

- Einführung
- Kommunikation
- Software Qualität
- Vorgehensmodelle
- Requirements Engineering
- Software Architektur und – Design
- Konfiguration Management

Einführung

- Motivation
- Themen des Software Engineerings
- Themen dieser Veranstaltung

- Motivation
- Themen des Software Engineerings
- Themen dieser Veranstaltung

Ein paar Zahlen :

(Quelle <http://de.statista.com/statistik/faktenbuch/128/a/branche-industrie-markt/it-telekommunikation/softwareindustrie/>):

- In Deutschland gab es 2011 ca. 75.000 Unternehmen in der Softwareindustrie.
- Branchenführer in Deutschland ist SAP. 2009 machte das Unternehmen etwa 1,4 Milliarden Euro Umsatz.
- Der Gesamtumsatz von Microsoft lag im Jahr 2013 bei rund 77,85 Milliarden US-Dollar.

Vgl: BIP von Slowenien 2011: 49,588 Milliarden USD

Zehn-Jahresvergleich:

Der Beschäftigungsanteil der Branche ist im jährlichen Durchschnitt um knapp 4% gestiegen. Somit gehört die Branche im letzten Jahrzehnt in Bezug auf Beschäftigungswachstum zu den Branchen mit den deutlichsten Zugewinnen.

Quelle:

<http://www.isi.fraunhofer.de/isi-wAssets/docs/t/de/publikationen/Fraunhofer-ISI--Software-Atlas-Deutschland-2012.pdf>

Prognose

Prognose, dass der Branche bis 2030 ein Beschäftigungswachstum von 80 Prozent bevorsteht.

Quelle: <http://www.studie-life.de/life-studien/vernetztes-arbeiten/bedeutung-der-ict-fuer-wirtschaft-und-gesellschaft/stellenwert-der-ict-gesamtwirtschaftlich/>

Studie zur Bedeutung der IKT Industrie in Deutschland:

https://www.destatis.de/DE/Publikationen/Thematisch/UnternehmenHandwerk/Unternehmen/IKT_BrancheDeutschland5529104139004.pdf?__blob=publicationFile

Innovationen

Mehr als 80 % der Innovationen in der deutschen Wirtschaft
beruhen auf IK.

Quelle:

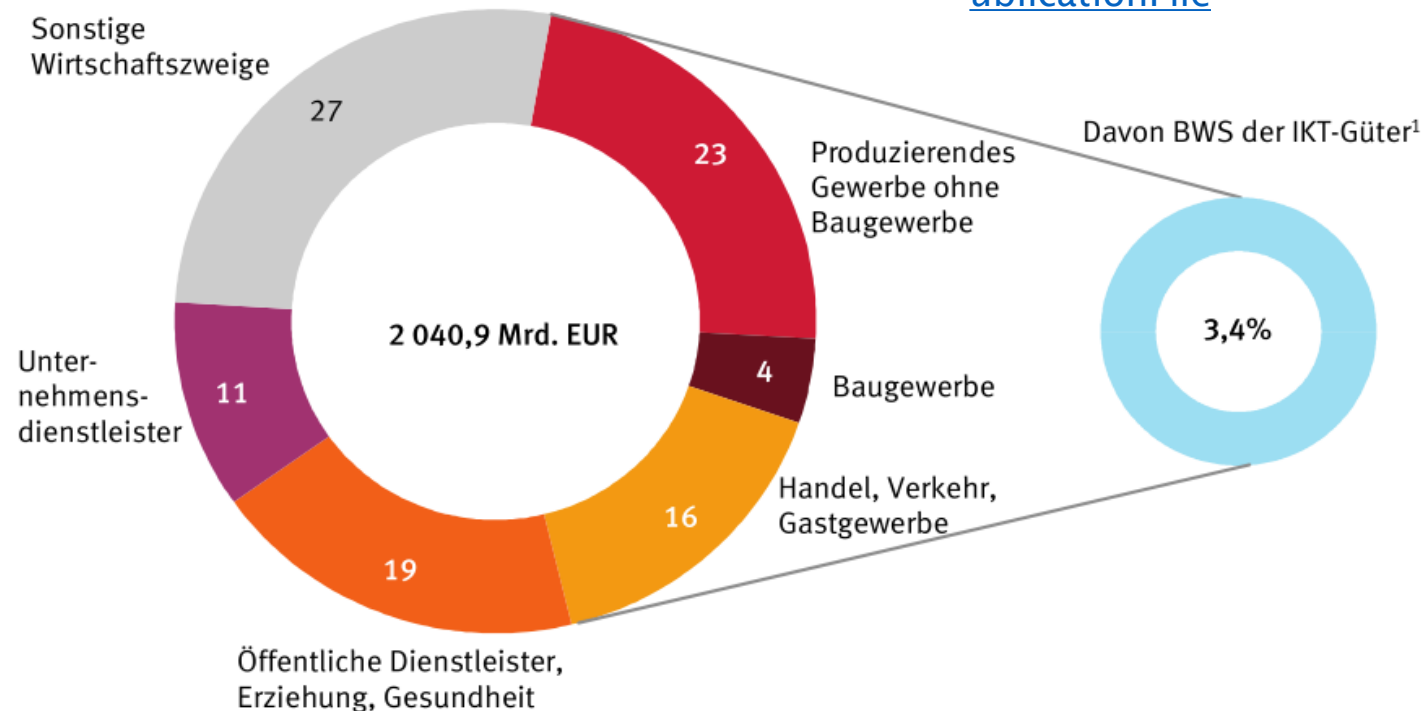
https://www.destatis.de/DE/Publikationen/Thematisch/UnternehmenHandwerk/Unternehmen/IKT_BrancheDeutschland5529104139004.pdf?__blob=publicationFile

