

SCRUM bei SIX Card Solutions

Bestandsaufnahme, Rückblick und Zukunft eines Scrum Projekts

Christoph Loher (Christoph.Loher@six-group.com)

Stefan Kinigadner (Stefan.Kinigadner@bsgroup.ch)

7. April 2010



Agenda

SIX Card Solutions, BSgroup Technology Innovation AG

Wo stehen wir heute?

Welche Erfahrungen haben wir gemacht?

Welches sind die kommenden Herausforderungen?

SIX Card Solutions AG

- Lösungen für kartenbasierten elektronischen Zahlungsverkehr
 - Händler
 - Kartenverarbeiter
 - Kartenherausgeber
- Über 1 Mia Kartentransaktionen pro Jahr
- Über 800 Mitarbeiter in 5 Ländern
- Dienstleistungen in 27 Ländern

Business Solution Group TI AG

Geschäftsbereiche

- § **Technology Innovation (TI)**
IT Consulting, Integration,
Individualsoftware, Outsourcing
- § Client Services (CS)
Banken- und
Versicherungsprozesse
- § Core Solutions (CS)
Core- und Back-End-Systeme
- § Advanced ERP (AE)
SAP FS, BI, PI
- § Application Services (AS)
Application Management
- § Infrastructure Services (IF)
OS, SM, IT Operations Outsourcing

Kerndaten

- § Mitarbeiter 2009: 250
- § Umsatz 2009: 49 Mio. CHF
- § Kunden: Führende Schweizer
Unternehmen und Organisationen
der **Finanz-** und **Versicherungs-**
branche, Life Science, Energie
und **Public**
- § Aktionariat: Partner und
Mitarbeiter
- § Innovation: Technologie- und
Universitätspartnerschaften

Zusammenarbeitsmodelle

- § Gesamtprojektverantwortung
(GU)
- § Projektverantwortung
(Werkvertrag)
- § Beratungsmandate
- § Expert Consulting

Business Solution Group – We create value.

SIX CS + BSG TI AG

- Zusammenarbeit seit 2004
- Aquiring (PASS)
- Issuing (iGate)
- Methodisch
- Software Engineering

Aktuell stellt BSG TI AG sechs Experten

Warum SCRUM?

Wasserfall-Vorgehen

- Release überladen
- Qualitätsprobleme

seit längerer Zeit SCRUM-Erfahrung in Entwicklung
QA und Business Engineering kaum involviert

Das Projekt iGate

- Geld- und Buchungsfluss steuern und überwachen
- Teil einer Programm-Organisation
- Batchapplikation im Backend
- Verarbeitung von 4 Millionen Transaktionen täglich
- File-basierte Verarbeitung

