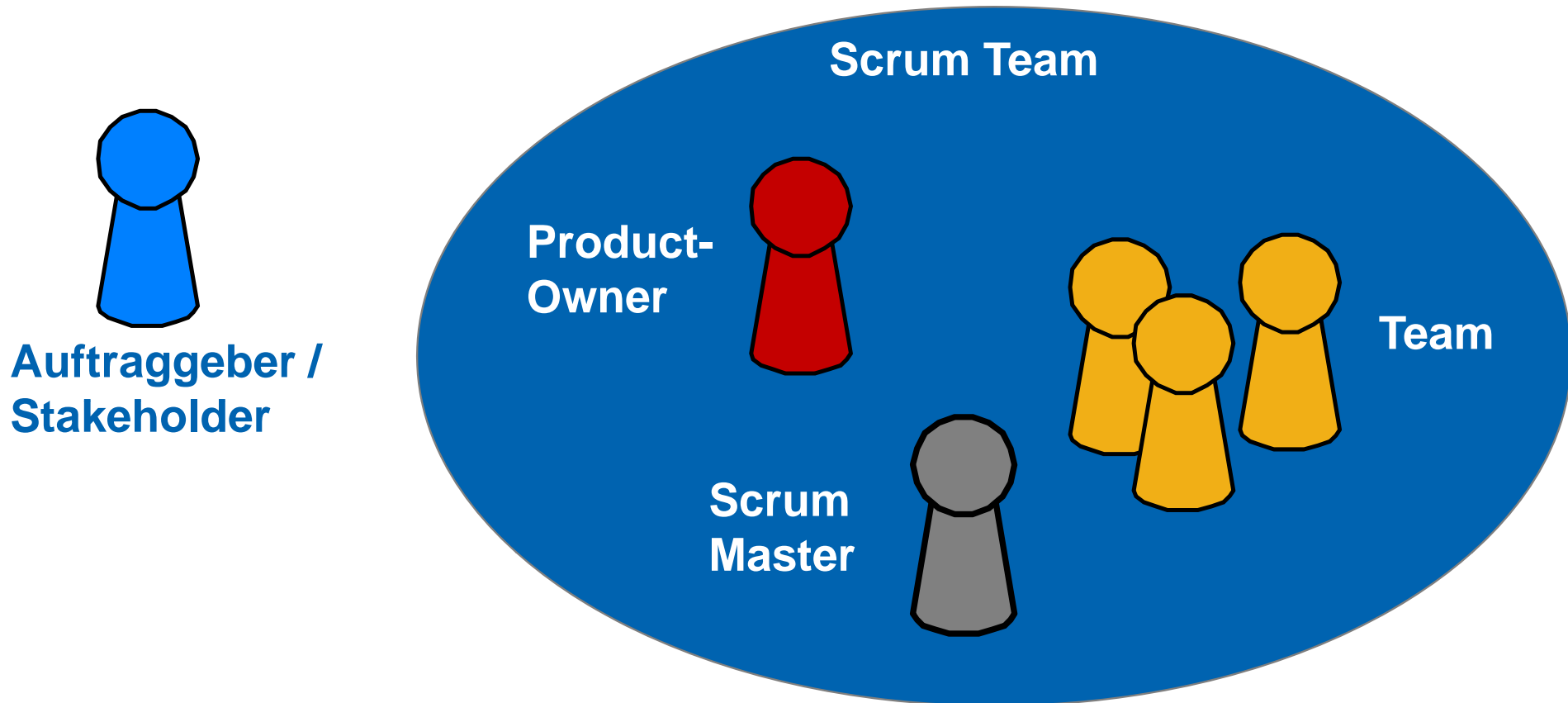
Umgang mit Hindernissen auf dem Weg zum Release



Daily Scrum-Meeting

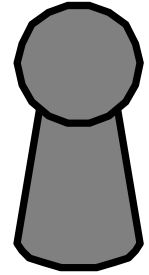
Umgang mit Hindernissen auf dem Weg zum Sprint-Ziel

Rollen, das Scrum Team



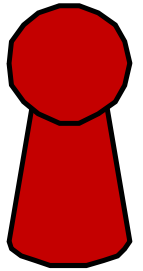
Das Scrum-Team ist selbst dafür verantwortlich seine Arbeit zu organisieren und niemand außerhalb des Scrum-Teams kann ihm vorschreiben, wie es seine Arbeit tun soll.

