

Vorlesung Softwaretechnik - Einführung -

Prof. Dr. Klaus-Peter Fährnich

WS 2006/2007

Verantwortlicher Lehrstuhl: Betriebliche Informationssysteme

Modulturnus: jedes Wintersemester

Verwendbarkeit: Pflichtmodul im Bachelor of Science Informatik

Workload

- Vergabe von Leistungspunkten:
Klausur als Modulklausur
- Empfohlen für 3. Semester
- Dauer: 1 Semester
- Arbeitsaufwand: 5 LP = 150 Arbeitsstunden (Workload)

Lehrformen

- Vorlesung "Softwaretechnik" (2 SWS)
= 30 h Präsenzzeit und 70 h Selbststudium = 100 h
- Übung "Softwaretechnik" (1 SWS)
= 15 h Präsenzzeit und 35 h Selbststudium = 50 h

Übungsbetrieb/Kalender

Ort: Mensa Jahnallee
Zeit: Di 13:00 - 14:30; Start: 17.10.

Oktober 2006

Woche	Mo	Di	Mi	Do	Fr
B	2	3	4	5	6
A	9	10	11	12	13
B	16	17	18	19	20
A	23	24	25	26	27

November 2006

Woche	Mo	Di	Mi	Do	Fr
B	30	31	1	2	3
A	6	7	8	9	10
B	13	14	15	16	17
A	20	21	22	23	24
B	27	28	29	30	1

Dezember 2006

Woche	Mo	Di	Mi	Do	Fr
A	4	5	6	7	8
B	11	12	13	14	15
A	18	19	20	21	22
B	25	26	27	28	29

Januar/Februar 2007

Woche	Mo	Di	Mi	Do	Fr
A	1	2	3	4	5
B	8	9	10	11	12
A	15	16	17	18	19
B	22	23	24	25	26
A	29	30	31	1	2

Übungstermine
 Vorlesungstermine
 Abgabe der Übungsserien
 freie Tage
 Klausurtermin: *wird noch festgelegt*

Fällt der Übungstermin auf den 22.11. (Buß- und Betttag), so ist einer der anderen angebotenen Übungstermine zu besuchen.

Übungsgruppen

Name	Zeit	Ort	Leiter	Belegung
1	mittwochs, A-Woche, 09.30–11.00	SG 3-01	Thomas Riechert	20 / 30
2	donnerstags, A-Woche, 11.15–12.45	SG 3-01	Thomas Riechert	17 / 30
3	mittwochs, B-Woche, 09.30–11.00	SG 3-01	Thomas Riechert	15 / 30
4	donnerstags, B-Woche, 11.15–12.45	SG 3-01	Thomas Riechert	21 / 30

- Der **Lehrstuhl „Betriebliche Informationssysteme“** besteht am Institut für Informatik der Universität Leipzig seit dem Jahr 2000 unter Leitung von Prof. Dr. Klaus-Peter Fährnich.
- Der Mitarbeiterstamm von derzeit **14 Mitarbeitern** wird ergänzt und erweitert durch **7 Doktoranden** und etwa **10 studentischen Hilfskräften**.



Auf **Anwendungssysteme** und die zur Entwicklung notwendige **Softwaretechnik** fokussiert, umfassen die Forschungs- und Lehrgebiete:

- Betriebliche Informationssysteme,
- E-Business,
- Dienstleistungsinformatik,
- Content- und Wissensmanagement,
- Software Engineering,
- Enterprise Application Integration,
- Service Engineering und
- Service Management.

Modulübersicht: Bachelor / Master

Wintersemester

- **Softwaretechnik (Bachelor)**
 - Vorlesung Softwaretechnik (30h)
 - Übung Softwaretechnik (15h)
 - Empfohlenes Semester: 3 - Pflichtmodul

- **Einführung in das symbolische Rechnen (Bachelor)**
 - Vorlesung: Einführung in das symbolische Rechnen (30h)
 - Übung: Einführung in das symbolische Rechnen (15h)
 - Empfohlenes Semester: 5 (Bachelor) - Wahlpflichtmodul

- **Software aus Komponenten (Master)**
 - Vorlesung: Software aus Komponenten (30h)
 - Seminar Komponentensoftware (15)
 - Empfohlenes Semester: 1 (Master) - Wahlpflichtmodul

- **Softwaresystemfamilien (Master)**
 - Vorlesung Generative Softwareentwicklung (30h)
 - Vorlesung Software Ökonomie (30h)
 - Seminar Generative Softwareentwicklung (30h)
 - Empfohlenes Semester: 1/3 (Master) – Wahlpflichtmodul

