

---

# Definition of READY

Wann kann's endlich losgehen?

Agile Breakfast Konstanz , 5. Juli 2011

Felix Rüssel

# Vorstellung

---

## Felix Rüssel

- Freier Berater für agile Methoden, Projektmanagement & Sourcing
  - [www.agile-rescue.com](http://www.agile-rescue.com)
- Nearshore Scrum Teams
  - [www.agile-nearshoring.com](http://www.agile-nearshoring.com)
- Agile Methoden seit ca. 2004
  - Blog (~2008): [www.armerkater.de](http://www.armerkater.de)



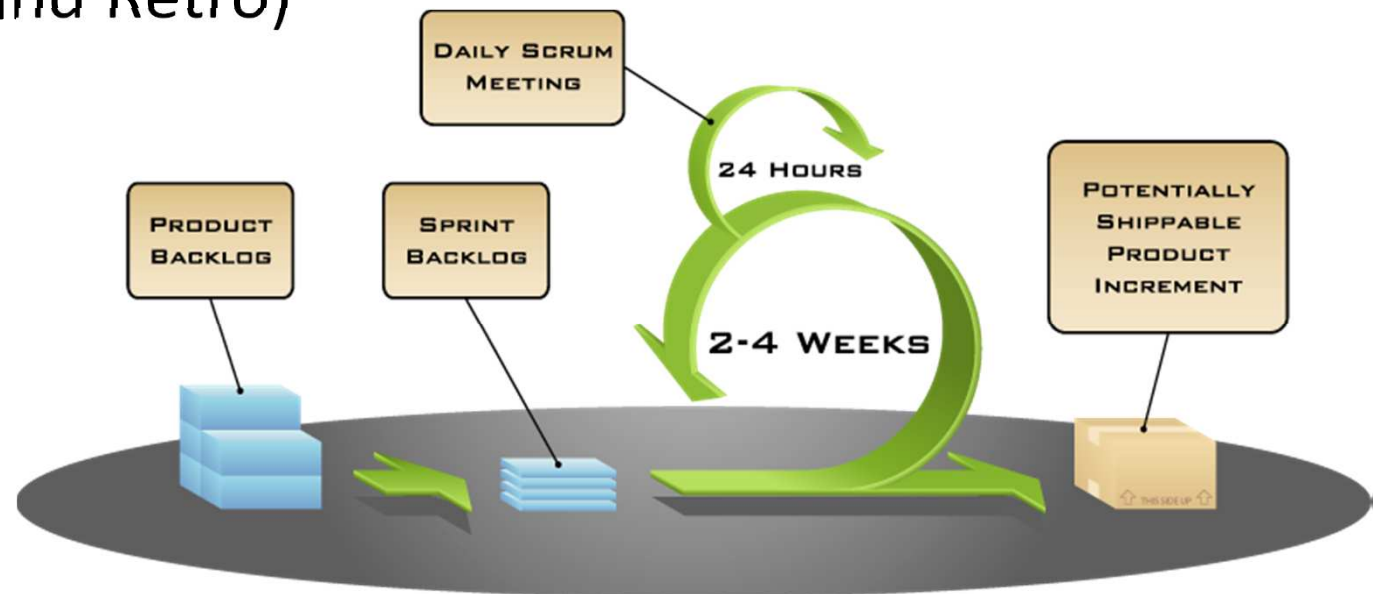
# Agenda

---

- Vorstellung
- Einführung / Überblick
- Bekanntheitsgrad / Bedeutung
- Definition of READY
- Wie wird's READY?
- Zusammenfassung

# Scrum Prozess

- PBL mit PBIs (meist User Stories)
- Sprint Planning: PBL -> SBL
- Sprint/Umsetzung: SBL -> Increment
- Review (und Retro)
- Release!



Quelle: <http://www.mountaingoatsoftware.com>

COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

# Wertschöpfung im Sprint

---

- Ich verstehe **warum** und **was** ich tun soll
- Ich habe eine Vorstellung, **wie** ich es umsetzen kann
- Ich **kann** es umsetzen (**READY!**)
- und **setze es um**
  - und während Umsetzung werde ich nicht abgelenkt
- Ich habe es erfolgreich umgesetzt (**DONE!**)
- Es ist das, was gewünscht & **benötigt** wird
- Es erzeugt einen **Mehrwert**

# Probleme / Herausforderungen

---

- Keine **klaren** Formulierungen, Kriterien, Ziele
- Zielerreichung nicht **test-/messbar**
- **Termine** nicht genannt
  
- Keine / zu viele **Details**
- **Risiken** nicht identifiziert
  - Abhängigkeiten zu anderen Teams, zu externen Stakeholdern, technische Risiken, ...
  