Scrum Master



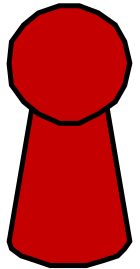
- Der Scrum-Master ist dafür verantwortlich, dass die Scrum-Regeln und -Abläufe eingehalten werden und die nötigen Rahmenbedingungen vorhanden sind.
- Der Scrum-Master unterstützt das Scrum-Team und das Umfeld dabei die Prozesse mit Leben zu füllen und weiterzuentwickeln
- Der Scrum-Master ist *nicht* der „Chef“, er hat eher eine Coaching- und Unterstützer-Rolle

Produkt Owner



- Der Produkt-Owner ist die Person die das Produkt-Backlog verantwortet. Er entscheidet unabhängig darüber welche Anforderungen, mit welcher Priorität enthalten sind.
- Der Produkt-Owner ist für den Nutzen und die Wirtschaftlichkeit der im Produkt-Backlog enthaltenen Anforderungen verantwortlich.
- Der Produkt-Owner wird in seinen Entscheidungen von niemanden überstimmt. Er alleine setzt die Prioritäten für die Arbeit des Teams.

Und was macht der Projektleiter?



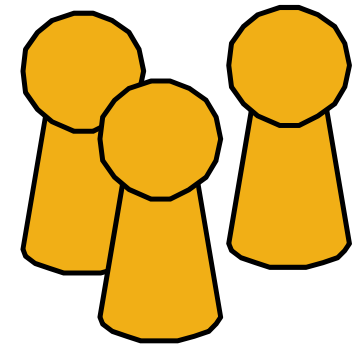
- Projektleiter werden in Scrum-Projekten nicht mehr benötigt.
- Die Aufgaben des Projektleiters verteilt sich auf die Rollen Product Owner, Scrum Master und Team.
- Zentrale Entscheidungsinstanz bezüglich
 - der Projektergebnisse ist der Product Owner
 - der Zusammenarbeit ist der Scrum Master
 - der konkreten Arbeitsschritte ist das Team

Und wo ist der Projektleiter?



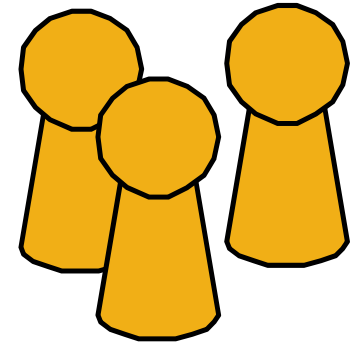
Aufgaben nach PMI	Übernommen durch
Scope-Management	Product Owner für Produkt / Release Team für Sprint
Zeit-Management	Product Owner für Releaseplan Team für Sprint Backlog
Kosten-Management	Product Owner
Kommunikationsmanagement	Product Owner für Release-Reporting Team für Sprint-Reporting
Risikomanagement	Product Owner mit Hilfe des Teams
Qualitätsmanagement	Product Owner für Produktmerkmale Team für qualitätssichernde Maßnahmen Scrum-Master für die Anwendung des Prozesses
Lieferantenmanagement	Product Owner und Team
Personalmanagement	Management für die Bereitstellung der Mitarbeiter Team für Produktivität und kontinuierliche Prozessverbesserung

Das Team



- Das Team entwickelt aus den Features im Backlog ein funktionierendes Produkt.
- Das Team organisiert sich selbst. Es gibt im Team keine Titel oder besondere Rollen.
- Das Team ist so besetzt, dass alle benötigten Fähigkeiten und Fachkenntnisse im Team vorhanden sind
- Nur das Team selbst legt fest, wie viel Arbeit es in einem Sprint erledigt.
- Teams sollten nicht größer als 7 Mitglieder sein.
- Die Team-Zusammensetzung kann - wenn nötig - am Ende eines Sprints geändert werden.

Und wo sind ...



- ... die Software-Architekten? – Mitglied eines oder mehrerer Teams
- ... die Anforderungsanalytiker? – Mitglied des Teams
- ... die Qualitätssicherer? – Mitglied des Teams
- ... die Technischen Redakteure? – Beides sinnvoll, je nach Kontext: in oder außerhalb des Teams
- ... die Support-Mitarbeiter? – Außerhalb des Teams!

