

5.– 8. September 2011
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Anleitung zum Ruinieren eines Scrum-Teams

Wie man in kürzester Zeit jegliche Agilität in der Softwareentwicklung verliert

Udo Wiegärtner

complement AG, Nürnberg



Stoppt Scrum!



- ...Pläne nichts mehr wert sind?
- ...Softwareentwickler unsere Prozesse mit Füßen treten?
- ...wir dauernd mit dem Auftraggeber reden anstatt zu implementieren?
- ...wir auf jede Veränderung reagieren würden?
- **Das wäre doch Anarchie, oder?**
- **Agile Anarchie!**
- **Agile Scrum Anarchie!**



- **Agile Entwicklung wird uns nicht kampflos besiegen!**
- **Scrum muss weg!**
- Sie brauchen Scrum ja nur mal rückwärts lesen



Udo Wiegärtner

- **Fa. complement AG, Nürnberg**
- **Bereichsleiter Software Engineering / Berater**
- **Kämpfe in allen „Schützengräben“**
 - Softwareentwickler
 - Product Manager
 - Entwicklungsleiter
- **Ich war einer von „denen“**
(vielleicht bin ich es ja immer noch)
- **Scrum Ausbildung**
 - Scrum Master (Scrum Alliance)
 - Scrum Master I + II (scrum.org)
 - Professional Scrum Developer Trainer (scrum.org / Microsoft)
- **Scrum seit 2006**
- **Einführung von Scrum in mehreren Firmen und etlichen Teams**

Zerbrechliche Agilität





- **Als Scrum Skeptiker**
 - ...die wunden Punkte von Scrum-Teams identifizieren
 - ...Einführung und Betrieb von Scrum nachhaltig stören
- **Als Scrum Befürworter**
 - ...ihr Scrum verbessern, indem Sie den Dozenten komplett ignorieren
- **...nicht behaupten, ich hätte Sie nicht gewarnt**



- **Scrum in 3 Minuten**
- **Stören von Scrum**
 - Wann stören?
 - Wo stören?
 - Rollen
 - Artefakte
 - Timeboxes
 - Regeln
 - Tools



- **Scrum = Gesunder Menschenverstand mit hübschen englischen Begriffen**
- **Scrum = Instinktives Teamverhalten, falls im Projekt mit Rücken zur Wand**
- **Und dieses Verhalten besteht oft aus...**



- **Das Wichtigste zuerst**
- **Lauffähige Software**
- **Sachen gleich „richtig“ fertig machen**
- **Überschaubare Zyklen**
- **Gute Kommunikation**
- **Kein Prozess-Schnickschnack**
- **Rückversicherung beim Kunden**
- **Alle für einen – einer für Alle**
- **→ kommt Ihnen das bekannt vor?**

Und wie boykottiere ich das?



Ansatzpunkte sind überall

- Einführung und Laufzeit
- Management u. Entwickler
- Auftraggeber
- Alle Bestandteile von Scrum
 - Rollen, Artefakten, Timeboxes, Regeln
- Tools
- Nachhaltiges Stören sollte da ja wohl einfach sein, oder?
- Möge unser teuflisches Spiel beginnen...



- **Rückhalt im Management vermeiden**
- **Scrum als Allheilmittel anpreisen**
- **Team bloß nicht ins Boot holen**
- **Nur Begriffe ändern, z.B.**
 - Monate → Sprints
 - Personentage → Story Points
- **Sofort grundlegende Scrum-Regeln ändern, z.B.**
 - Sprint 3 6 Wochen
 - Daily Scrum täglich 1x pro Woche
 - ~~Streng priorisieren~~ Alle Features sind Prio 1a
- **Keine „Einschwingphase“ für Team**



Sabotageziel:

Scrum-Rollen



- **Product Owner**
- **Team**
- **Scrum Master**



- Will Scrum, denkt Wasserfall
- Rollenverschmelzung hemmt
 - Product Owner + Scrum Master = Projektleiter
- Product Owner nicht greifbar für Team
- Zu „busy“ fürs Daily Scrum
- Auftraggeber ist Freund Feind
- Schlampert bei Product Backlog (siehe spätere Folie)



- **Groooooße Teams sind behäbig (Worst Case: > 9 Entwickler)**
- **Tools anstatt mündliche Kommunikation**
- **Mut zur De-Spezialisierung nehmen (Rockstars)**
- **Keine Chance zur Eigenverantwortung geben**
- **Tester u. Business Analyst aus Team fernhalten**
- **Juniors und Seniors trennen**
- **Bloß kein Pair Programming**



- **Disziplinarischer Vorgesetzter?**
- **Arbeitsverteiler im Daily Scrum**
- **Löst Probleme selber anstatt Hilfe zur Selbsthilfe**
- **Nicht greifbar**
- **Vertrauen in Team ist überflüssig**
- **Zu weich für diese Welt**



Sabotageziel:

Scrum-Rollen

Sabotiert!





