

# Agile for the masses

Java Forum 2022 / moritz löser

Für alle die „agil“ oder  
Scrum blöd finden

# Wer bin ich was mache ich hier

- Sponsored Talk :-D
- Entwickler
- „Notgedrungen“: Scrum und Agile Coach
- Denkanstoß, Diskussionsgrundlage

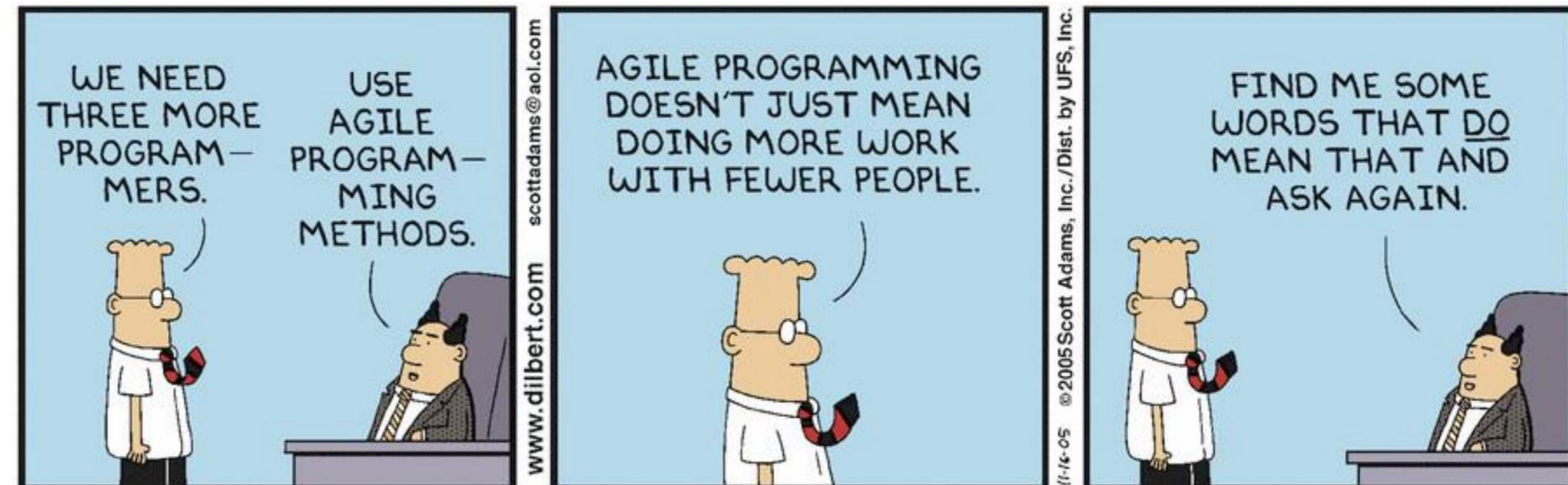
# Was soll das – noch ein Vortrag über agil?

- Unversöhnliche drei Fronten von Entwicklern:
  1. die nur noch „agil“ arbeiten wollen und können (ich)
  2. die „agil“ in ihrem Alltag erleben und es als Last empfinden – es steht im Weg
  3. die sehr zufrieden mit ihrer Situation sind – so lange man ihnen mit „agil“ fernbleibt
- Ziel die letzten zwei Gruppen ins Boot holen
  - Mindestens gegenseitig etwas mehr Verständnis aufbringen
- Allgemein „agil“ – teilweise mischen sich Dinge mit Scrum



# Missverständnisse

- Hypothese 1: Gruppe 2 hat nie agil gearbeitet, Gruppe 3 arbeitet bereits agil
- Hypothese 2: Unterschiedliches Verständnis von agil
- Was ist agil?



# Vorschlag: Agiles Manifest

„Wir erschließen bessere Wege, Software zu entwickeln, ... haben wir diese Werte schätzen gelernt:

- Individuen und Interaktionen mehr als Prozesse und Werkzeuge
- Funktionierende Software mehr als umfassende Dokumentation
- Zusammenarbeit mit dem Kunden mehr als Vertragsverhandlung
- 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.“