Der Sprint

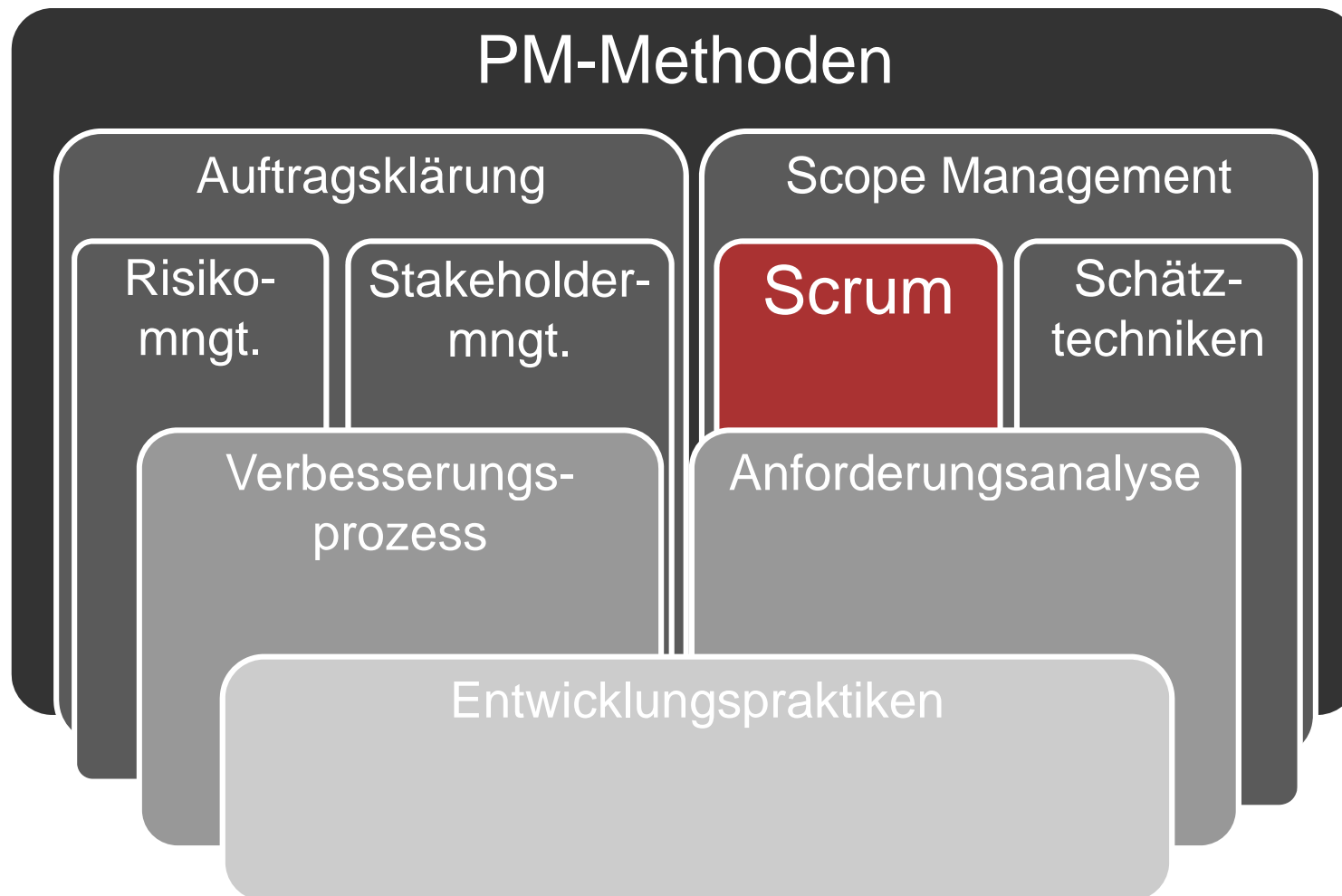
- Sprint ist der Scrum Name für eine Iteration
- Ein Sprint hat eine gleichbleibend feste Länge (typische Größenordnung 2-4 Wochen)
- Innerhalb eines Sprints bleiben Team-Zusammensetzung und Sprint-Ziel unverändert
- Zu einem Sprint gehören
 - Sprint-Planung
 - Analyse- und Entwicklungsarbeit
 - Sprint-Review
 - Sprint-Retrospektive



Und wo bleibt der Projektplan?