iGate startete im Juni/Juli 2009 mit SCRUM

- Product Backlog wird aus „konventionellem“ Grob-/Detailkonzept abgeleitet

Grobkonzeption



•

Wo stehen wir heute

Applikation

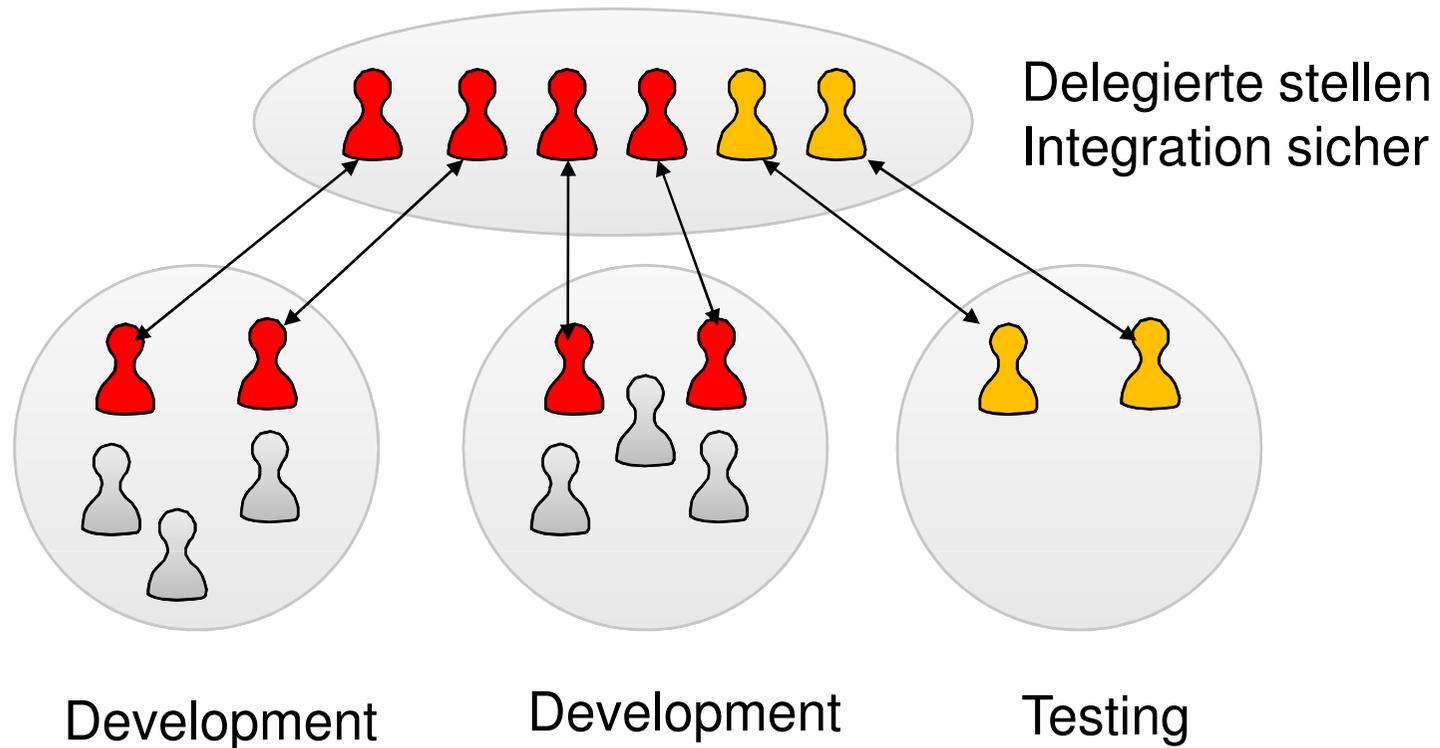
- 50 % der Applikation implementiert
 - 60 % Batch/Backend
 - 10 % Frontend
- erster Release nach dem nächsten Sprint
- 75%+ Unit-Testabdeckung
- permanenter Regressionstest (24h)

Wo stehen wir heute

Organisatorisch

- 3 Teams
 - spezifische Rollen
 - zwei Development Teams, ein QA Team
 - 1 Product Owner (Proxy) mit Team
 - 1 Scrum Master
-
- alle Scrum Disziplinen implementiert
 - 3 Wochen Sprints
 - Scrum Tools: Pinwand + Excel

Seit März 2010 arbeiten wir in drei Teams



Scrum of Scrum der Rollenverantwortlichen

- Trifft sich wöchentlich, bei Bedarf auch häufiger
- Fixe und wechselnde Personen
- Abstimmung der Teams untereinander

Höhere Velocity dank Teamsplit?

- Teamsplit wurde von Aussen verordnet
- Velocity
 - vorher 36 Storypoints bei 140 Manntagen
 - nachher 47 Storypoints bei 90 Manntagen
- ähnliche Erfahrung in Weihnachtszeit
- Done vs. Done Done
 - Commitment Entwicklung vs QA

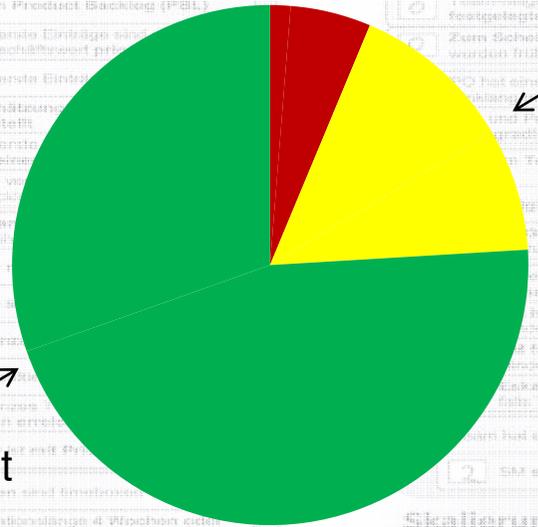
Drei Teams aus Sicht des Scrum Masters

(Die Quintessenz)
Ist dem entspricht, kann der Rest der Checkliste ignoriert werden.

