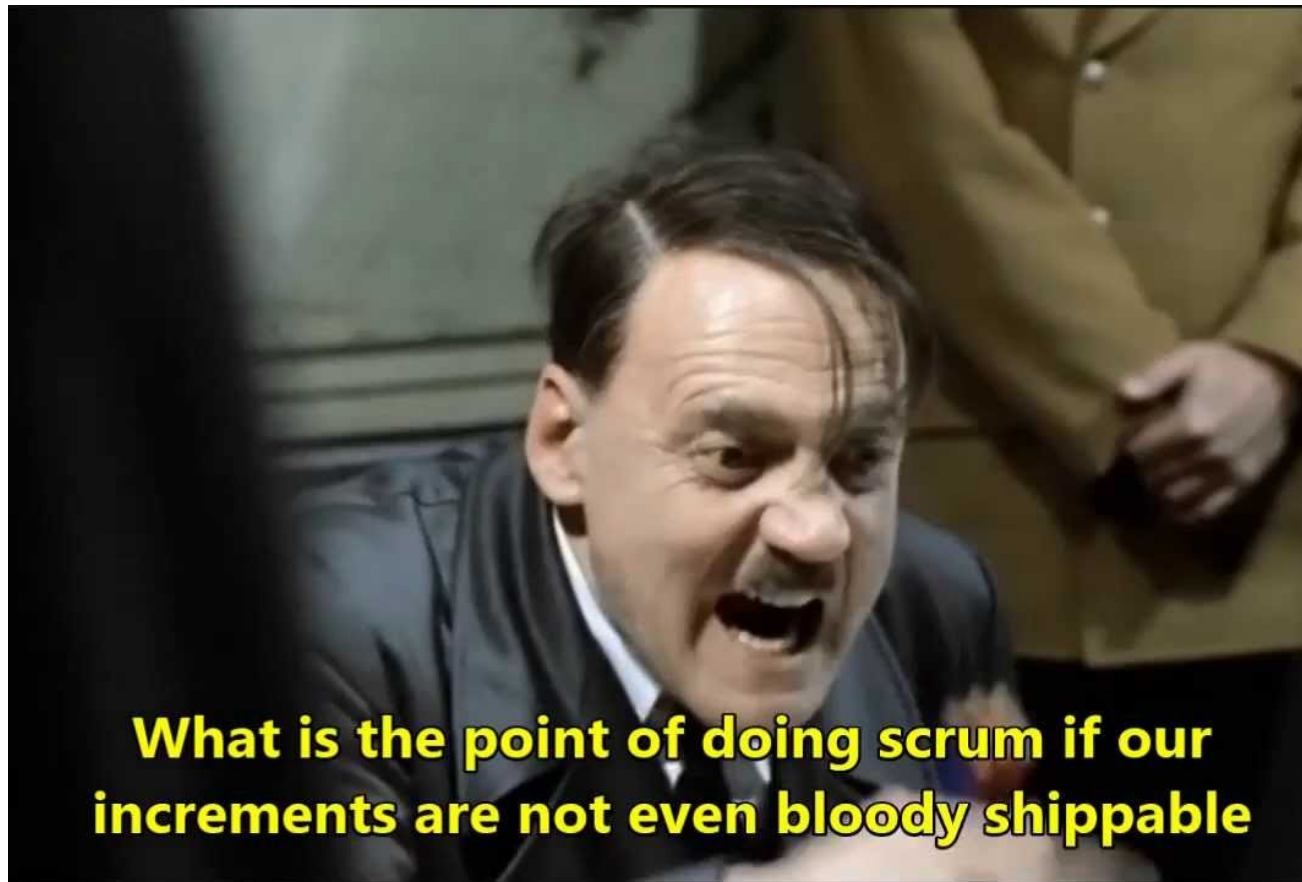


Modul 426

Agile Softwareentwicklung



Agile Entwicklerteams (Kap.14)