Bruttowertschöpfung durch IKT

Schaubild 1 **Bruttowertschöpfung in Deutschland 2009**
in %

Quelle

https://www.destatis.de/DE/Publikationen/Thematisch/UnternehmenHandwerk/Unternehmen/IKT_BrancheDeutschland5529104139004.pdf?__blob=publicationFile

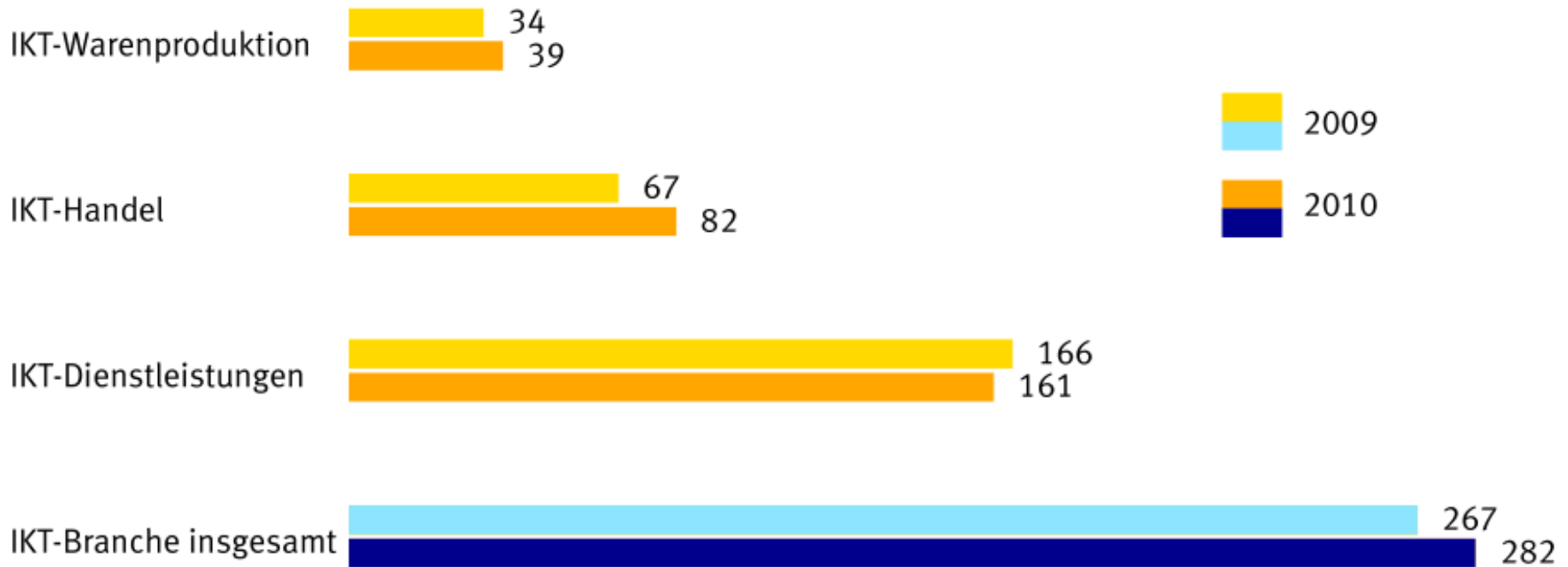


¹ Ohne Handel und Leasing von IKT-Produkten.

2013 - 08 - 0126

IKT Umsätze

Schaubild 3 Entwicklung der Umsätze in der IKT-Branche im Jahresvergleich
in Mrd. EUR



Quelle

2013 - 08 - 0128

https://www.destatis.de/DE/Publikationen/Thematisch/UnternehmenHandwerk/Unternehmen/IKT_BrancheDeutschland5529104139004.pdf?__blob=publicationFile

Umfang von Software Lösungen

- Bank of New York Mellon: 1 12 500 Cobol Programme mit 343 million lines of code
(<http://www.computerworld.com/article/2504568/data-center/the-cobol-brain-drain.html>)
- SAP's application environment has more than 250 million lines of code
(<http://hurwitz.com/blogs/marcias-blog/entry/how-does-sap-turn-250-million-lines-of-code-into-modular-services>)
- Healthcare.gov: 500 million lines of code
(http://www.nytimes.com/2013/10/21/us/insurance-site-seen-needing-weeks-to-fix.html?pagewanted=2&_r=1)

Softwarefehler:

- Handy bis zu 600 Fehler (dh. 3 Fehler pro 1000 loc)
- Windows95 bis zu 200.000 Fehler (dh. 20 Fehler pro 1000 loc)
- Space Shuttle weniger als 1 Fehler pro 10.000 loc
- Gefundene Defekte in 1000 Zeilen Quellcode: (M. Cusumano, MIT 1990)
 - 1977: 7 - 20 Defekte
 - 1994: 0,05 - 0,2 Defekte

0,1%-Defektniveau bedeutet:

- pro Jahr: 20.000 fehlerhafte Medikamente 300 versagende Herzschrittmacher
- pro Woche: 500 Fehler bei medizinischen Operationen
- pro Tag: 16.000 verlorene Briefe in der Post, 18 Flugzeugabstürze
- pro Stunde: 22.000 Schecks nicht korrekt gebucht –

Quelle: Dr.-Ing. Sandro Schulze, Software Engineering 1 - WS 2013/2014

Motivation – Fehler in Software

Folgeschwere Fehler in Software:

Siehe z.B. <http://top-10-list.org/2010/05/03/ten-costliest-software-bugs/>



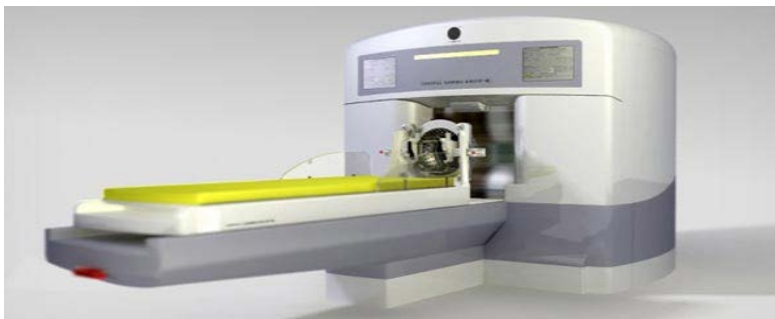
Mars Climate Orbiter

➔ 1999 verlorengegangen aufgrund eines Einheitenfehlers

➔ 125 Mio Dollar Schaden

Siehe auch

http://de.wikipedia.org/wiki/Mars_Climate_Orbiter



Therac-25 Medical Accelerator

➔ zwischen 1985 und 1987
Möglichkeit eines schweren Funktionsfehlers

➔ 5 tote Patienten

Siehe auch

<http://de.wikipedia.org/wiki/Therac-25>

Motivation – Fehler in Software



Siehe auch
<http://mentalfloss.com/article/12635/date-1983-stanislav-petrov-single-handedly-prevented-nuclear-war>



1960: RADAR System auf Grönland
Wird durch aufgehenden Mond
irritiert

→ beinahe Atomschlag gegen
Russland

Quelle: Thaller: Software-Test, Verlag Heinz Heise, 2.
Auflage

1983: Falscher Alarm in einem
Soviet Bunker

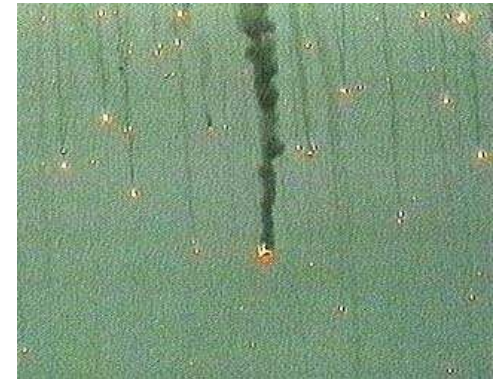
→ Beinahe dritter Weltkrieg

Geplanter Start 31. Aug 2003,
tatsächlicher Start 1. Jan 2006;
3.5 Milliarden Euro Einnahmeausfälle

Motivation: Beispiel für fehlendes/falsches Testen



- 4. Juni 1996: Absturz einer Ariane-5-Rakete aufgrund eines Softwarefehlers
- Softwareteile aus der Vorgängerbaureihe Ariane-4 wurden ohne Test in die neue Baureihe übernommen
- Wichtige Rahmenparameter waren außerhalb der zulässigen Grenzen
- Kosten des Ariane-5-Programms bis 1996: ca. 8 Milliarden US-\$
- Wert des zerstörten Satelliten: ca. 500 Millionen US-\$