1	Auslieferung funktionierender, getesteter Software alle 4 Wochen (oder kürzer)
2	Auslieferung des am meisten beschriebenen Geschäftswertes
2	Prozess wird kontinuierlich verbessert
1	Klar definierter Product Owner (PO)
2	PO ist erreichbar, zu geschäftlichen Zeiten
4	PO hat das Wissen, zu geschäftlichen Zeiten
2	PO hat direkten Kontakt mit dem Team
1	PO hat direkten Kontakt mit Stakeholdern
2	PO spricht mit einem Stakeholder (falls PO ein Team ist)
2	Team hat einen Sprint Backlog
2	Backlogpriorität sichtbar
2	Täglich aktualisiert
2	Gibt es ausschließlich dem Team
2	Daily Scrums wird durchgeführt
2	Ganzes Team nimmt teil
2	Probleme & Hindernisse werden durch den PO beseitigt
2	Demos wird nach jedem Sprint durchgeführt
2	Lauffähige, getestete Software wird geliefert
2	Feedback von Stakeholdern & PO wird aufgenommen
2	Definition of Done (DoD) vorhanden
2	DoD konsistent in jeder Iteration
1	Team berücksichtigt DoD

Kernelemente von Scrum
(Sind die Scrum-Elemente, ohne diese sollte es nicht Scrum genannt werden.)

2	Retrospektive wird nach jedem Sprint durchgeführt
4+	Prozess ist in kontinuierlicher Verbesserungsvorschlägen
1	Einige Vorschläge wurden tatsächlich umgesetzt
0	Ganzes Team + PO nehmen teil
2	PO hat ein Product Backlog (PBL)
2	Obere Einträge sind Geschäftsbedarf priorisiert
2	Obere Einträge
2	Schätzungen erstellt
2	Obere an einem
2	PO von Backlog
2	Sprint Maximal 4 Wochen
2	PO hat einen Sprint Backlog
2	Täglich aktualisiert
2	Gibt es ausschließlich dem Team
2	Daily Scrums wird durchgeführt
2	Ganzes Team nimmt teil
2	Probleme & Hindernisse werden durch den PO beseitigt
2	Demos wird nach jedem Sprint durchgeführt
2	Lauffähige, getestete Software wird geliefert
2	Feedback von Stakeholdern & PO wird aufgenommen
2	Definition of Done (DoD) vorhanden
2	DoD konsistent in jeder Iteration
1	Team berücksichtigt DoD



Scrum Checkliste

6% - nicht umgesetzt

18% - umgesetzt, aber...

76% - umgesetzt

Empfehlungen sind nicht immer notwendig
Das meiste hiervon wird normalerweise benötigt, aber nicht immer alles. Ausnahmen!

2	Team hat alle Fähigkeiten, um Backlog Einträge "Done" zu bekommen
0	Team-Mitglieder haben keine festgelegten Rollen
2	Zum Sprintbeginn vorzeitige Iterationen werden frühzeitig im PBL
2	PO hat einen Sprint Backlog
2	Täglich aktualisiert
2	Gibt es ausschließlich dem Team
2	Daily Scrums wird durchgeführt
2	Ganzes Team nimmt teil
2	Probleme & Hindernisse werden durch den PO beseitigt
2	Demos wird nach jedem Sprint durchgeführt
2	Lauffähige, getestete Software wird geliefert
2	Feedback von Stakeholdern & PO wird aufgenommen
2	Definition of Done (DoD) vorhanden
2	DoD konsistent in jeder Iteration
1	Team berücksichtigt DoD

Scaling
Für größere Teams ist Scrum eine Herausforderung.
Chief Product Owner vorhanden (falls viele POs)
Abhängige Teams führen Scrum of Scrums durch
Abhängige Teams integrieren in jeden Sprint

Positive Hinweise
Wesentliche Hinweise über guten Scrum-Erfolgsfaktor:
Sprint hat ein hohes Energie-Level
Überstunden sind selten und werden bezahlt
Diskussion, Kritik und Experimentieren mit dem Prozess

PO = Product Owner SM = Scrum Master PBL = Product Backlog DoD = Definition of Done
http://www.dziwi.com/scrumchecklist | Version 2.1 (2008-08-17)