- **Auftraggeber** unbekannt?!

# Probleme / Herausforderungen

---

- Eine grundlegende Ursachen für schlechte Produktivität und Probleme im Sprint: Stories die nicht bereit sind.
  - Unterbrechungen, Diskussionen, falsche Annahmen, ...
- Muster: „Chaos rein -> Chaos raus“

Frage: Wer kennt das Problem?

# READY als Quality Gate

---

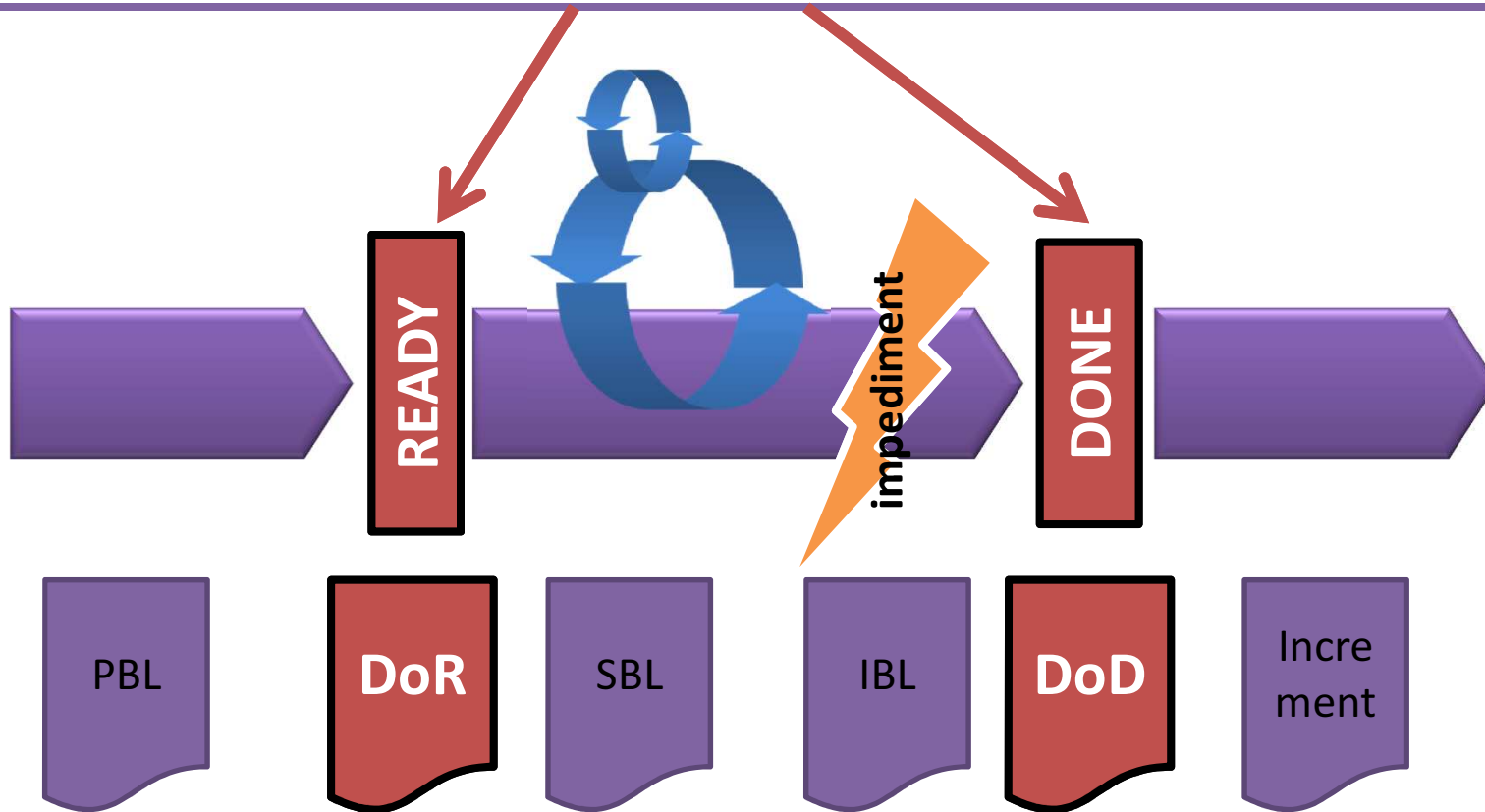
- Nur Stories, die bereit (**READY**) sind werden bearbeitet
  - Für Schätzungen muss das Schätzobjekt klar sein
    - verständlich, Umfang/Größe, Abhängigkeiten
  - Für Implementierungen müssen Vorbedingungen geklärt und Abhängigkeiten klar sein
    - Stakeholder! Andere Teams! Technik!
- Was nicht bereit (READY) ist, kommt **nicht** in einen Sprint!