- Für ein Scrum-Projekt benötigen Sie keine Gantt-Diagramme
- Eine Work-Breakdown-Structure ist sinnlos in einem Scrum-Projekt
- Eine Feature-Breakdown-Structure ist in manchen Situationen sinnvoll
- Weit in der Zukunft liegende Aufgabenpakete werden weder identifiziert noch geplant
- Die Planung ist rollierend – adaptiv
- Die Grobplanung erfolgt auf Basis von Produktmerkmalen (Features)
- Die Feinplanung erfolgt auf Basis von Aufgaben
- Die Grobplanung hat einen Zeithorizont von ein paar Monaten
- Die Feinplanung hat einen Zeithorizont von ein paar Wochen

Scrum ist nur ein Rahmenwerk



Product-Backlog

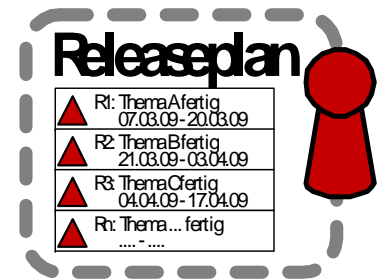
- Eine Liste mit allen Dingen die getan werden müssen um ein erfolgreiches Produkt zu entwickeln
 - Anforderungen
 - Features
 - Verbesserungen / Überarbeitungen
 - Fehlerbehebungen
 - Technologien
 - ...
- Die Liste ist eindeutig nach Priorität sortiert, der Wert der Einträge muss einschätzbar sein
- Das Product-Backlog wird permanent weiterentwickelt
- Es sollten nur jeweils die wichtigsten Punkte der Liste detailliert werden.
- Alle tragen zum Backlog bei, aber nur der Produkt-Owner entscheidet, was aufgenommen wird



Release-Plan

- Der Release-Plan enthält
 - die Ziele für ein Release
 - die Kriterien wann ein Release fertiggestellt ist
 - die Hauptrisiken für das Release
 - die umzusetzenden Anforderungen

- Ziel der Release-Planung ist ein Plan zu erhalten, den sowohl das Scrumteam wie auch der Rest des Unternehmens so verstehen und mit dem über die Ziele und Investment-Entscheidungen kommuniziert werden kann.

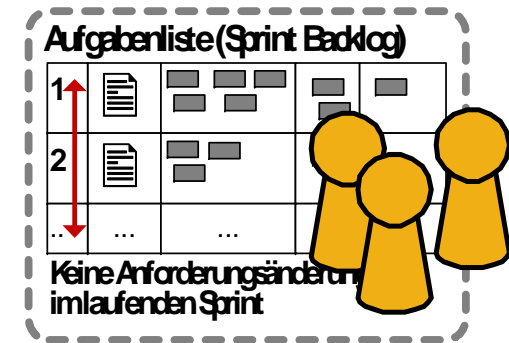


Sprint-Goal

- Das Sprint-Goal ist eine Zusammenfassung des Ziels, welches mit dem Sprint erreicht werden soll.
- Das Sprint-Goal motiviert die zu entwickelnde Funktionalität und gibt Orientierung wenn aufgrund von Problemen entschieden werden muss, wie Alternativlösungen aussehen könnten.

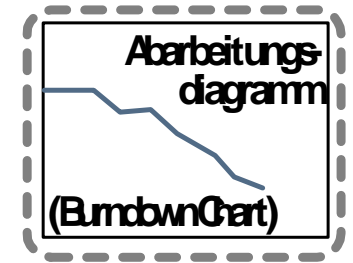
Sprint-Backlog

- Enthält die Aufgaben, die im Sprint erledigt werden müssen, um aus einer Anforderung im Product-Backlog ein funktionierendes Stück des Produkts zu entwickeln.
- Die Verantwortung für das Sprint-Backlog liegt alleine beim Team



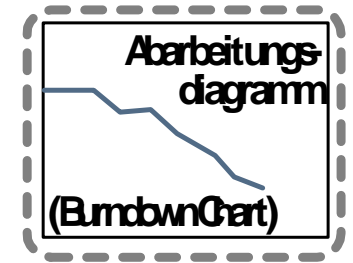
Sprint-Burndown-Chart

- Der Sprint-Burndown-Chart visualisiert den Restaufwand für die Abarbeitung der Aufgaben eines Sprints.