Drei Teams aus Sicht des Scrum Masters

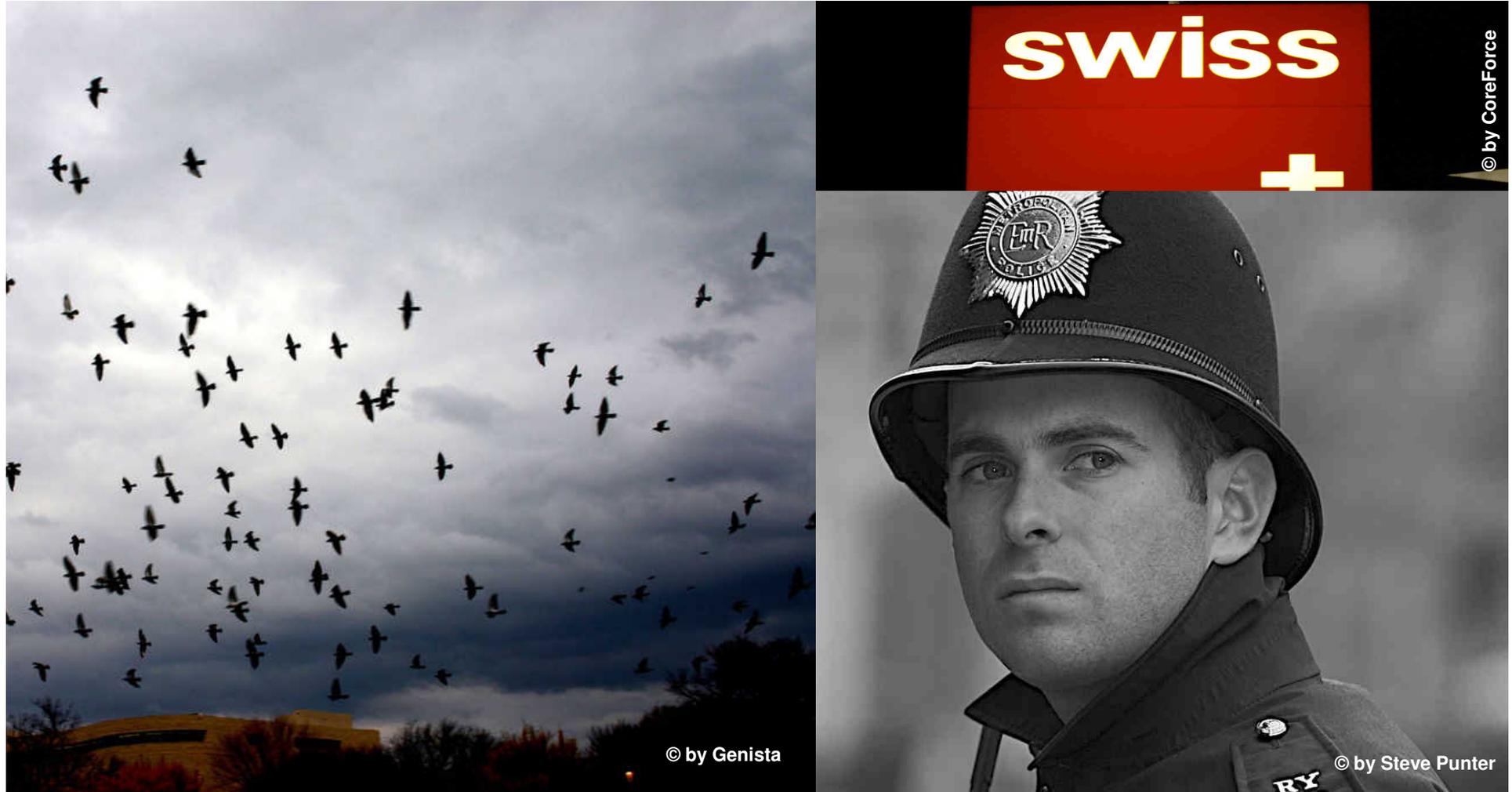
- Zu Beginn des Projektes 6 Personen im Team
 - 1 Person mit 100% Verfügbarkeit fürs Team
 - 5 Personen mit diversen Altlasten
- Ursprüngliche Projektplanung definierte ein Wachstum des Teams um 5 weitere „Development Ressourcen“
- peu à peu wuchs die Mitgliederzahl des Teams
- Sprints funktionieren – aber keine High-Performance
- Ende Dezember – grosser Sprint mit gutem Abschluss
Weniger Teammitglieder bekommen mehr fertig!