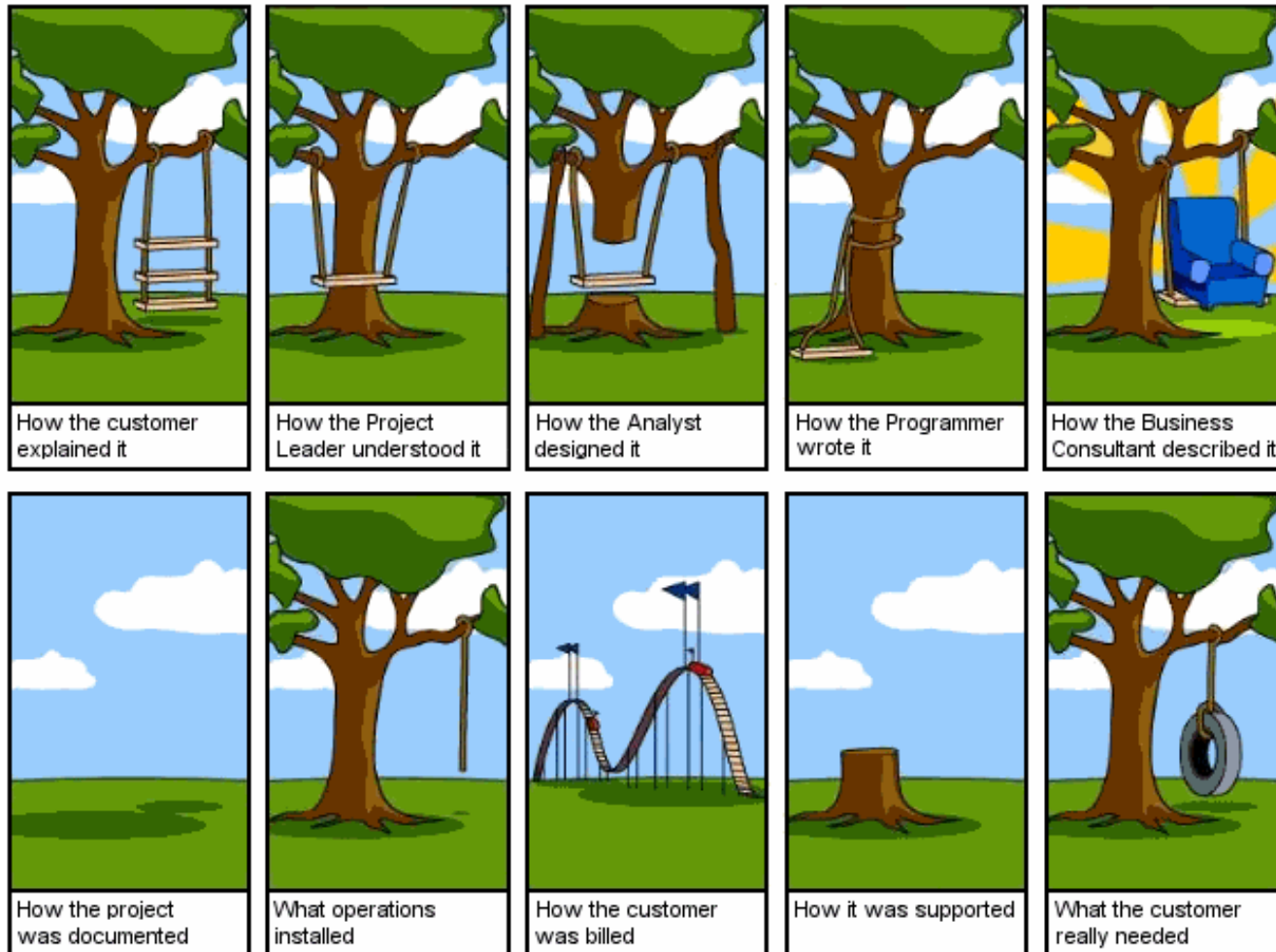
Siehe auch http://de.wikipedia.org/wiki/Ariane_5#Fehlgeschlagener_Erstflug

Softwarefehler 2013:

<http://www.computerwoche.de/a/zehn-spektakulaere-softwarefehler,2553543>

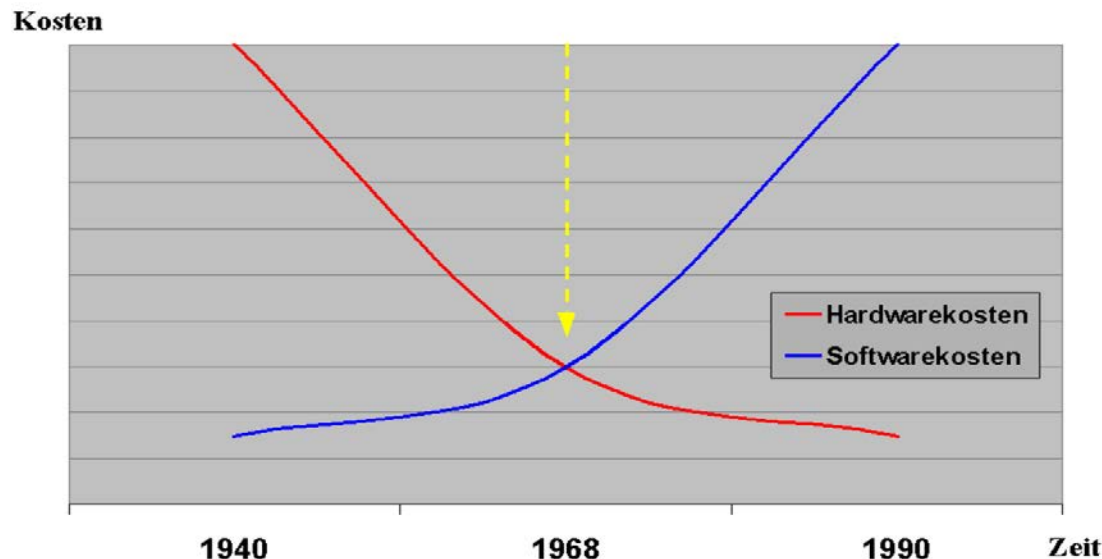
Weitere Beispiele in *G.E. Thaller: Software – Test, 2. Auflage, Heise, Hannover*