# READY als Quality Gate

---

- Verantwortung: Product Owner
  - der PO trägt Verantwortung dafür, dass Stories für die SchätZRunden vorbereitet werden (READY sind)
  - nur READY Stories gehen in ein Planning / in ein Sprint
- Team: Im Planning kann auf DoR verwiesen werden
  - Der Scrum Master achtet auf Einhaltung
  - Diskussion schwammiger Stories wird unterbunden mit Verweis auf „Vorbereitung“ / DoR

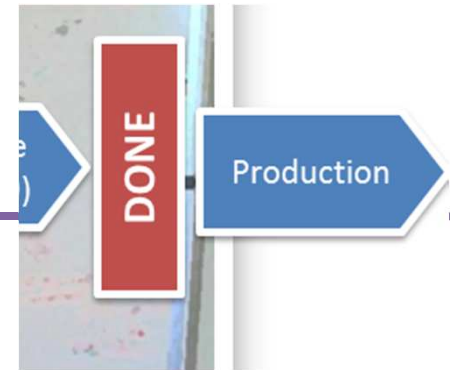
# READY & DONE: Die 2 stabilen Zustände



# Scrum Board

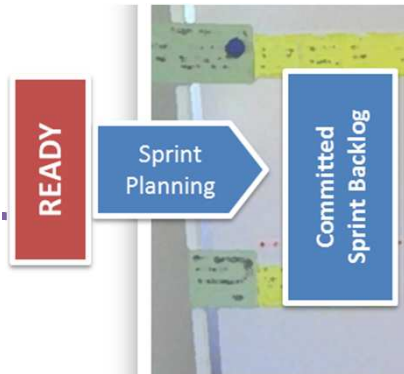


# Definition of **DONE** (DoD)



## Definiert wie etwas den Sprint **VERLÄSST**

- Dinge werden mit einem gemeinsam vereinbarten Qualitätsstandard erstellt. Sie sind dann **wirklich** erledigt („real working product“).
- **DONE** in Verantwortung des Teams
- Team erarbeitet **Definition of Done**, der Product Owner akzeptiert oder verhandelt nach.
- Im Review muss Story DoD erfüllen, sonst lehnt PO Abnahme ab.
- Beispiel:
  - Umgesetzt, getestet, dokumentiert
  - Keine zusätzlichen technischen Schulden
  - Akzeptanzkriterien erfüllt
  - Standard X erfüllt
- Nur ca. **50% aller Scrum Teams** liefern wirklich **DONE**
  - **Jeff Sutherland**: A Practical Roadmap to Great Scrum: A Systematic Guide to Hyperproductivity (<http://www.scrum-day.de/vortraege/2010-11-17-ScrumDay-Practical%20Roadmap.pdf>)



# Definition of **READY** (DoR)

---

## Definiert wie etwas den Sprint **BETRITT**

- Vereinbarung, wann etwas für Umsetzung bereit ist.
- **READY** ist in der Verantwortung des Product Owners
- Der Product Owner erarbeitet eine Definition of Ready, das Team akzeptiert oder verhandelt nach.
- Planning: Team kann Stories, die nicht READY sind ablehnen!
- Beispiel:
  - Geklärt: Warum? Was? Wie?
  - Granularität ok, testbar, ...
- Weniger als **1%** aller Scrum Teams arbeiten konsequent mit einer Definition of **READY**
  - **Jeff Sutherland**: A Practical Roadmap to Great Scrum: A Systematic Guide to Hyperproductivity (<http://www.scrum-day.de/vortraege/2010-11-17-ScrumDay-Practical%20Roadmap.pdf>)

# Scrum Board



# Bekanntheitsgrad

---

- Definition of Ready / READY State
  - Thema auf Konferenzen seit **Anfang 2009**
  - Trotzdem bisher noch nicht wirklich im Fokus!

