

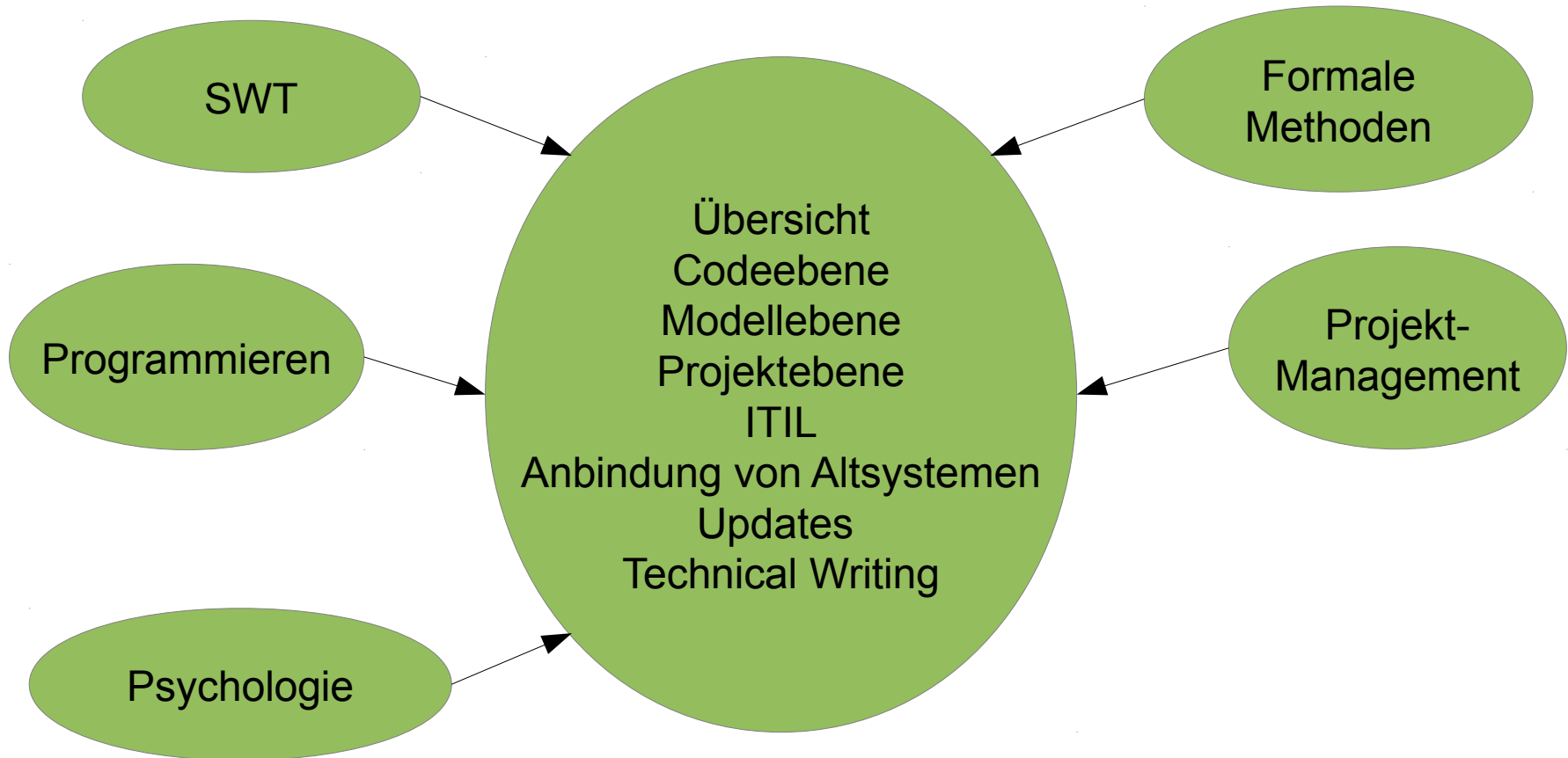


# Software- Engineering für langlebige Systeme

## VL11 WarpUp

- Nochmal zurückschauen und Verbindungen sehen
- Und nun...

## Was haben wir gesehen:



# Softwareerosion

Quellen ...

