

Labor

Agile Software Entwicklung

Informationsveranstaltung

Stephan Kiesling, Jil Klünder

{stephan.kiesling, jil.kluender}@inf.uni-hannover.de

15.10.2015

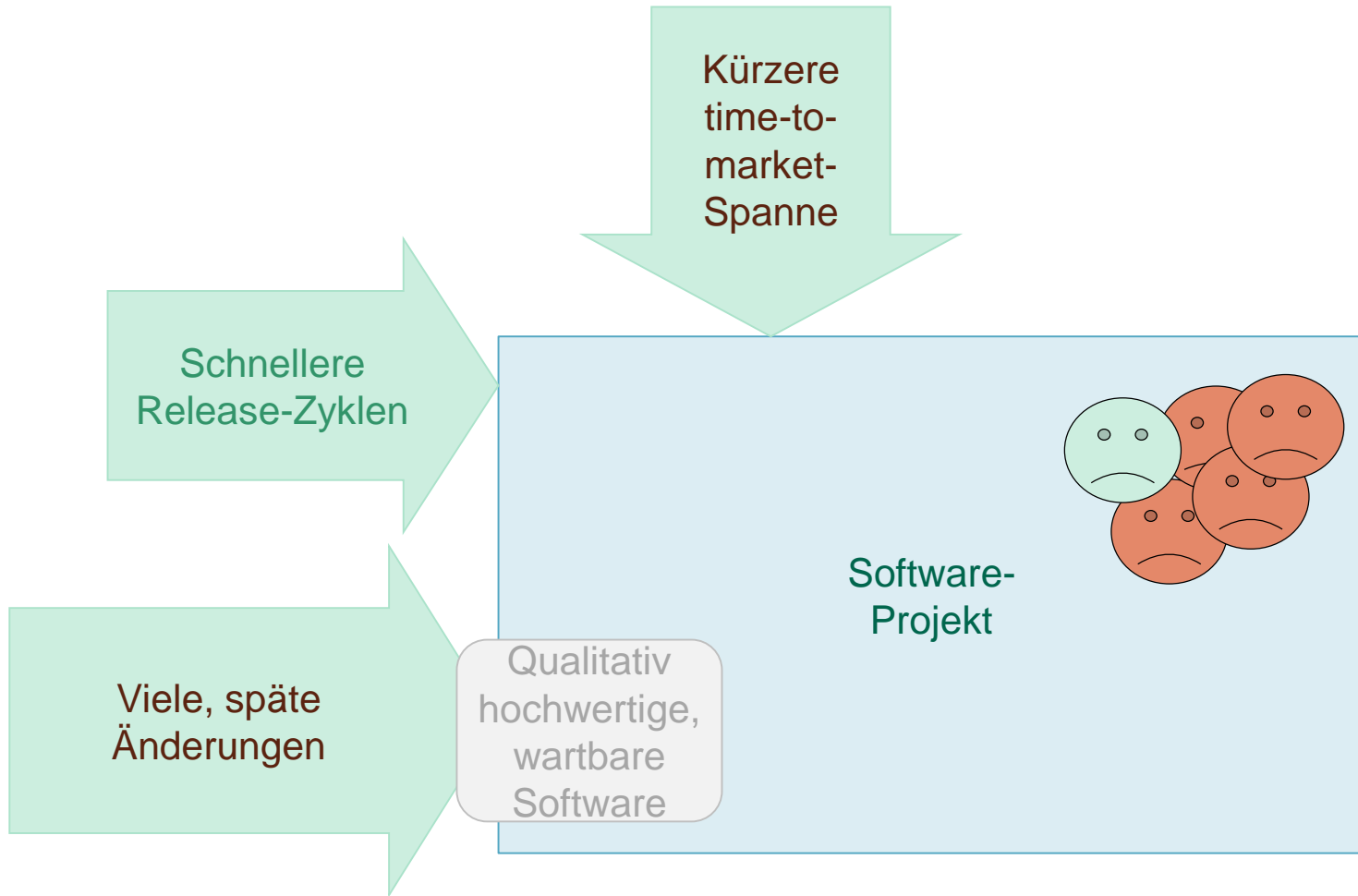


Jil Klünder

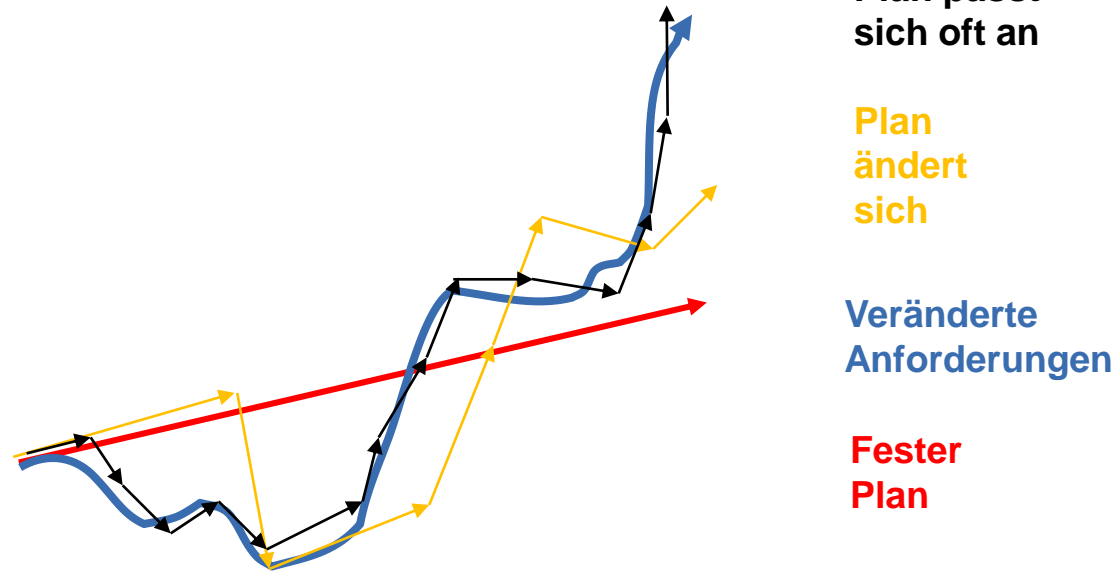


Stephan Kiesling

Probleme in heutigen Projekten



Planung hat ihre Grenzen!



- Daher: Nur nächste Schritte genau planen, spätere erstmal grob.
- Voraussetzung: Ständige Fortschrittskontrolle und Feedback

“*Agilität* ist die Fähigkeit einer Organisation, flexibel, aktiv, anpassungsfähig und mit Initiative in Zeiten des Wandels und Unsicherheit zu agieren.” (onpulson.de)

Agile Softwareentwicklung ist eine **Einstellung** und kein Prozess. Werte und Prinzipien stehen im Vordergrund.

Embrace Change

“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, Alistair Cockburn
Martin Fowler, Jim Highsmith, Ron Jeffries
Ken Schwaber und 11 andere

www.agilemanifesto.org

Ziel der Veranstaltung

- ... ist es, auf spaßvolle Art und Weise...
 - Agile Methoden besser zu *verstehen*
 - in einem XP-Projekt *gearbeitet* zu haben
 - Konsequenzen aus dem XP-Aufbau zu *erleben*
- Es geht *nicht* darum...
 - programmieren zu lernen
 - wild zu „hacken“

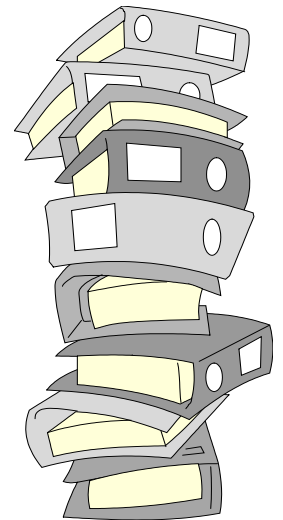
eXtreme Programming

- führt Praktiken ein, die die Werte unterstützen
 - Continuous Integration für funktionierende Software
 - On-Site Customer für Kommunikation