Drei Teams aus Sicht des Scrum Masters

- PO im März: verordnet Split
- Team:
Q: „Wollt Ihr den Split?“
A: Nein bzw. Unsicherheit

Warum ist das so?

Drei Teams aus Sicht des Scrum Masters



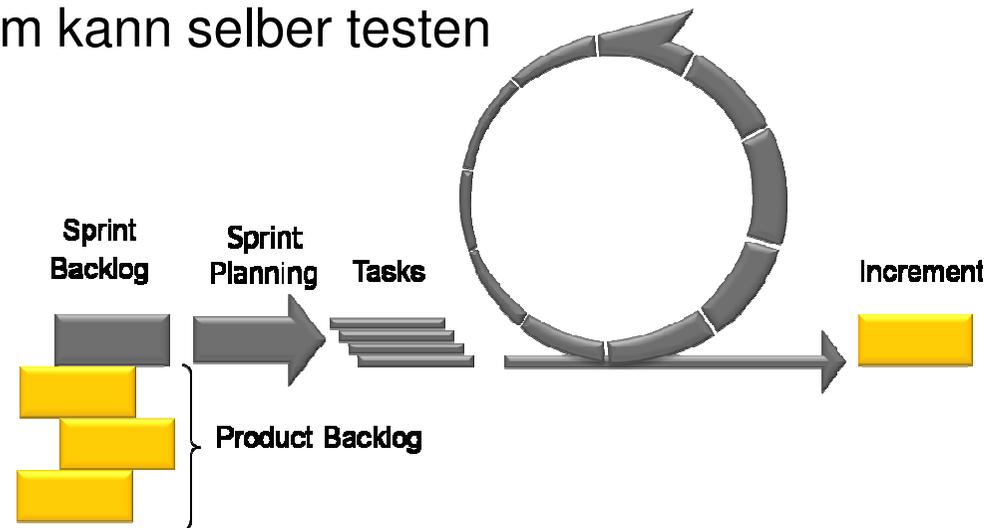
QA Team als separates Team

Ressourcenengpass QA

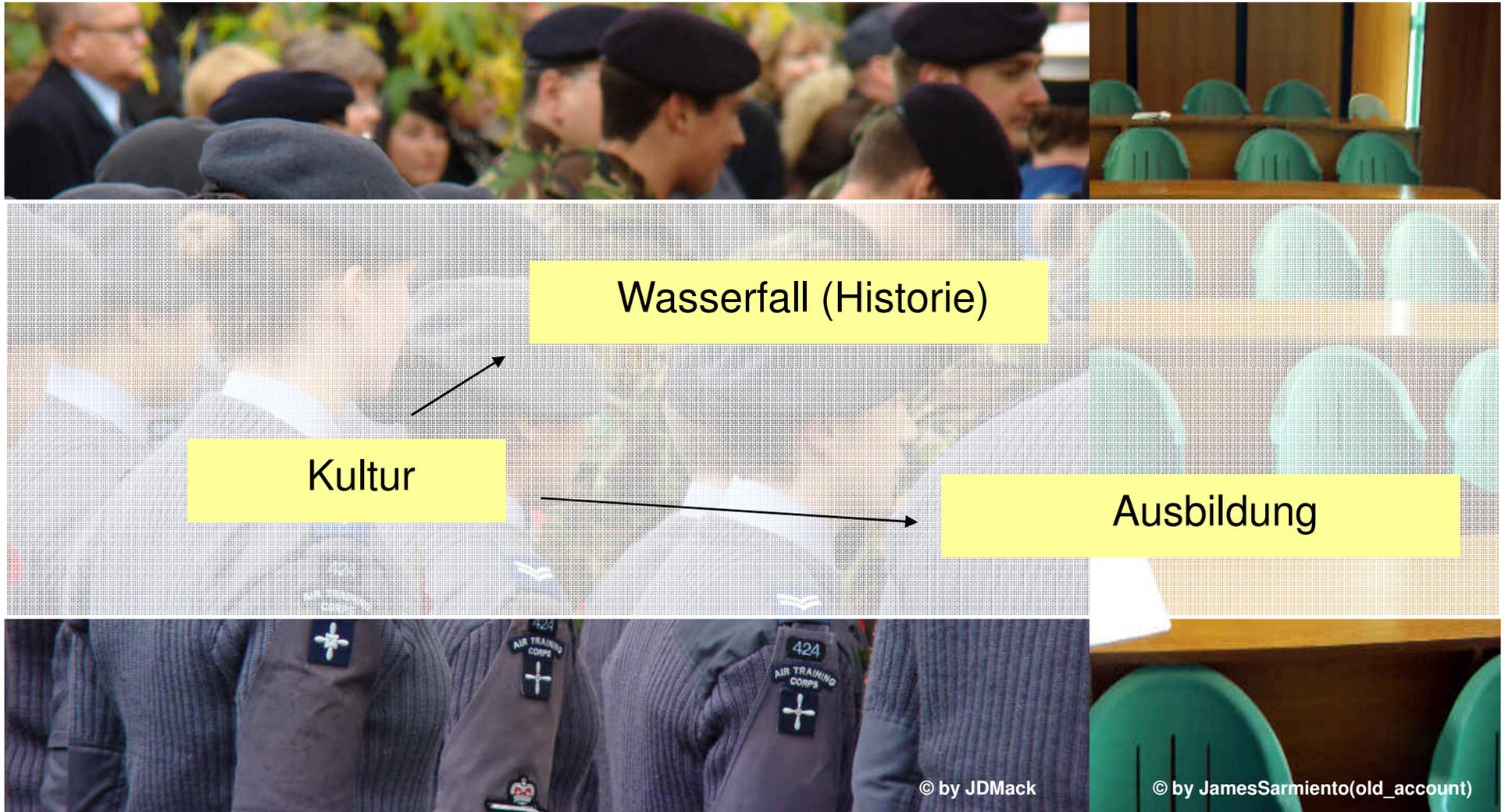
- fehlendes QA-Know-how blockiert beide Teams