Release-Burndown-Chart

- Der Release-Burndown-Chart visualisiert den Restaufwand für die Fertigstellung eines Release



Inkrement / Definition-of-Done (DoD)

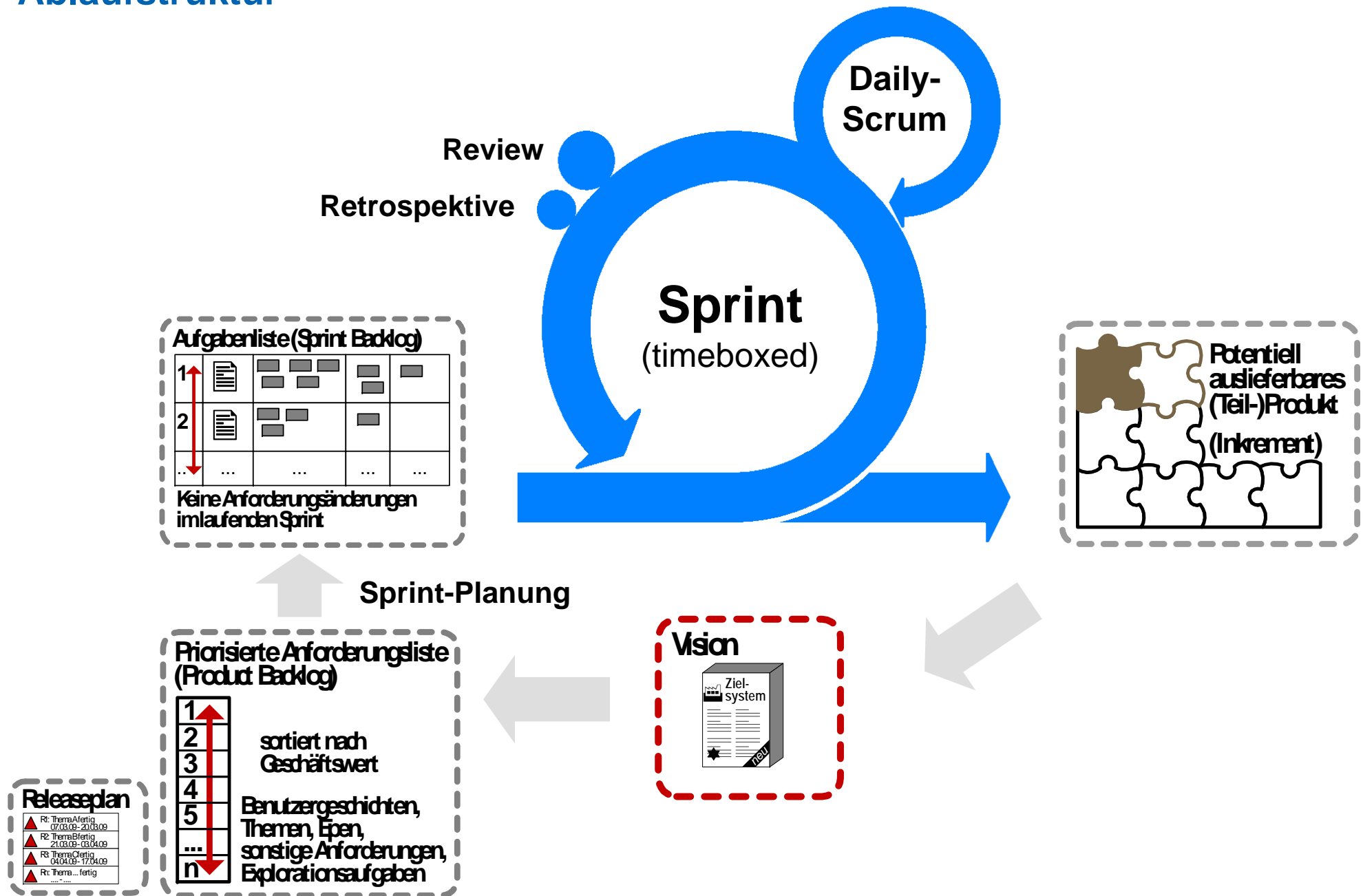


Ein **Inkrement** bezeichnet die Funktionalität, die am Ende einer Iteration fertig gestellt wurde