Sabotageziel:

Scrum-Artefakte



- **Product Backlog**
- **Sprint Backlog**
- **Burndown Chart**
- **?**



- Nicht vorbereitet zum Sprint Planning Meeting
- In Komponenten / Dokumenten denken
- Zu große User Stories
- Nicht in Aktoren denken
- User Story ersetzt komplettes Pflichtenheft 💣
- Nicht streng priorisiert
- Priorisierung nicht mit Kunden abgestimmt
- Kein zyklisches „Grooming“
- Nichtssagende User Stories *[Beispiel...]*



User Story laut Scrum-Fans:

- **Bestandteile:**
 - Titel
 - Beschreibung
 - „Als <Aktor> will ich <Feature> um zu <Begründung>“
 - Akzeptanzkriterien
 - Priorität
 - Business Value



Das Feature

- **Der Kunde will ein Gefühl dafür kriegen, wie viele Benutzer seine Webanwendung nutzen**



So würden die Scrum-Fans es formulieren

- **Titel:**
 - Mitloggen der Benutzerzahlen
- **Beschreibung:**
 - Als Administrator will ich die Benutzerzahlen der Weblösung XY auswerten können um bei Lastspitzen zusätzliche Hardware aktivieren zu können
- **Akzeptanzkriterien:**
 - Einfache Textdatei als Logfile genügt
 - Mindest-Inhalte: Max Anzahl der gleichzeitigen Benutzer je Stunde
 - Zeitraum letzte 30 Tage genügt
 - Aktualisierung 1x je Nacht
 - Speicherort des Logfiles muss den Admins und Projektmanagern zugänglich sein. Kein Zugriff für alle anderen
- **Priorität:**
 - 15/25
- **Business Value:**
 - 100/1000



Scrum-Saboteure würden es so machen

- **Titel:**
 - Webservice für Benutzertabelle
- **Beschreibung:**
 - Als Webservice will ich die Benutzer aus der Tabelle CurrentUsers aus der Datenbank auslesen und per Filestream in die Datei myLog.txt schreiben
- **Akzeptanzkriterien:**
 - Parameterliste WebService xyz
 - Name der Komponente: myWebservice
 - Weitere Details wie besprochen
- **Priorität:**
 - Ist Teil des Lastenhefts, muss deshalb unbedingt rein
- **Business Value:**
 - ?
- **Hell Yeah, Baby!**



- Altlasten aus vorherigem Sprint nicht mitführen („Debt“ / „undone work“)
- Nicht sichtbar im Daily Scrum
- Viele Tasks „in progress“
- Nicht visualisieren, warum „in progress“ (z.B. überall fehlt nur noch der Test)
- So würden „die da“ es vielleicht machen *[Beispiel...]*