Motivation: Beispiel für Fehler in den verschiedenen Projektphasen



Motivation - Software Krise

- Programmsysteme der 60er Jahre wurden zunehmend komplexer aber es gab
 - keine geeigneten (Programmier-)Sprachen
 - keine geeigneten Methoden/Vorgehensweisen
 - keine geeigneten Werkzeuge
- Folgen (bis heute!)
 - Software-Kosten stiegen kontinuierlich (Hardware-Kosten fallen, s. unten)
 - extrem viele Softwareentwicklungsprojekte scheiterten.



Motivation - Status der Digitalisierung



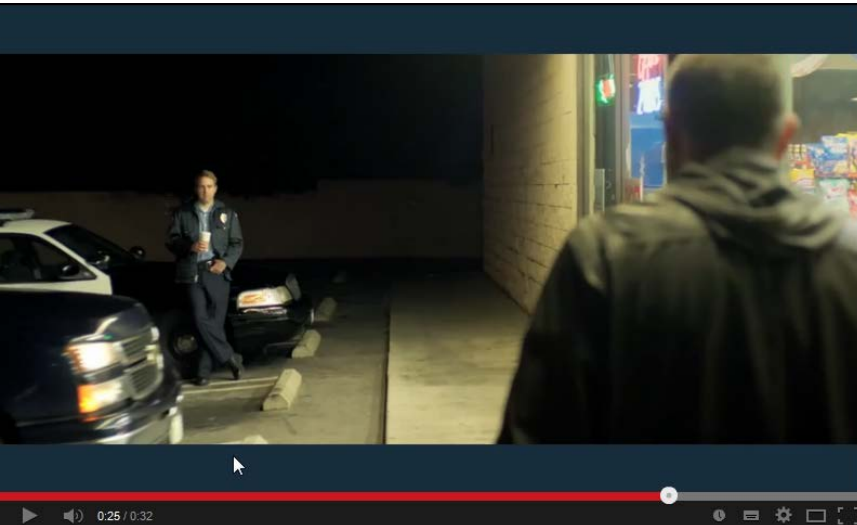
Autonome Systeme:

<http://www.youtube.com/watch?v=pMFdlmwMDRY>

Google Self Driving Car:

<https://www.youtube.com/watch?v=M6iSBiU8bv8>

<https://www.youtube.com/watch?v=6TjNJGWJcwK>



Predictive Analytics:

<https://www.youtube.com/watch?v=iY3WRvXVogo>

Predicting crime:

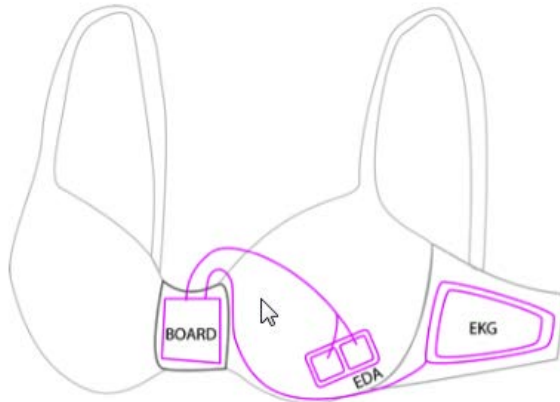
<https://youtu.be/ho10naUH5ME>

Motivation - Status der Digitalisierung



Brain reading:

<https://www.youtube.com/watch?v=76lIQtE8oDY>

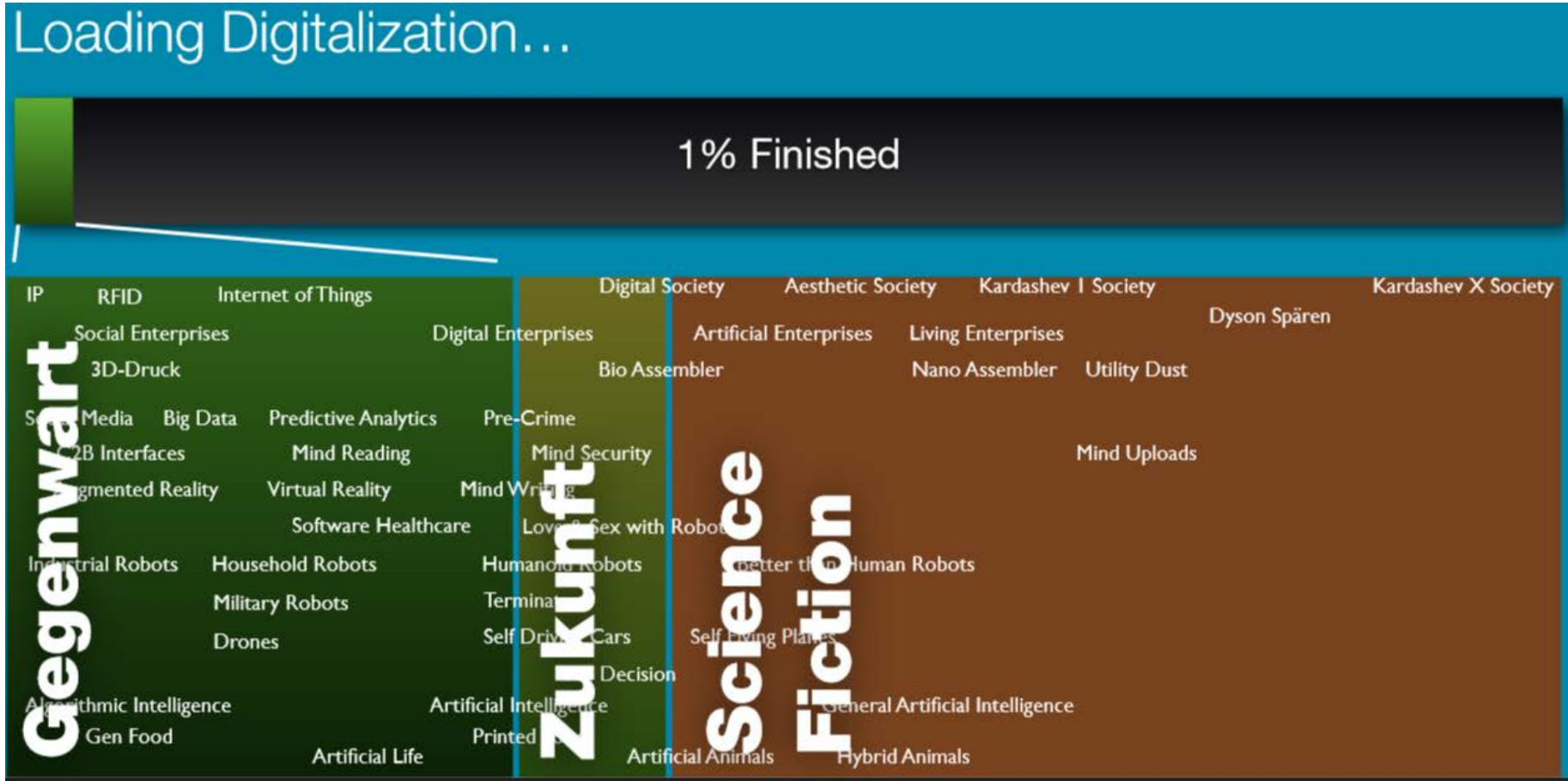


Wearables:

<http://www.cs.rochester.edu/hci/pubs/pdfs/FoodMood.pdf>

Motivation – Gegenwart und Zukunft

Quelle: Prof. Tim Bruysten am Jahrestreffen der SW Foren Leipzig am 4.3.2014



Motivation – Arten von Software

- Produkt oder Individualentwicklung
- Systemsoftware (Betriebssystem, Compiler, Editor,...) oder Anwendungssoftware
- Produktintegriert oder für reine Computersysteme
- Datenintensiv oder berechnungsintensiv
- Monolithisch oder verteilt
- Standalone oder mit anderen Anwendungen integriert
- ➔ Durch Anforderungen und Domäne charakterisiert.

- Software ist immateriell.
- Software unterliegt keinem Verschleiß.
- Software veraltet.
- Es gibt keine Software Ersatzteile: Defekte sind immer Konstruktionsfehler.
- Software ist schwer zu vermessen.
- Software gilt als leicht änderbar.
- Software unterliegt einem ständigen Anpassungsdruck.