## Änderungen in der Systemumgebung

- Prozessoränderungen
  - Umstellung 8 → 16 → 32 → 64 Bit-Systeme (s. Übung)
  - Prozessorgeschwindigkeit (TurboPascal- Zählerüberlauf)
- Änderungen im Betriebssystem
  - Geänderte Zugriffsrechtssysteme
- Neue/Geänderte Schnittstellen

## Verlust von Wissen

- Designentscheidungen können nicht mehr nachvollzogen werden
  - Wichtige Gründe werden bei Umbauten nicht bedacht
- Bedeutung von Magic Numbers geht verloren
- Funktionen werden außerhalb der vorgesehenen und geprüften Bereiche genutzt

## Unterschiede in der Programmiererfahrung

- Code Anderer wird fehlinterpretiert
- „Verschlimmbessern“
- Unangemessene Verallgemeinerung/Vereinfachung

## Fehler in der Arbeitsorganisation

- Programmierer wissen es besser als Designer
  - Design und Code passen nicht zusammen
  - Testcases passen nicht zur Code-Basis
- Schlecht designter Prototyp wird Produkt
- Häufiger Wechsel der Programmierer/ Designer
- Updatefähigkeit wird später ergänzt
  - Teilweise werden Veränderungen „zurückgebaut“
- Keine „deprecated“-Verwaltung
  - Viel alter Code bleibt im System
- Aufgabenteilung nicht gelebt



## Schlechter Code

- Code nicht verständlich
- Wenn eine Stelle geändert wird, müssen viele (weit entfernte) Code-Stellen mit angepasst werden.
- Wenn der Code geändert wird, ergeben sich viele Folgefehler

Achtung:

- Schlechter Code ist Auswirkung und Quelle zugleich!
- Hier beginnt ein Teufelskreis

## Schlechtes Design

- Das Design ist zu komplex/zu einfach
- Es existieren viele (unübersichtliche) Abhängigkeiten (s. Findbugs-Bsp)
- Entscheidungen sind nicht dokumentiert
- Verschiedene widersprüchliche Spezifikationen
- Design für Zielsprache nicht angemessen/passend

### Achtung:

- Schlechtes Design ist Auswirkung und Quelle zugleich!
- Hier beginnt ein Teufelskreis

## Inkonsistenzen zwischen Artefakten

- Code
- Datenbank
- Design/Modell
- Dokumentation
- Tests

Achtung:

- Inkonsistenzen zwischen Artefakten sind Auswirkung und Quelle zugleich!
- Hier beginnt ein Teufelskreis

# Abschließende Bemerkungen

## Mein Rat für weiteres Lernen

- Englisch
  - Alle wichtige Literatur kommt im Deutschen erst mit erheblicher Verspätung
  - Dokumentation
- Viel Programmieren in verschiedenen Sprachen
  - Es gibt nicht eine gute Sprache
  - Verschiedene Paradigmen
- Sozial- und Psychologie-Grundlagen
  - Verstehen der Effekte in einer Gruppe
- Interkulturelles Wissen
- Projekt-Management

## Six Sigma

- Managementsystem zur Prozessverbesserung, statistisches Qualitätsziel und zugleich eine Methode des Qualitätsmanagement
- Der Six-Sigma-Kernprozess: DMAIC
  - Define (D)
  - Measure (M)
  - Analyze (A)
  - Improve (I) (bzw. Engineer (E) bei neuen Prozessen)
  - Control (C)

6σ

## Fehler sind schlecht

| Sigma level | DPMO    | fehlerhaft  | fehlerfrei % | Stromausfall |
|-------------|---------|-------------|--------------|--------------|
| 1           | 691.462 | 69 %        | 31 %         | 246d         |
| 2           | 308.538 | 31 %        | 69 %         | 110d         |
| 3           | 66.807  | 6,7 %       | 93,3 %       | 24d          |
| 4           | 6.210   | 0,62 %      | 99,38 %      | 53h          |
| 5           | 233     | 0,023 %     | 99,977 %     | 2h           |
| 6           | 3,4     | 0,00034 %   | 99,99966 %   | 1,7Min       |
| 7           | 0,019   | 0,0000019 % | 99,9999981 % | 0.8s         |