Visualisierung Sprint Backlog für Scrum-Fans



Sprint 3

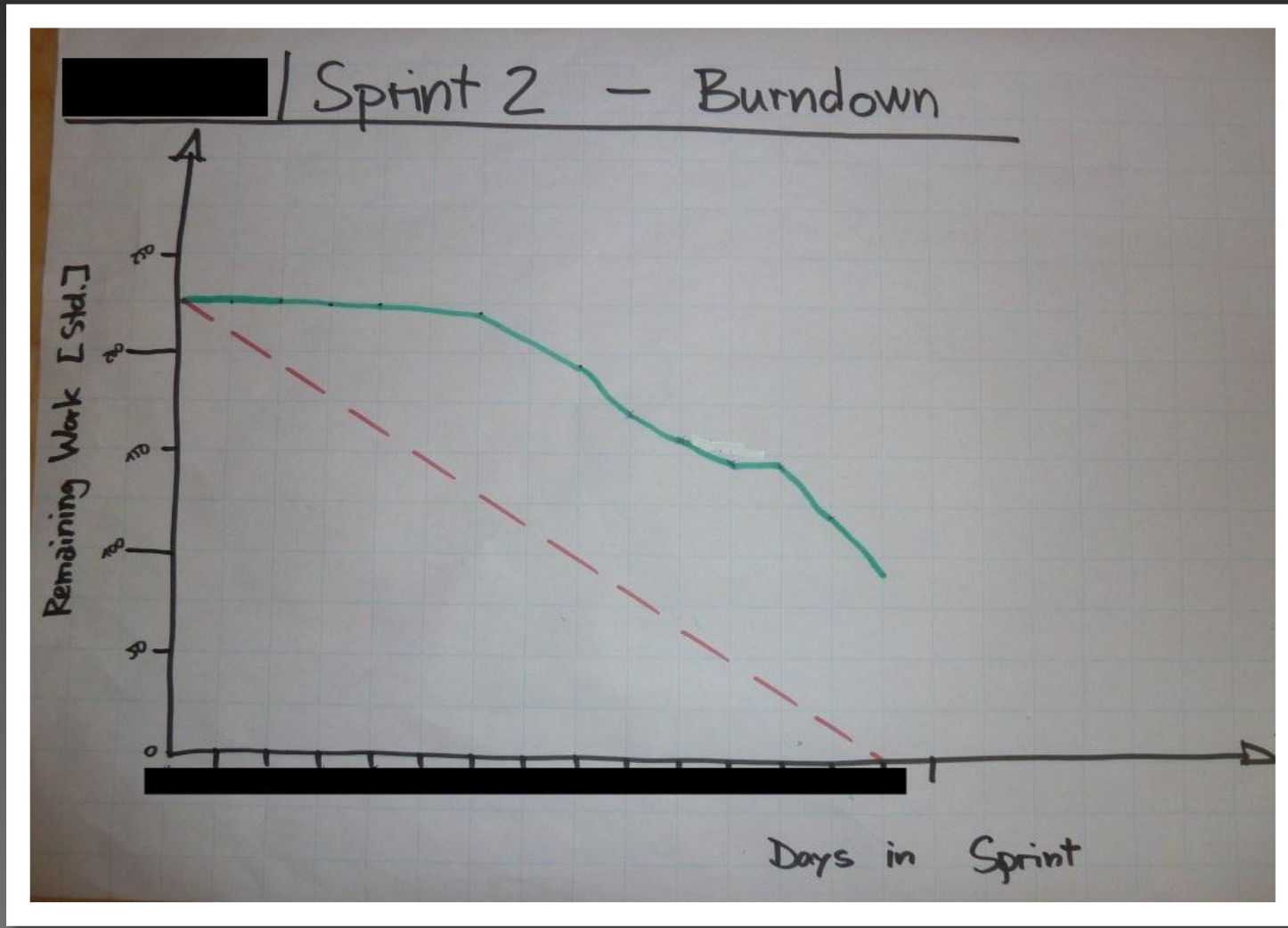
Prio	Story	Task	Designdok. Entw.-HB	Implement	Testplan	Testdurchf.	Status		
							Not Started	In Progress	Done
		Proj-Auswahl Rechtekonzept einbauen + Spec + Test		X	X	X			R
		-//- Bugfix		X					R
		Analyse Performance Bottlenecks		X	X	X			H
		Rechtekonzept einbinden		X	X	X			H
		Grid Selektionsseite einbinden		/				I	
		Refactoring Proj-Detailseite		/	X			I	
		Validierung Eingabefelder		/	X			R	
		Button-Funktionalität							
		Code Review							
		Analyse Performance Bottlenecks		X	X	X			H
		Benutzer Selektion + Detail + DB erweitern	X	X	X	X			A
		Grid einbinden		X	X	X			A
		Button-Funktionalität		X	X	X			A

Caching umschalten
@include action + CSS + Image Faktoren



- **Nicht sichtbar im Daily Scrum**
- **Als Management-Kontrollinstrument anstatt als Team-Helferlein nutzen**
- **Keine Konsequenz aus ungünstigem Burndown ziehen**
[Beispiel...]

Burndown Chart ohne Konsequenz





Sabotageziel:

Scrum-Artefakte

Sabotiert!





Sabotageziel:

Scrum-Timeboxes



- **Sprint**
- **Sprint Planning Meeting**
- **Daily Scrum**
- **Sprint Review Meeting**
- **Sprint Retrospective Meeting**
- **Genereller Sabotage-Tipp: Timeboxes grundsätzlich immer sprengen**

Scrum Timeboxes: Sprints verlangsamten



- **Warum nicht gleich 6 Wochen anstatt 3?**
- **Karenzzeit zwischen Sprints zum „Fertigmachen“**
- **Takt häufig wechseln**
- **Umpriorisieren während Sprint**
- **Sprint nie abbrechen**



- **Product Backlog nicht aktuell**
- **Kein/schlechtes Sprint-Ziel**
- **Diskussionen über den Sinn von User Stories**
- **Product Owner „anschießen“**
- **Große Stories nicht herunterbrechen**
- **Lead Developer / Rockstar schätzt alleine ab**
- **Planning Poker ist eh kindisch**
- **Endlos-Meetings**