- Wir: Softwareentwickler/ Programmierer

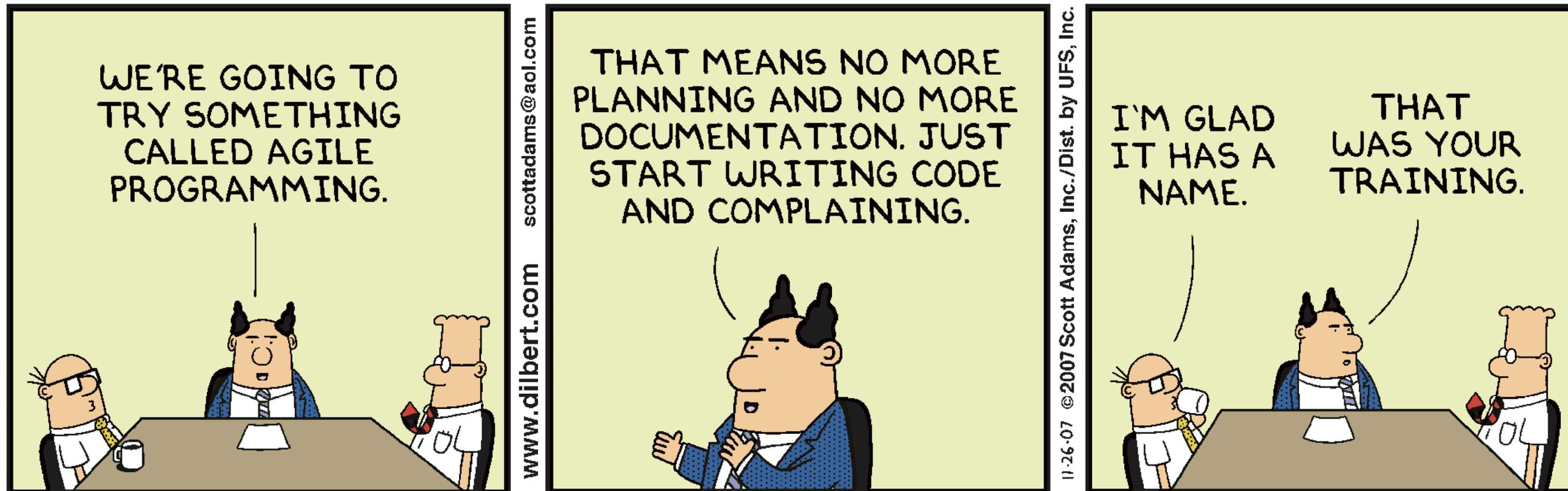
# Etwas konkreter

„Wir folgen diesen Prinzipien:

- ...
- Funktionierende Software ist das wichtigste Fortschrittsmaß.
- Einfachheit -- die Kunst, die Menge nicht getaner Arbeit zu maximieren -- ist essenziell.
- Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen.
- Agile Prozesse fördern nachhaltige Entwicklung.  
Alle sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.
- ...“

# Warum so viele Praxisprobleme?

- Unterschiedliche Vorstellung von agile
- Agiles Manifest für viele unbekannt – leider auch Entwickler
- Andere Erwartungen/ Vorstellungen -> Missverständnisse



# Indizien für falsches Verständnis

- Unzufriedenheit -> Personalfluktuation
- Parteien außer Fachleuten und Entwickler nehmen direkten Einfluss (Architekten, Manager, QA)
- Stark eingeschränkte Kontrolle der Entwickler über Vorgehen, Werkzeuge, Prozesse
- Kein Verständnis / Überzeugung der Entwickler vom fachlichen Ziel (Zweifel werden nicht gehört)
- Kein echtes schnelles Feedback von Nutzern/ Nutznießern (z.B. „externer PO“)
- „Retro macht keinen Sinn“

# Indizien (Zombie Scrum) als Fragen

- Wählen Entwickler und Fachleute alle Termine selbst?
- Wählen Entwickler all ihre Werkzeuge selbst?
- Wählen Entwickler die Architektur?
- Wählen Entwickler alle Maßnahmen zur QS?
- Wählen die Entwickler Ihre Arbeitslast (wie viel und wie lange)?
- Dienen existierende Vorgaben (erstellt von Entwicklern) der Qualität und der Zusammenarbeit?
  - Gibt es Mittel Wirksamkeit zu prüfen?
- Gibt es Möglichkeiten alle Aspekte bei Bedarf anzupassen?

# Agile/Scrum Anti-Pattern

- PO ist extern  
(einem Menschen der das Produkt versteht Scrum / agil beizubringen ist einfacher einem Scrum-Profi das Produkt beizubringen)
- QS ist separiert in Zeit und Personen (Testteam/ Tester/ Testphase)
- Architektur ist separat vorgegeben / Es existieren „Architekten“
- Akzeptanzkriterien welche der PO nicht überprüfen kann

# Was tun? (Entwickler)

- Beansprucht „agil“ im Sinne des Manifests – werft andere Vorstellungen über Bord („Scrum“ impliziert nicht allein agil)
- Beansprucht Kontrolle, übernehmt Verantwortung
- Lehnt Verantwortung ab über Dinge die ihr nicht kontrolliert (z.B. wenn jemand anderer die QS Maßnahmen bestimmt lehnt die Verantwortung über Qualität ab)
- Haltet zusammen als Team (manche vertragen mehr Verantwortung als andere)
- Ihr seid sehr wichtig, aber nehmt euch nicht zu wichtig :-P
- Seid bestimmt aber freundlich



# Was tun? (Management)

- Projekt -> Produkt
- Erklärt was die Ziele und Visionen sind (das Warum) ...
- ... und fragt wie ihr euren Kollegen helfen könnt sie (für euch) zu erreichen
- ... und gebt ihnen was sie brauchen (Verantwortung, Kontrolle, Material, Hilfe)
- Der nachhaltige „Nutzen“ jedes einzelnen maximiert sich bei maximaler Zufriedenheit



# Agil werden in Kürze

- Projekt -> Produkt
  - Schöpft Wert, wird gemessen
  - Laufzeit unbegrenzt
  - Möglichst schnell Feedback von Nutznießern, Fester Teil des Prozesses
- Zwei Parteien pro Produkt/Team: Fachleute („PO“), Entwickler
  - Alle andere unterstützen nach Vorgabe des Teams
- Fachleute überzeugen Entwickler vom Produkt („committent“)
- Team entscheidet über eingesetzte Werkzeuge und Praktiken (und passt ggf. an)
- Fachleute organisiert Feedback von Nutzern

# Fragen? Diskussion!

- Agil verstanden?
- Realitätscheck?
- Motiviert „agil“ im eigenen Umfeld durchzusetzen/ dafür zu kämpfen/ auszuprobieren?
- Grenzen von „agil“

➤ Vielen Dank!