## Lean

- „integriertes soziotechnisches System, dessen Kernzielsetzung
- die Beseitigung von Verschwendung ist,
- indem gleichzeitig
  - lieferantenseitige,
  - kundenseitige und
  - interne Schwankungen reduziert oder minimiert werden“

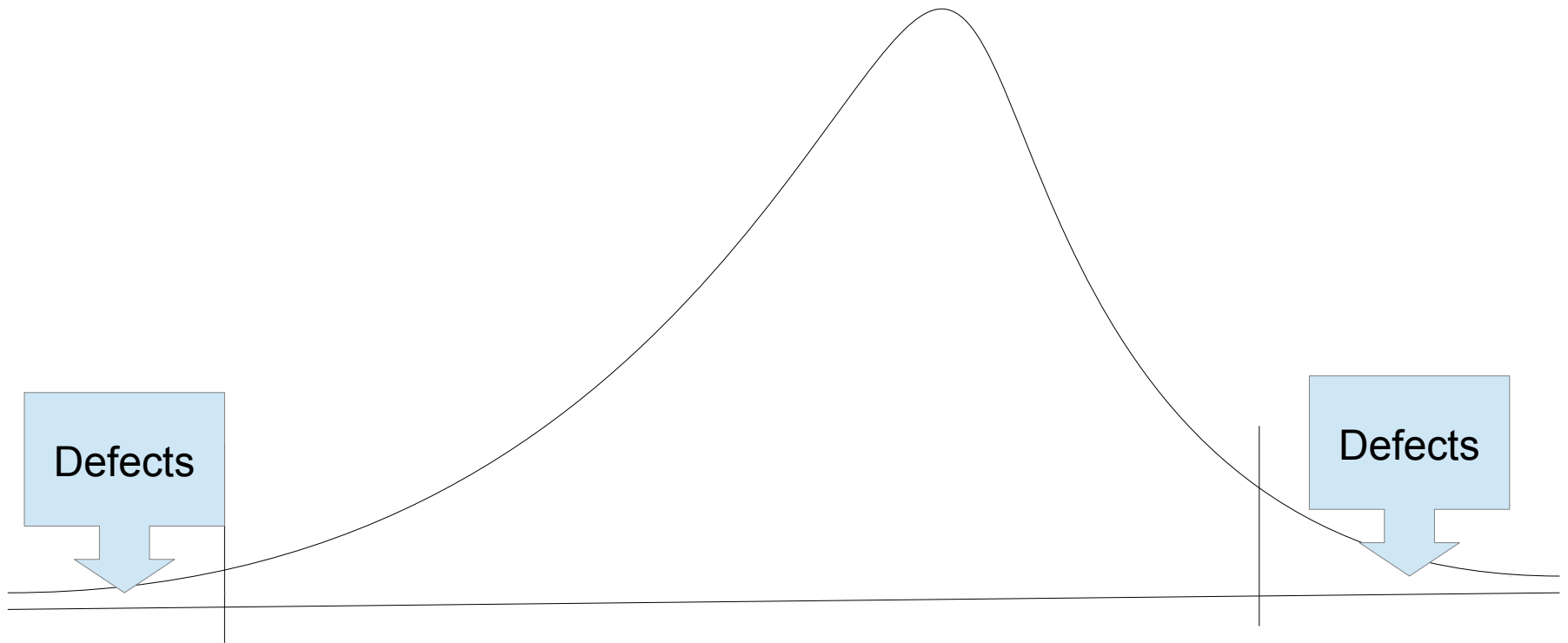
(Shah, R.; Ward, P. T.)



## Waste

- Überproduktion
- Bestände
- Transport
- Wartezeit
- Aufwändige Prozesse
- Lange Wege
- Fehler
- Ungenutztes Potenzial

## Defects



## Live Long Learning – Es geht weiter

- Informatikwissen halbiert sich in 3 Jahren!
- Die Arbeitsagentur geht nach 10 Jahren arbeiten in einem anderen Bereich davon aus, das man im ursprünglichen Bereich ungelernt ist
  
- Lerne immer weiter
- Aber verfalle nicht in Hypes!

# The End