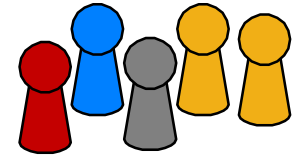
Die **Definition-of-Done** legt fest, wann eine Funktionalität als so weit fertiggestellt gilt, dass nicht mehr mit Nacharbeiten gerechnet werden muss.

Idealerweise gilt eine Funktionalität erst dann als fertig wenn diese produktiv eingesetzt wird. Dafür müssen die Funktionalitäten noch im Sprint in Produktion gebracht werden!

Ablaufstruktur

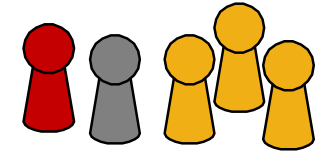


Initialfüllung des Product-Backlogs



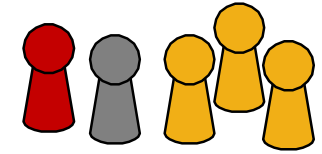
- Das Produkt-Backlog wird zum Beginn des Projekts einmal so weit gefüllt, dass
 - Die Idee des Produkt klar wird
 - Genug wissen über die Anforderungen in der Breite vorhanden sind um ein Architekturentwurf zu machen
 - (Eine Abschätzung des Umfangs für die Releaseplanung erfolgen kann)

Sprint Planing P1



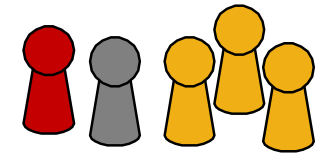
- Der Product-Owner präsentiert dem Team die Top-Einträge des Product-Backlogs
- Gemeinsam wird entschieden, welche Anforderungen umgesetzt werden sollen.
- Berücksichtigt werden müssen Produktstand, Teamkapazität, bisherige Performanz.
- Größenordnung 4h bei 4 Wochen Iteration
- Das Team entscheidet allein über die Menge der Anforderungen die umgesetzt werden können

Sprint Planing P2



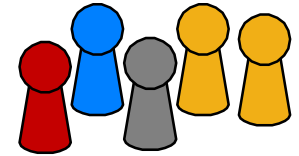
- Das Team entwickelt aus den ausgewählten Anforderungen, die notwendigen Aufgaben zur Umsetzung und füllt damit das Sprint-Backlog
- Größenordnung 4h bei 4 Wochen Iteration
- Der Product-Owner unterstützt weiter bei der Klärung, wie die einzelnen Anforderungen zu verstehen sind.

Daily Scrum



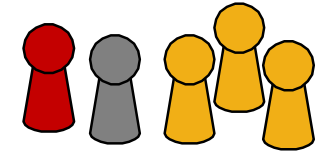
- Ein ca. 15 minütiges Meeting des Scrum-Team
- Jeder beantwortet ausschließlich die folgenden Fragen:
 - Was habe ich seit dem letzten Meeting *erreicht*?
 - Was plane ich bis zum nächsten Meeting zu erreichen?
 - Welche Hindernisse stehen mir zur Zeit im Wege?
- Praktischerweise unmittelbar vor dem Mittagessen
- Der Scrum-Master ist für das Stattfinden und die Disziplin im Meeting verantwortlich
- Niemand außerhalb des Scrum-Teams darf sich zu Wort melden, zuhören ist erlaubt.

Sprint Review



- Demonstration der erreichten Funktionalität des Produkts gegenüber den Stakeholdern
 - Bericht über den Entwicklungsverlauf und die geplanten Arbeiten
 - Benennen von Problemen und Hindernissen
 - Größenordnung 4h bei 4 Wochen Iteration
-
- Das Feedback der Stakeholder fließt anschließend in das Produkt-Backlog ein.