## Ergebnis einer Stichprobe

Mailingliste „scrumdevelopment“

- <http://groups.yahoo.com/group/scrumdevelopment>
- „definition of done“: **1001**
- „definition of ready“: **6**
- „ready state“: **14**

# Bekanntheitsgrad

---

- Bekanntheit == Anwendung?!
- READY außerhalb der **Komfortzone** der meisten Teams?
  - Viele Scrum-Implementationen fokussieren stark auf Umsetzungsphase
  - Viele Teams unterschätzen Bedeutung von READY
    - Es wird gejammert, aber dann passiert nichts.
  - Für Product Owner ist READY aufwändig! Arbeit!
    - Und die meisten POs sind chronisch überarbeitet!

# Bedeutung

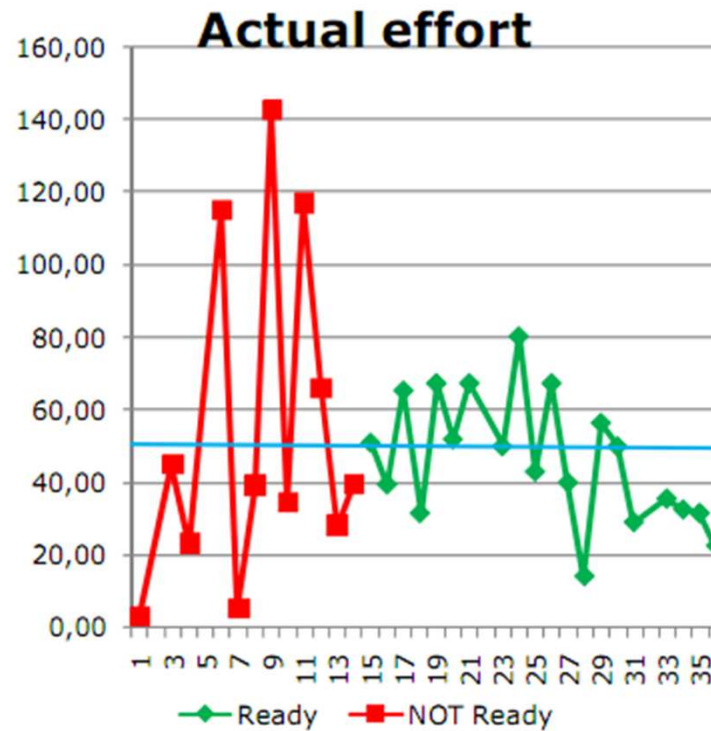
---

- READY wichtiges Element, für eine hohe **Produktivität**
- Jeff Sutherland: **Hyperproductivity**
  - Agile teams that are 4 to 8 times more productive than average teams in the industry. Nur wenige Teams erreichen dies.
  - First doubling of velocity with **DONE**
  - Second doubling with **READY** state
  - Third doubling with **Self-Organization**

<http://www.scrum-day.de/vortraege/2010-11-17-ScrumDay-Practical%20Roadmap.pdf>
- Weniger als **1%** aller **Scrum Teams** arbeiten konsequent mit einer Definition of **READY**

# Bedeutung

## Bedeutung von READY auf den Aufwand / Varianz



Objective: 50h

Quelle: [http://agile2009.org/files/session\\_pdfs/Going%20from%20Good%20to%20Great%20with%20Scrum%20Session.PDF](http://agile2009.org/files/session_pdfs/Going%20from%20Good%20to%20Great%20with%20Scrum%20Session.PDF)

# Ausflug:

## HYPERPRODUCTIVITY WITHIN SCRUM

---

### Basic Truths about Hyperproductive Scrum

- Everyone must be trained in Scrum framework
- Backlog must be **READY** before taking into Sprint
- Software must be **DONE** at the end of the Sprint
- Pair immediately if only one person can do a task
- No Multitasking
- Physical Scrum Board
- Short sprints (often 1 week)
- Burn down Story points only
- Everything (including support) is **prioritized** by PO
- Servant leadership – it's not about you

**Jeff Sutherland:** A Practical Roadmap to Great Scrum: A Systematic Guide to Hyperproductivity (<http://www.scrum-day.de/vortraege/2010-11-17-ScrumDay-Practical%20Roadmap.pdf>)

---

# DEFINITION OF READY



# Definition of READY

---

## Kurz:

- Story ist **READY**, wenn Team versteht was gemeint ist und es so umsetzen kann, dass es **DONE** ist.
- **READY** ist es, wenn das Team sagt  
„Ja! Jetzt verstehen wir's!“

# Definition of READY

---

## Etwas genauere Beschreibung von READY

- Anforderung verständlich beschrieben:
  1. **Warum?**
    - Zielsetzung / Problemstellung klar,
    - Geschäftswert verstanden
  2. **Was?**
    - Gewünschtes Ergebnis verstanden, Akzeptanzkriterien formuliert
  3. **Wie?**
    - Implementierungsstrategie/-konzept bekannt
    - Kosten / Größe bekannt

Zuerst WARUM, dann WAS, dann WIE!