SCRUM

- Gerüst, das den Grundablauf agil hält
 - kurze Iterationen (Sprints), die gemeinsam geplant werden
 - Zusammenspiel von Rollen

- - Änderungen auch für AG schwer durchsetzbar

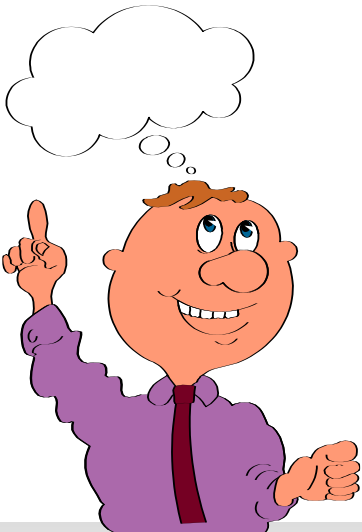


Wie viel Planung braucht ein Projekt? (B. Boehm)

ad hoc

- + wenig Planungsaufwand
- + individuelle Freiheit
- - Ergebnis unvorhersehbar
- - abhängig von “Helden”

feingranulare
Verträge



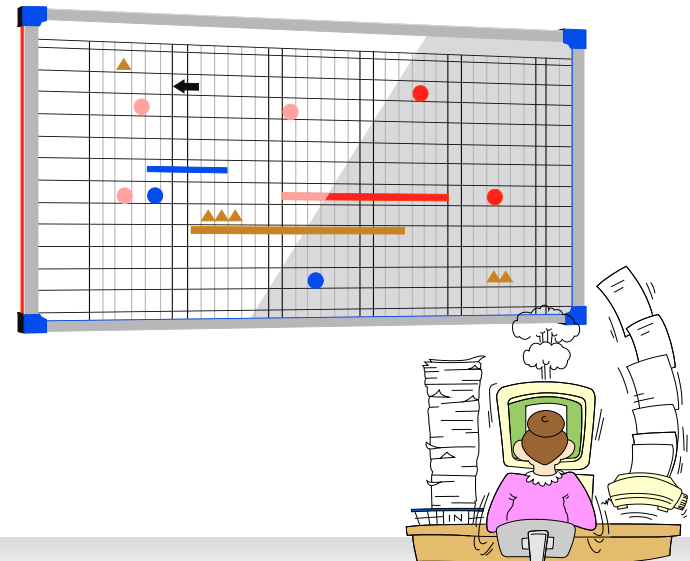
Wie viel Planung braucht ein Projekt? (B. Boehm)

Meilenstein- u. Plangetrieben

- + langfristige Vorhersagen
- + gute Zustandskontrolle
- - Änderungen aufwändig
- - unrealistische Annahmen
schwer zu eliminieren

ad hoc

feingranulare
Verträge



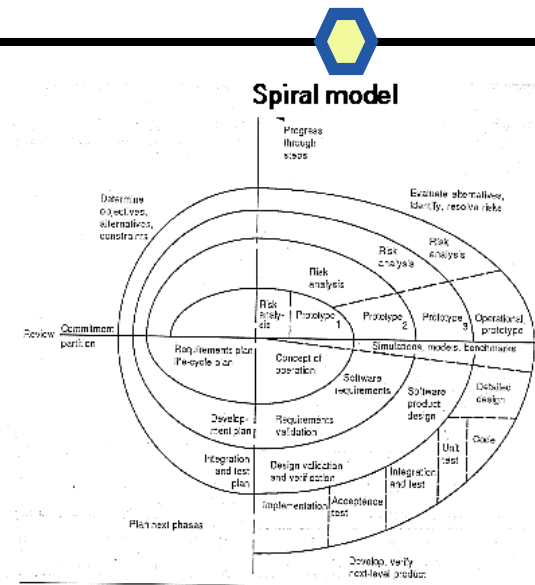
Wie viel Planung braucht ein Projekt? (B. Boehm)