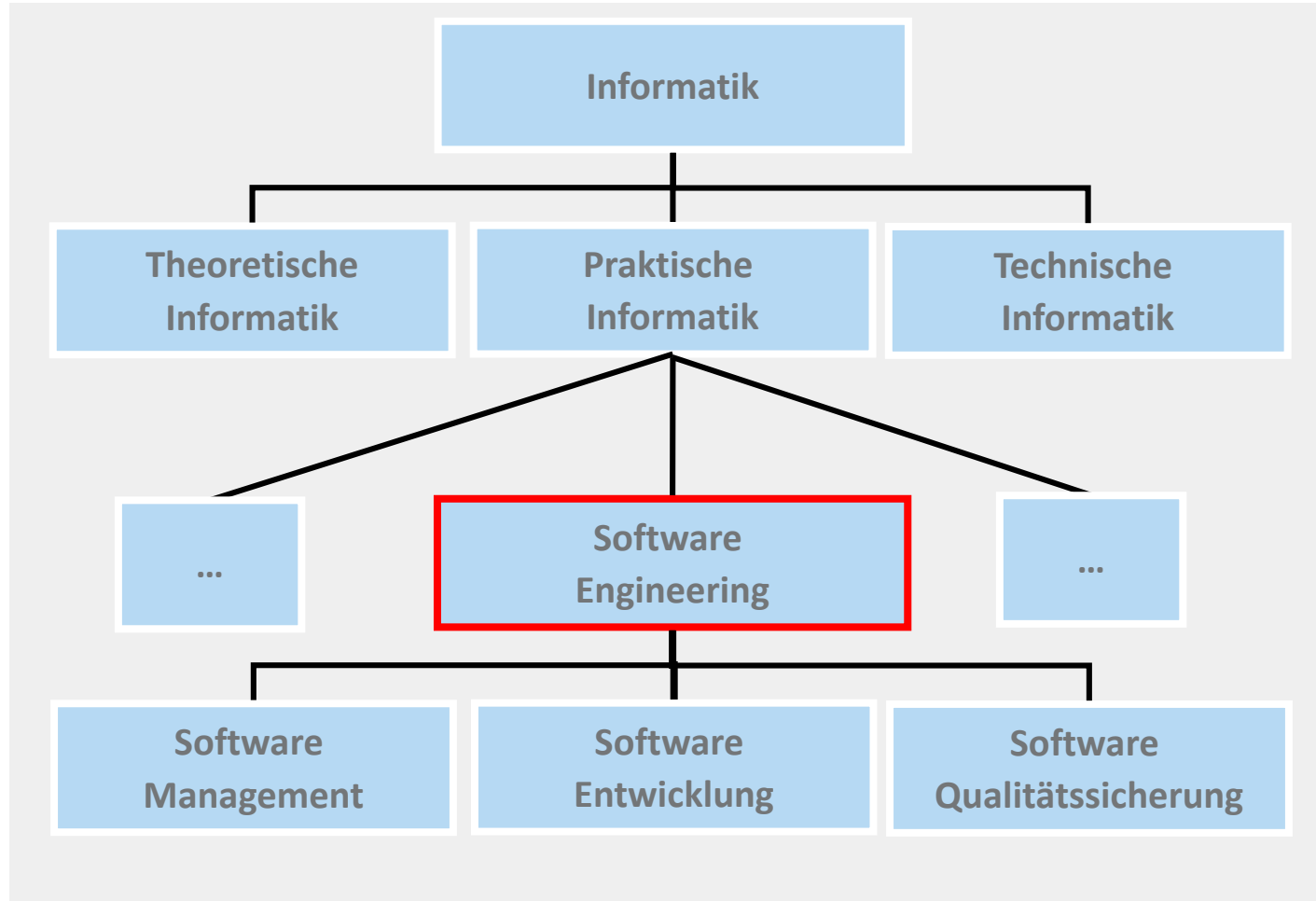
Google wirbt um SW Ingenieure



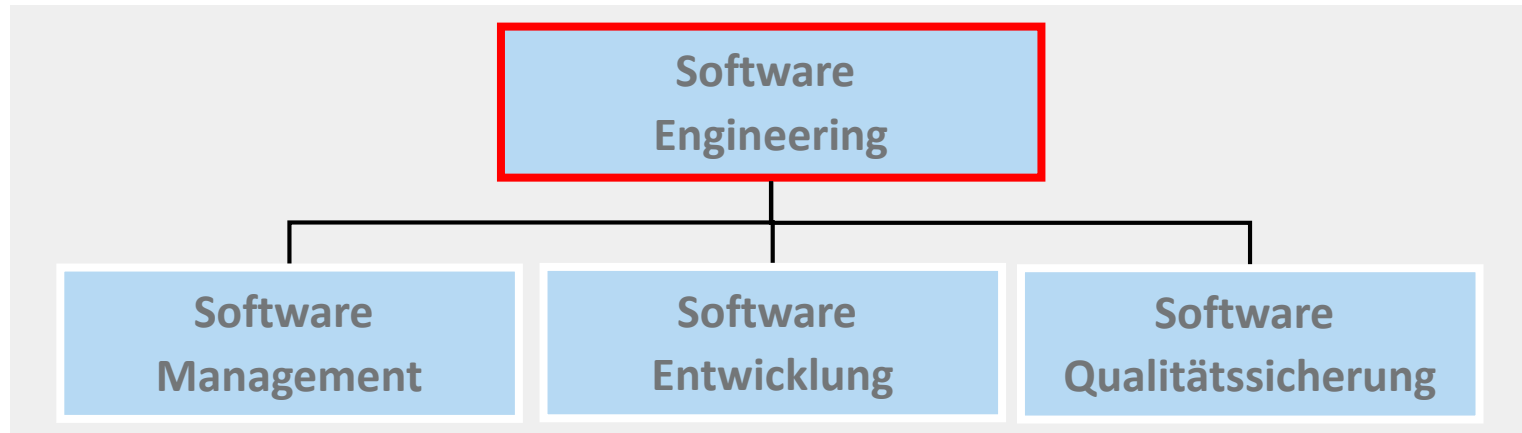
Einführung

- Motivation
- Themen des Software Engineerings
- Themen dieser Veranstaltung

Einordnung Software Engineering



Einordnung Software Engineering



- **Software-Management**
 - Planung, Organisation, Leitung und Kontrolle einer Software-Entwicklung
- **Software-Entwicklung**
 - Aus einem geplanten Software-Produkt ein fertiges Software-Produkt entwickeln, das die geforderten Eigenschaften besitzt.
- **Software-Qualitätssicherung**
 - Sicherstellung der geforderten Produkt- und Prozessqualität einer Software-Entwicklung durch geeignete konstruktive und analytische Maßnahmen.