Sprint Retrospektive



- Grundstruktur:
 - Was ist im Sprint passiert?
 - Was lief besonders gut? Was war schwierig?
 - Was wollen wir im nächsten Sprint anders machen?
- Wichtig:
 - Alle beteiligen
 - Ursachen ergründen und nicht nur Symptome bearbeiten
 - Kleine Schritte machen, welche das Scrum-Team selbst unmittelbar im nächsten Sprint umsetzen kann
 - Störungen im Umfeld werden durch den Scrum-Masters adressiert. Solange sie bestehen muss das Team Wege finden damit umzugehen.
- Größenordnung 3h

CMMI

CMMI: **Reifegradmodell** der Prozeßreife.

Beginn der Arbeiten **1986** am SEI (Carnegie Mellon University, Pittsburgh).

Ziel: **Lieferantenbeurteilung** für des US-Verteidigungsministerium.

5 Level:

- 1) Initial
- 2) Managed
- 3) Defined
- 4) Quantitativ Managed
- 5) Optimizing

CMMI Level 2: Generische Praktiken

Generische Praktik	Erläuterung	Umsetzung mit Scrum
2.1: Erstellen einer organisationsweiten Strategie	Konkrete Kriterien festlegen für den Einsatz oder Nicht-Einsatz von Scrum	Außerhalb von Scrum. Möglichkeit, die Absichten bzgl. Scrum zu kommunizieren.
2.2: Prozess planen	Operationalisierung von Scrum	Außerhalb von Scrum; auch mit Scrum-Mechanismen machbar. Elemente der Op. sind Sprint-Planning, -Review, -Retrospective und das Daily-Scrum.
2.3: Ressourcen bereitstellen		Für das konkrete Projekt erfolgt dies im Sprint-Planning.
2.4: Verantwortlichkeit zuweisen	Rollen und Verantwortungen definieren.	Besetzung der Rollen Product Owner, Scrum Master und Team (funktionsübergreifend).

CMMI Level 2: Generische Praktiken

Generische Praktik	Erläuterung	Umsetzung mit Scrum
2.5: Personen schulen	Auswahl der Themen und Personen nach Bedarf.	Außerhalb von Scrum. Z.B. dieser Zertifizierungsworkshop
2.6: Konfiguration managen	Versionen der verschiedenen Liefergegenstände kontrolliert verwalten.	Scrum schreibt dazu nichts vor. Etabliert ist die Verwendung von Versionskontrollsystemen.
2.7: Betroffene einbeziehen	Stakeholder-Management	Einbeziehung des Kunden ist empfohlene Praxis. Product Owner und Scrum Master sind verantwortlich. Z.B. Product Owner Board.

CMMI Level 2: Generische Praktiken

Generische Praktik	Erläuterung	Umsetzung mit Scrum
2.8: Prozess überwachen und steuern	Messung des Projektfortschritts, Einleitung von Maßnahmen.	Daily-Scrum mit Sprint-Burndown-Chart. Sprint-Review mit Release-Burndown-Chart. Velocity-Extrapolation.
2.9: Prozesseinhaltung objektiv überwachen		Scrum Master als Prozess-Coach für das konkrete Projekt. Ggf. Scrum-Competence Center für übergreifende Unterstützung.
2.10: Status-Review mit höherem Management		Sprint-Planning, Sprint-Review, Daily-Scrum. Dazu als Artefakte: Product-Burndown-Chart und die Hindernisliste (Impediment Backlog).

Kontakt



GPM Deutsche Gesellschaft für Projektmanagement e.V.

Hauptgeschäftsstelle Nürnberg

FrankenCampus – Frankenstraße 152

90461 Nürnberg

Tel.: +49 (0)911 43 33 69-0

Fax: +49 (0)911 43 33 69-99

E-Mail: info@GPM-IPMA.de

Internet: <http://www.GPM-IPMA.de>

Veranstaltungshinweise der GPM Region Düsseldorf/Rhein-Ruhr

http://www.gpm-ipma.de/ueber_uns/regionen/.....

Bei Fragen zum Veranstaltungsprogramm oder zu Netzwerk-
möglichkeiten können Sie sich gerne an die

Region Düsseldorf / Rhein-Ruhr

Guido Bacharach

Tel.: +49 (0) 208 59 08 24

E-Mail: g.bacharch@GPM-IPMA.de

Dieter Staudt

+49 (0) 211 9449-3717

d.staudt@GPM-IPMA.de

wenden.