Idee: Testfälle vor Sprintbeginn

- Vollständigkeit sicherstellen
- Entwicklungsteam kann selber testen



QA Team als separates Team



Detailkonzept als PBL

3	Einführung	
3.1	Projektziele	
3.3	Abhängigkeiten zu weiteren Projekten / Dienstleistungen.....	
4	Grobdesign	
4.1	Technische Grundlagen	
4.2	Systemtopologie.....	
4.3	Funktionsübersicht.....	
5	Systemanforderungen	
5.1	System A	
5.2	System B	
5.10	Weitere Systeme / Applikationen / Tools.....	
5.11	Externer Zugriff.....	
6	Qualitätsanforderungen	
6.1	Benutzerfreundlichkeit	
7	Sicherheitsanforderungen	
8	Anforderungen an die QA	
8.1	Zertifizierung	
9	Betrieblichen Anforderungen	
10	Anforderungen an die operationellen Prozesse	
11	Migrationskonzept	

User Story entspricht nicht einem Use Case

Mapping zwischen Anforderungen und User Story

- Vollständigkeit
- relevante Anforderungen
- effiziente Verweise auf Detailkonzept

auch Detailkonzept entwickelt sich weiter

- nachführen der User Stories
- Konsistenz

Umfang Detailkonzept

sinnvolle User-Story-Grösse

Wo stehen wir heute - Zusammenfassung

- QA und Business nicht voll im Team
- Teamsplitt gegen Willen verordnet, kommt gut an
- Anforderungen sind „aus der alten Welt“
- Unternehmenskultur (Wasserfall) vs. Agile Ansätze

- Und trotzdem einen guten Nokia-Test?

Ja, wir machen Scrum But(t)...



© by Hitchster

Herausforderungen für die Zukunft

- User Stories
 - Right-Sizing und Abhängigkeiten
 - verständlich (für Team und Business)
- Umgang mit Rollenkonflikte finden
- Doppel- und Trippelrollen
- Beweis dass wir Qualität liefern
- Dokumentation
 - Endanwender
 - Support (Pikett, 3rd Level)

Herausforderungen für die Zukunft

- Engineering Practices im Team (DEV- & QA-Engineers)
 - TDD, Iterative vs. Componentbased
- Täler und Gipfel überwinden
- SCRUM in Wasserfallstrukturen „einebnen“
- Integration Business, Integration QA
- Impedimentmanagement
- Scrum
 - Believe
 - Courageous
 - Los lassen und Vertrauen haben

Fragen?



Vielen Dank!

Haben Sie noch Fragen? Bleiben Sie mit uns in Kontakt:

Christoph.Loher@six-group.com

Stefan.Kinigadner@bsgroup.ch