Meilenstein- u. Risikogesteuert

- ++ Risiken aktiv ausgeräumt
- + Teilergebnisse früh
- kaum langfristig planbar
- relativ aufwändig

ad hoc

Meilenstein- u. feingranulare
Plangetriebenen Verträge



Wie viel Planung braucht ein Projekt? (B. Boehm)

eXtreme Programming

- + früh Kernergebnisse
- ++ Änderungen unproblem.
- in sehr großen Proj.
Zusatzplanung nötig
- viel Selbstdisziplin nötig

ad hoc

Meilenstein- u.
Risikogesteuert

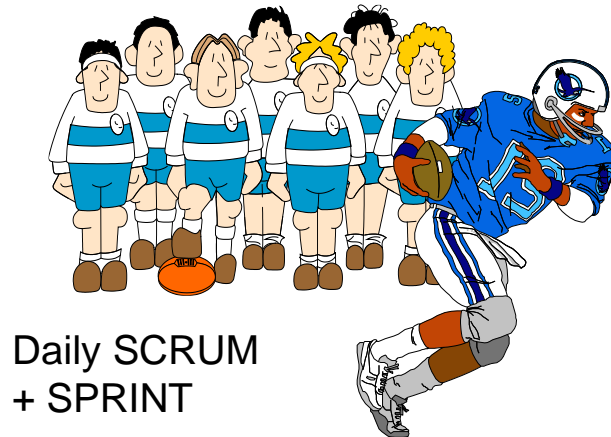
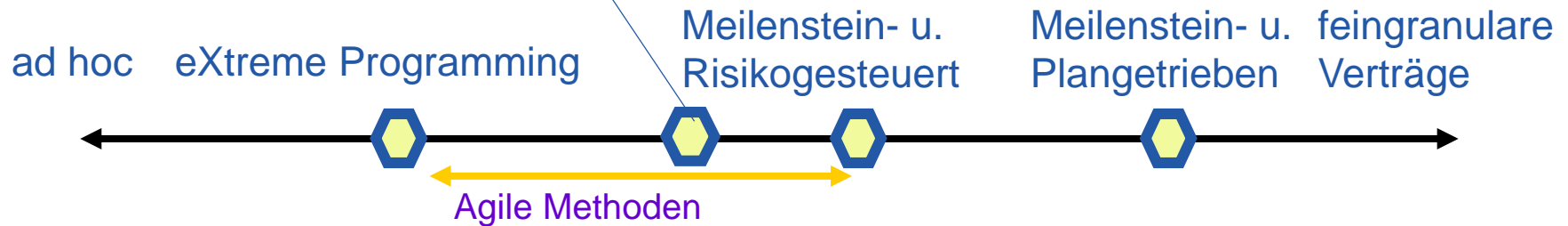
Meilenstein- u. feingranulare
Plangetrieben Verträge



Wie viel Planung braucht ein Projekt? (B. Boehm)

Agiles Beispiel: SCRUM

- + mittelfristig geplant
- + reagiert schnell
- hängt an MA-Qualifikation
- Endprodukt nicht spezifiziert



Daily SCRUM
+ SPRINT





Left whiteboard section with pinned documents and a diagram.

Pinned Documents:

- Top left: Document with diagrams and text.
- Top right: Document with diagrams and text.
- Middle left: Document with text and diagrams.
- Middle right: Document with text and diagrams.
- Bottom left: Document with text and diagrams.
- Bottom right: Document with text and diagrams.

Diagram:

- A hand-drawn diagram showing a process flow.
- Labels include: "100%", "Übungsaktion", "Übungsaufgabe", "Hauptübung", "Vorlesung", "Zeit 15", "Zeit 30".
- Arrows indicate the flow between different stages or tasks.