- Was heißt schon „täglich“?
- „Hinsetzmeetings“ bremsen uuunglaublich
- Hindernisse (Impediments) versanden lassen
- Keine Konzentration auf die 3 Fragen
- Zu „weicher“ Scrum Master
- Sprint Backlog nicht sichtbar im Raum
- Alle lieben technische Detaildiskussionen
- Burndown nicht „live“ pflegen, Konsequenzen nicht mit Product Owner diskutieren



● Sprint Review

- Der Kunde stört nur
- Lead Developer / Rockstar präsentiert alles
- Product Owner blenden durch unfertige Stories

● Sprint Retrospective

- „Stuhlkreis“-Atmosphäre schafft Unbehagen ;-)
- Auf keinen Fall strukturiert vorgehen
- Nichts aufschreiben
- Maßnahmen vom letzten Retrospective nicht nachverfolgen



Sabotageziel:

Scrum-Timedboxes

Sabotiert!





Sabotageziel:

Scrum-Regeln

Scrum-Tools



● Scrum-Regeln

● Definition of Done (DoD)

- Durch Management diktieren anstatt durch Team
- Definition of Done weder leben noch nachjustieren
- Unsinnige Inhalte (z.B. Einchecken nicht Teil der DoD)
- *[Beispiel...]*

● Tools

- Supertool ersetzt Scrum-Grundverständnis
- Flipcharts eher meiden, da sehr effektiv
- Team Foundation Server:
 - Macht gute Teams viel besser
 - Macht schlechte Teams evtl. noch schlechter
- Urban Turtle: Viel zu preisgünstig, um gut zu sein *[Beispiel...]*

Urban Turtle (wieder was für Scrum-Fans)



The screenshot displays the Urban Turtle task board within the Visual Studio 2010 Team Web Access environment. The interface is organized into a Kanban-style board with columns for 'To Do', 'In Progress', and 'Done' for two distinct work items.

Work Item 8678:

- To Do - 0 work items:** Empty column.
- In Progress - 0 work items:** Empty column.
- Done - 1 work items:** One task card: "Deployment in [redacted] Sprint 3" (ID 8758).

Work Item 8585:

- To Do - 0 work items:** Empty column.
- In Progress - 3 work items:**
 - Task 8586: "Projekt - Selektionsseite" (8 hours).
 - Task 8587: "Projekt Detailsite" (13 hours).
 - Task 8612: "Projekt - Löschen" (1 hour).
- Done - 18 work items:**
 - Task 8818: "FinalGrid: Enthält-Suche bei Textfeldern immer vor..." (8 hours).
 - Task 8694: "Testplan [redacted] durchführen" (8 hours).
 - Task 8760: "Selektion - Performance Bottlenecks analysieren" (8 hours).
 - Task 8666: "Selektionsseite: Code Review durchführen" (8 hours).
 - Task 8840: "Validierungsverhalten der Felder Startdatum und En..." (8 hours).
 - Task 8839: "Kennzeichnung der Felder Startdatum und Enddatum a..." (8 hours).
 - Task 8838: "Speichern der Datumsangaben im Projektzeitraum ist..." (8 hours).
 - Task 8805: "[redacted] - Detail - Feldlänge validieren" (8 hours).
 - Task 8783: "[redacted] Prüfung auf von Datum bei bestehenden Da..." (8 hours).
 - Task 8782: "[redacted] Datumsfelder bei Fehler markieren" (8 hours).



Sabotageziel:

Scrum-Resistenz
Scrum-Tools

Sabotiert!





- Puuuh, fast geschafft



- **Scrum = Gesunder Menschenverstand**
- **Ausbremsen von Scrum = Gesunden Menschenverstand ausschalten**
- **Fettnäpfe lauern überall**
 - Scrum-Bestandteile
 - Rollen / Artefakte / Timeboxes / Regeln / Tools
 - Bei Einführung / im laufenden Betrieb
 - Intern / beim Kunden
- **Viele Ansatzpunkte zum Ausbremsen vorhanden**
- **Wir müssen sie nur nutzen**

Wenn es mal soweit ist, wird Sabotage schwer





- **Scrum ist vielleicht gar nicht soo schlecht**
- **Aber...**
 - ...wenn wir nur beharrlich sind, sitzen wir dieses Agile Scrum Anarchie auch noch aus
- **Das bleibt unter uns, oder?**
- **Vielen Dank und...**



Stoppt Scrum!

Ihre Fettnäpfe?



Udo Wiegärtner

udo.wiegaertner@complement.de

Tel: (09 11) 2550976-14

Mobile: (01 76) 12550114

www.complement.de