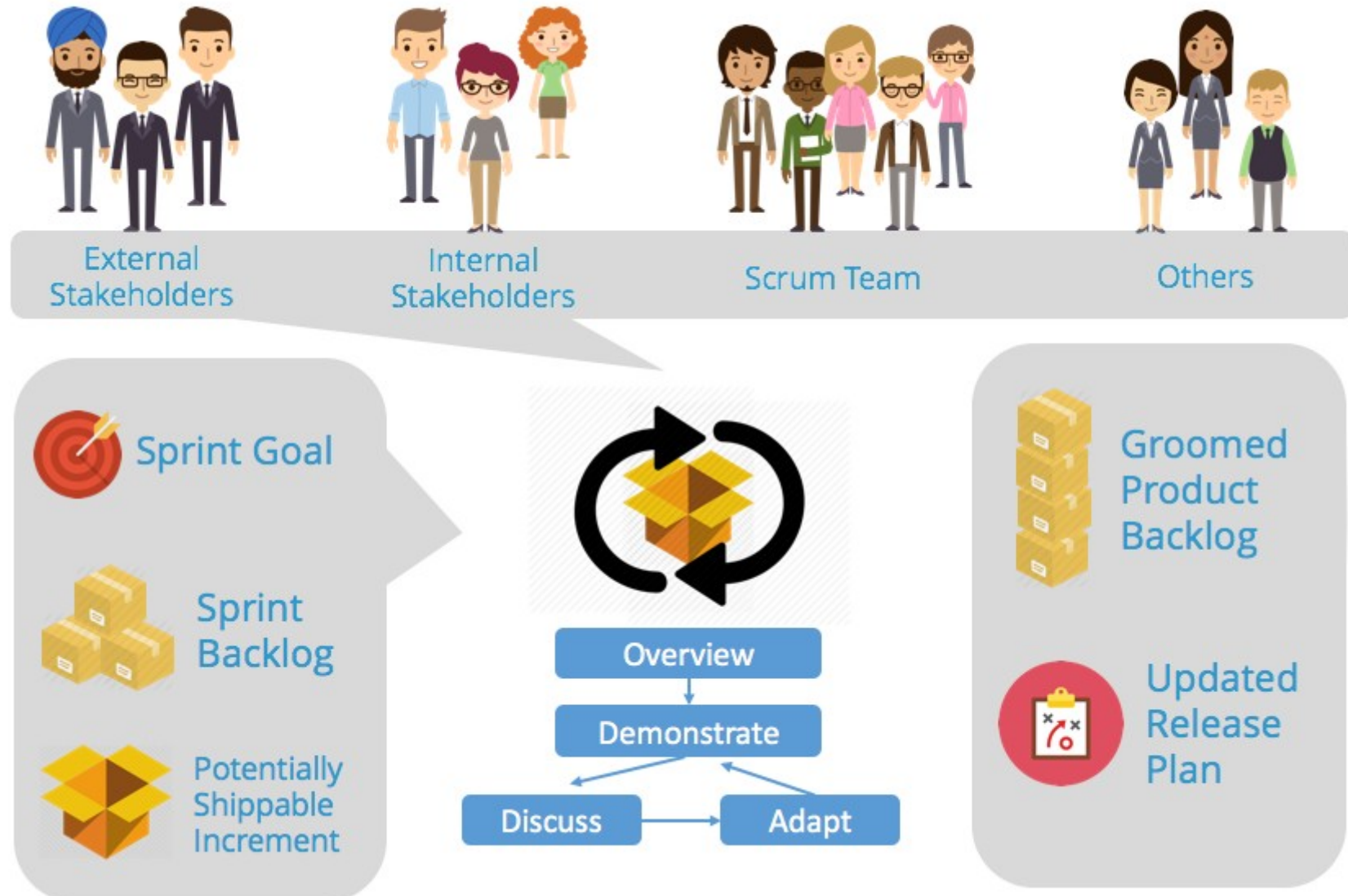
- kollektive Verantwortung
- selbstorganisiert
- Team > Summe(mitglieder)
 - Heldentum, Wissensmonopole vermeiden
 - sharing is caring: Spezialist-Generalist
- unterstützt PO bei Erstellung der User Stories

Broken Window Theorie & Agile

- Verwahrlosung leeres Hauses beginnt mit einem zerbrochenen Fenster
- bald
 - mehr zerbrochene Fenster
 - Unrat
 - Verwahrlosung
- gleiches Prinzip (nicht nur) beim Teamgeist und -prinzipien
- Wehret den Anfängen:
 - offene Bugs
 - unvollständige User Stories
 - fehlende Unit-Tests
 - ausgefallene Daily Scrums



Sprint Review (Kap. 3.8.3)



Sprint Review (Kap. 3.8.3)

- Wer: PO, Stakeholders, Team und ScrumMaster
- Was: Team präsentiert **fertige** User Stories
- Wozu: PO akzeptiert Lösungen (oder auch nicht)
- Was es nicht ist:
 - Motivationsmeeting
 - Folienschlacht ohne Demo
 - Planungsmeeting
 - individuelle Leistungsbeurteilung
 - ...

Retrospektive (Kap. 3.8.4)

- Wofür haben wir mehr/weniger Aufwand gebraucht als veranschlagt und warum?
- Welche Techniken verbessern, neu einbeziehen?
- Was hat unsere interne Zusammenarbeit behindert?
- Was war besonders gut für unsere Zusammenarbeit?
- Wie verstärken wir diese internen positiven Elemente?
- Welche Störungen gab es von außen?

Retrospektive

- Scrum Master
 - moderiert, leitet an
- Don'ts:
 - gegenseitige Schuldzuweisungen
 - Kritik an einzelnen
 - Meeting zur Leistungsbeurteilung (kills team spirit)



Sinn: zukünftig besser (zusammen)arbeiten,
Prozessverbesserungen

Prüfungsvorbereitung

- Kapitel lesen
 - Manifest & Principles
 - 3 (Scrum, Rollen, Zuständigkeiten, Ablauf, Release-Planung)
 - 7 (CM)
 - 8 (CI)
 - 10 (Refactoring)
 - 13 (Pair Programming)
 - 15.2, 15.4
- Moduljournal inhaltlich wertvoll und bezogen auf neue Richtlinien fertigstellen