Right whiteboard section with a central mind map and several diagrams.

Central Mind Map:

- Central node: **PLP**
- Branches:
 - Vorlesung** (top left)
 - Themenbereich** (top left, below Vorlesung)
 - Thema** (top left, below Themenbereich)
 - Aktion/Ausübung** (top right, below V.)
 - Client-Sum** (top right, crossed out)
 - ORM** (top right, circled)
 - Altklausur** (top right, circled)
 - Mg** (top right, circled)
 - Datenbank** (middle right)
 - Export** (middle right, circled)
 - Kalender** (middle right)
 - Doc-Dokumente** (bottom right, circled)
 - Text-Editor** (bottom right, circled)
 - Profile-Kategorie** (bottom right)

Diagrams:

- Top Left:** A diagram showing a hierarchy: **Vorlesung** → **Themenbereich** → **Thema**.
- Top Right:** A diagram showing a hierarchy: **V.** → **Aktion/Ausübung**.
- Middle Left:** A diagram showing a form with fields: **Thema: Coverage**, **Zeit: 15 min**, **Aktion:** (with a list), **Bele**, **Witz**, **OK**.
- Middle Right:** A diagram showing a form with fields: **Thema: Coverage**, **objekt:** (with a list), **Gruppe:** (with a list), **Bele**, **Witz**, **OK**.
- Bottom:** A diagram showing a form with fields: **Bele**, **Witz**, **OK**.





Organisatorisches

15.10.2015

Anforderungen

- Programmierkenntnisse in *Java*
 - Es ist kein Programmierkurs
- Wille zur *Teamarbeit*
 - Es wird in wechselnden Paaren und im gesamten Team gearbeitet werden müssen
 - Kommunikation!
- *Pünktlichkeit*
 - Verspätungen muss das *ganze Team* nachholen
- *Selbstdisziplin*
 - Einige Praktiken muss man per Fleiß verinnerlichen

Zeitliche Organisation

- Die Veranstaltung findet als *Blockveranstaltung* statt
 - 08.02.2016 bis 12.02.2016
 - Jeweils von 09:00 – 18:00 Uhr
 - Mittagspause von 12:00 – 13:00 Uhr
- Vor dem Block findet das Labor in regelm. Terminen statt
 - Vorbereitende Kurse
 - Einarbeitung
 - Donnerstags, jeweils von 09:00 bis 12:00 Uhr
- Nach der Blockveranstaltung gibt es eine *Nachbereitung*
 - 18.02.2016 von 09:00 bis 12:00 Uhr
- Die genauen Termine stehen auf der Webseite des SE!

Alle Termine sind Pflichttermine!

Datum	Inhalt
15.10.	Einführungsveranstaltung
22.10.	Frei
29.10.	Frei
05.11.	Frei
12.11.	Frei
19.11.	Einführung in die agilen Methoden Teil 1
26.11.	Praktische Übung: Agile Hour Teil 1
03.12.	Einführung in die agilen Methoden Teil 2
10.12.	Praktische Übung: Agile Hour Teil 2
17.12.	Versionsmanagement & Test First
24.12.	Frei
31.12.	Frei
07.01.	[wird noch bekannt gegeben]
14.01.	Einführung in die Entwicklungstechnologien
21.01.	Spikes & Vorstellung des Projekts durch den Kunden
28.01.	Spikes
04.02.	Planning Game
08.02. – 12.02.	Blockwoche
18.02.	Bugfixing & LIDs

Anmeldeverfahren

- Es können max. *18 Studenten* teilnehmen
- Priorisierung:
 - Master vor Bachelor
 - Anwesende Studierende vor abwesenden
 - Studenten anderer Studiengänge
 - Danach: Losverfahren
- Anmeldung:
 - Gleich im Anschluss
 - Eine spätere Anmeldung ist nur möglich, wenn Plätze frei bleiben.