# Definition of READY

---

## Etwas genauere Beschreibung von READY

### Details

- **Abhängigkeiten** identifiziert und soweit möglich aufgelöst
  - Stakeholder identifiziert und involviert
    - Entscheidungsträger, andere Teams, Technik-Verantwortliche
  - Risiken identifiziert und soweit möglich aufgelöst
- Aufgabe mit Team **besprochen** und geschätzt
- **Größe** ist handhabbar
  
- „Fertig für das Planning Meeting“

# Weitere Definitionen

---

## Roman Pichler:

- Klar
  - Gemeinsames Verständnis was genau gemeint ist
- Testbar
  - Es gibt klare Entscheidungskriterien, ob etwas erfolgreich umgesetzt ist oder nicht.
- Handhabbar
  - Größe muss stimmen (nicht zu groß/klein)
  - Komplexität muss handhabbar sein
    - Unsicherheit, Abhängigkeiten

<http://www.romanpichler.com/blog/product-backlog/the-definition-of-ready/>

# Kontext beeinflusst DoR

---

- Die DoR ist abhängig von den organisatorischen Rahmenbedingungen
- **Einfache DoR:**
  - Team kann fokussiert arbeiten
  - kennt sich in Thema fachlich und technisch gut aus
  - wenige Stakeholder/Einflussfaktoren
- **Umfangreiche DoR:**
  - Risiko von Ablenkung groß
  - Team muss sich Thema erarbeiten (Diskussionen!)
  - Komplexe Stakeholderstruktur
  - viel Legacy-Software
  - viele Abhängigkeiten, viel Unklarheit

# Beispiel: Sehr einfache DoR

---

- User Story Format
  - As a ROLE I ...
- Team versteht die Story und die Akzeptanzkriterien
- Team glaubt, dass es Story in einem Sprint umsetzen kann

# Beispiel: Einfache DoR

---

- User Story Format
  - As a ROLE I ...
- Team versteht die Story und die Akzeptanzkriterien
- Team glaubt, dass es Story in einem Sprint umsetzen kann
- Layout/Design in Wireframe
- Stakeholder und Abhängigkeiten benannt
- How2Demo / Akzeptanzkriterien

# Wie wird's READY?

---

- Definition of READY erarbeiten
  - Vorgaben, was READY bedeutet
  - Aufgaben für Product Owner und Stakeholder
- Vereinbarung, wie Team den PO bei Klärung unterstützt.
  - Team liefert DONE und unterstützt PO bei READY
  - Team weist PO früh auf Klärungsbedarf hin
- **Ziel:** Es gibt immer genügend READY Stories
  - 1-2 x Team Velocity

# Flow to READY, Iterate to DONE

---

- Argumentation:

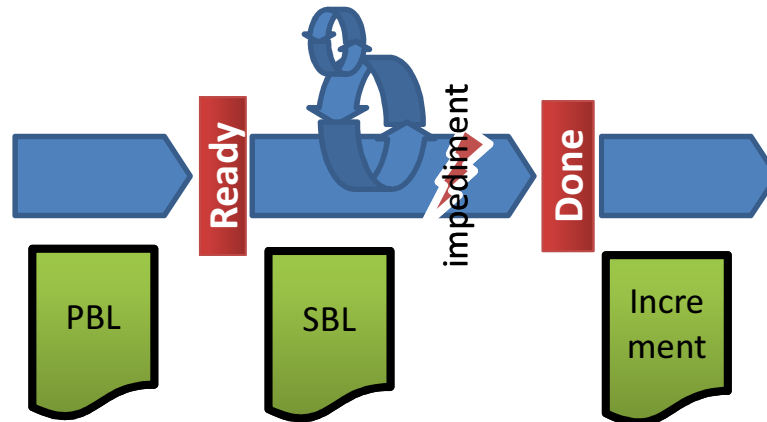
Arbeit PO unterscheidet sich von Arbeit des Teams:

- Anfragen kommen konstant, immerwährende Überarbeitung des PBL
- Kontinuierliche Arbeit, weniger iterativ



vgl. auch: <http://blog.xebia.com/2009/07/flow-to-ready-iterate-to-done/>

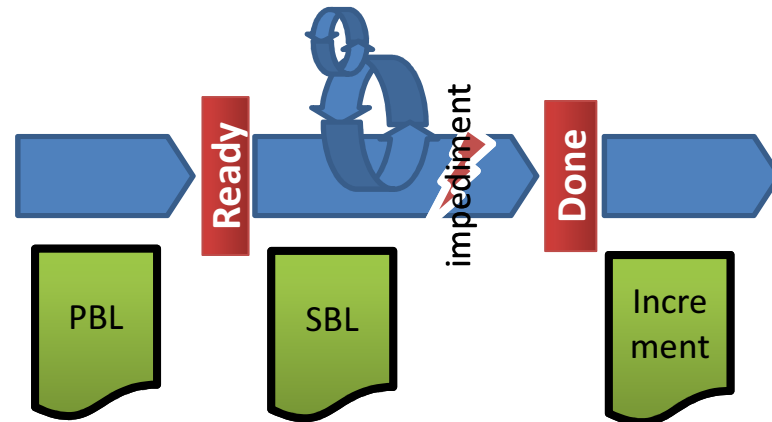
# Iteratives Produktmanagement



## Story: IDEE!

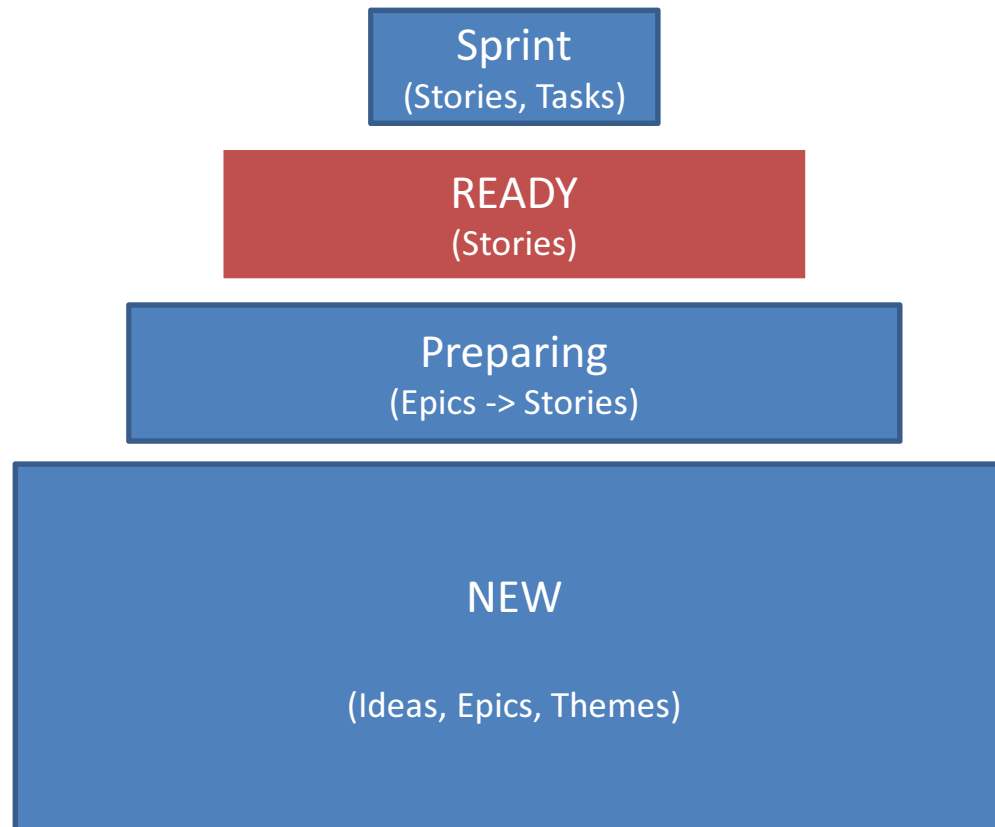
### Tasks:

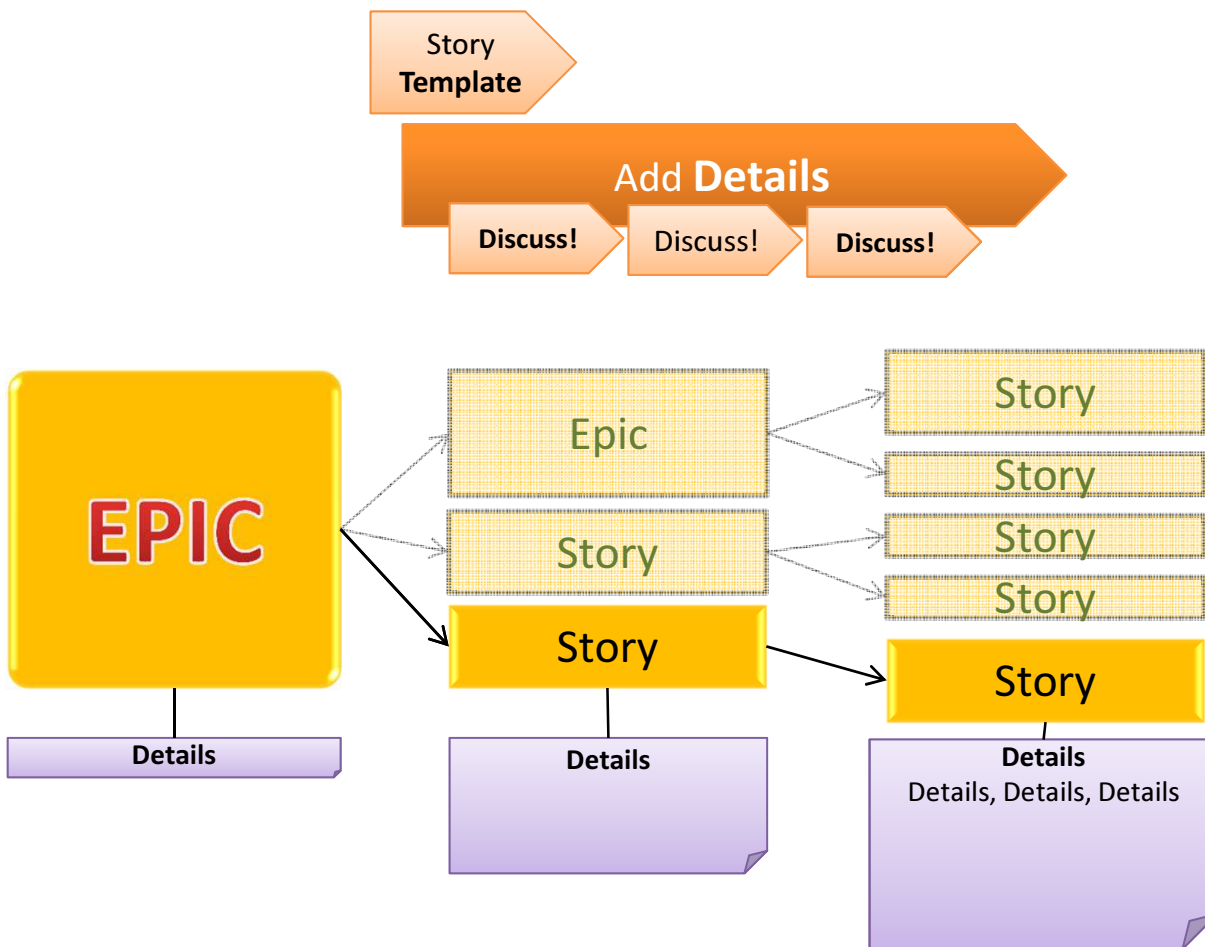
- Marktforschung
- Marktsegmentierung
- Marktstudie
- Kundenbefragung
- Konkurrenzanalyse
- SWAT Analyse