Themen des SWE

Projektmanagement

Vorgehensmodellierung

Software Entwicklungsmethoden

Requirements
Engineering

Software
Architektur
und -Entwurf

Software
Wartung

Re-Engineering
(Sanierung)

Qualitätsmanagement (incl. Testverfahren)

Notationen und Sprachen (UML, BPMN, Java, ...)

Werkzeugunterstützung (incl. CASE, SVN, ant, ...)

Einführung

- Motivation
- Themen des Software Engineerings
- Themen dieser Veranstaltung

Themen dieser Veranstaltung

Projektmanagement

Vorgehensmodellierung

Software Entwicklungsmethoden

Requirements
Engineering

Software
Architektur
und -Entwurf

Software
Wartung

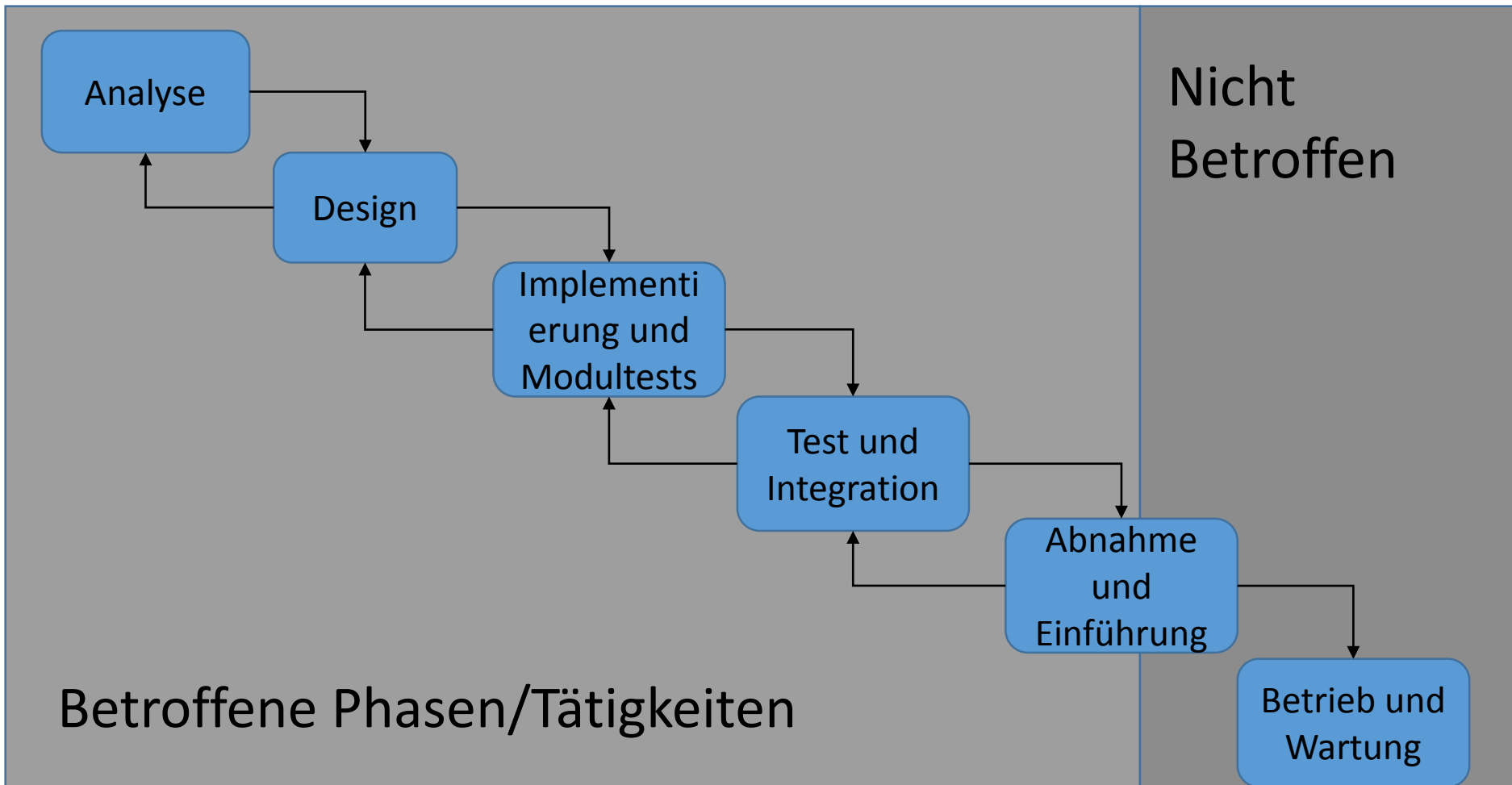
Re-Engineering
(Sanierung)

Qualitätsmanagement (incl. Testverfahren)

Notationen und Sprachen (UML, BPMN, Java, ...)

Werkzeugunterstützung (incl. CASE, SVN, ant, ...)

Themen der Vorlesung



Inhalt der Vorlesung

- Einführung
- Kommunikation
- Software Qualität
- Vorgehensmodelle
- Requirements Engineering
- Software Architektur und – Design
- Konfiguration Management