# Ein Blick ins Backlog

---

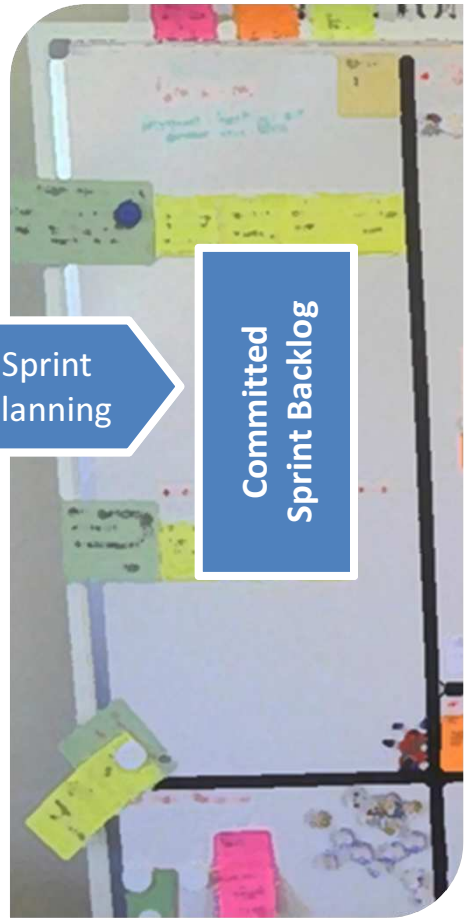




**READY**



**Committed Sprint Backlog**



# Life of a User Story

---

## Phase 1: Neue Idee / Epic

- Anforderung taucht auf und wird als Epic festgehalten

## Phase 2: User Story schreiben

- **WARUM:** Motivation beschreiben
  - Warum machen? Geschäftswert? Unsere Ziele?
- **WAS:** Anforderungen verständlich beschreiben
- **Check: Richtiges Team für die Umsetzung?**
- **WIE:** Konzept / Idee wie es realisiert werden soll
  - Entsteht bei Diskussion mit Team und Stakeholdern
- **DETAILS:**
  - Stakeholders identifiziert, dokumentiert und informiert
  - Voraussetzungen: Identifiziert und dokumentiert
  - Risiken: Identifiziert, beschrieben und Gegenmaßnahmen eingeleitet
  - Annahmen und Unklarheiten dokumentiert

# Life of a User Story

---

## Phase 3: User Story ist **READY**

- **Größe:** US passt in einen Sprint und ist gut handhabbar
- **Fragen,** Annahmen, unklare Aspekte geklärt
- **Konzept:** Implementierungsansatz diskutiert und dokumentiert
- **Schätzung:** Möglich und durchgeführt

# Template / Checkliste

---

- Hilfsmittel, um Stories READY zu bekommen
- Beispiel: Story Template
  - Hilft PO/PMs dabei eine gewisse Struktur einzuhalten – auch für mögliche Details.
  - Checklisten-Charakter („habe ich daran gedacht?“)
  - Details werden im Verlauf hinzugefügt, einige werden erst im Planning oder im Sprint ergänzt.
  - Template als Guideline, nicht als strenge Vorgabe
  - Story „lebt“ von Idee bis zur Umsetzung. Bis zum Review werden Details ergänzt.

# Beispiel für ein Template

---

- Name / Überschrift
  - Kurzer, einprägsamer Name
- Beschreibung
  - Story-Format (As a <type of role>, I want <some goal> so that <some reason>).
  - Weitere Beschreibung der Anforderung
- Motivation / Benefit / Wert
  - Warum soll Anforderung umgesetzt werden?
  - Quantifizierbare Benefits?

# Beispiel für ein Template

---

- Vorbedingung / Abhängigkeit / Annahmen
  - Dokumentation von Stakeholdern, offenen Fragen, Annahmen, Abhängigkeiten, Risiken
- Notizen / Konzept / Technische Details / Tasks
  - Notizen des PO/PM, die hilfreich sein könnten
  - Notizen aus Estimation, Planning
  - Notizen zur Konzeption / Umsetzung
  - Auflistung der Tasks
- How2Demo / Akzeptanzkriterien
  - Akzeptanzkriterien und Demonstration beschreiben

# Zusammenfassung

---

- 2 stabile Zustände in Scrum: READY & DONE
- READY: Quality Gate für Themen die einen Sprint betreten. Product Owner Qualität.
- DONE: Quality Gate für Themen, die einem Sprint verlassen. Team verantwortet Qualität.
- Starke Verbesserung der Produktivität durch READY
  - Jeff Sutherland: Hyperproductivity
- READY noch nicht weit verbreitet
- Beispiele für eine Definition of READY
- Zustände von Stories im Product Backlog

# Kontaktinformationen

---

## Informationen

- [www.agile-rescue.com](http://www.agile-rescue.com) (Consulting)
- [www.agile-nearshoring.com](http://www.agile-nearshoring.com) (Teams)
- [www.armerkater.de](http://www.armerkater.de) (Meinung)

## Kontakt

- [felix.ruessel@agile-rescue.com](mailto:felix.ruessel@agile-rescue.com)
- +49-721-7540-6985

# Quellen / Referenzen

---

- <http://www.romanpichler.com/blog/product-backlog/the-definition-of-ready/>
- <http://scrum.jeffsutherland.com/2009/07/ready-dynamic-model-of-scrum.html>
- [http://agile2009.org/files/session\\_pdfs/Going%20from%20Good%20to%20Great%20with%20Scrum%20Session.PDF](http://agile2009.org/files/session_pdfs/Going%20from%20Good%20to%20Great%20with%20Scrum%20Session.PDF)
- <http://jeffsutherland.com/SystematicREADYREADYChecklist.pdf>
- <http://jeffsutherland.com/JakobsenScrumCMMIGoingfromGoodtoGreatAgile2009.pdf>
- <http://blog.xebia.com/2009/07/flow-to-ready-iterate-to-done/>