Modulübersicht: Bachelor / Master

Sommersemester

- **Modul: Softwaretechnik-Praktikum (Bachelor)**
 - Softwaretechnik-Praktikum
 - Empfohlenes Semester: 4 - Pflichtmodul

- **Betriebliche Informationssysteme (Bachelor/Master)**
 - Vorlesung: Betriebliche Informationssysteme (30h)
 - Vorlesung: Einführung in XML (30h)
 - Seminar: Betriebliche Informationssysteme (30h)
 - Praktikum: Betriebliche Informationssysteme (30h)
 - Empfohlenes Semester: 6 (Bachelor), 2 (Master) - Wahlpflichtmodul

- **Software-Management (Master)**
 - Vorlesung Software Management (30h)
 - Vorlesung Software-Qualitätsmanagement (30h)
 - Vorlesung Engineering IT-basierter Dienstleistungen (30h)
 - Empfohlenes Semester: 2 (Master) - Wahlpflichtmodul



Pre Built Information Space

<http://www.prebis.de/>

- Rollen- und aufgabenangepasste Informationsbeschaffung;
- Entwicklung eines Systems für das Content Engineering von dynamischen, lernenden und aufgaben- und rollenangepassten multimedialer Informations-räumen.



Integration Engineering

<http://ie.informatik.uni-leipzig.de/>

- Entwicklung eines Vorgehensmodells zur Einbettung neuer oder modifizierter Geschäftsprozesse in bestehende IT-Landschaften mit Hilfe einer integrierenden, internetbasierten IT-Struktur.



ServCASE

<http://www.servcase.de/>

- Computer Aided Engineering für IT-basierte Dienstleistungen;
- Entwicklung und Umsetzung eines Referenzmodells für das Co-Design von Software und Services einschließlich entsprechender Softwarewerkzeuge.



OrViA

<http://www.orvia.de/>

- Das Forschungsvorhaben OrViA konzipiert und realisiert eine Architektur zur Orchestrierung und Validierung kooperierender Systemkomponenten in geschäftskritischen E-Business-Umgebungen für kleine und mittlere Unternehmen.



SoftWiki

<http://www.softwiki.de/>

- **Anwenderorientiertes stark verteiltes Requirements Engineering für evolutionäre Entwicklungsprozesse**
- Das Verbundvorhaben SoftWiki hat zum Ziel die Zusammenarbeit aller am Softwareentwicklungsprozess Beteiligten in Bezug auf Softwareanforderungen zu unterstützen. Potentiell sehr große und räumlich getrennte Stakeholdergruppen sollen dabei befähigt werden Software-Anforderungen auf einfachste Weise zusammenzutragen, semantisch anzureichern, zu klassifizieren und zu aggregieren.



Services made in Germany

<http://www.services-made-in-germany.de>

- Zielsetzung des Forschungsvorhabens ist es, wissenschaftlich-methodische Grundlagen für die Internationalisierung der deutschen Dienstleistungswirtschaft zu erarbeiten.
- Hierfür entwickeln die Projektpartner ganzheitliche Konzepte und Instrumentarien, so dass Dienstleistungen im internationalen Umfeld erfolgreich sind.

- Lernziele
- Taxonomie
- Definitionen
- Problematik der Softwareentwicklung
- Bedeutung von Software
- Produktivitätsfortschritt
- Standardsoftware
- Schwierigkeiten bei der Entwicklung
- Softwaretechnik

2. Unterschiede aufzeigen, die es zwischen Software und anderen Produkten gibt;
3. Beschreiben von Veränderungen der Software in den letzten 10 Jahren;
4. Hohe Portabilitätsanforderungen als Erschwernis bei der Software-Erstellung;
5. Problem bei hoher Änderungshäufigkeit während Entwicklung und Wartung;
6. Beschreibung der Disziplin Softwaretechnik anhand von Begriffen;
7. Terminologie der Begriffe System und Software.

Lernzieltaxonomie

Lernstoff in 4 Lernstufen unterteilt:

Lernstufe Wissen

Sie wird definiert durch elementare Kenntnisse. Darunter ist die Wiedergabe von Begriffen, Fakten, Klassifikationen und Kriterien zu verstehen.

Lernstufe Verstehen

Sie wird definiert durch funktionale Kenntnisse. Darunter sind u.a. Beschreibung von Verfahren, Methoden, Regeln und Gesetzmäßigkeiten zu verstehen.

Lernstufe Anwenden

Sie definiert durch den sachkundigen Umgang mit Formeln und Verfahren zur Lösung von Problemen, zu denen die Übertragung von „Wissen“ und „Verstehen“ in direktem Bezug auf einzelne und konkrete Situationen notwendig ist.

Lernstufe Beurteilen

Sie wird definiert durch die Lösung komplexer Aufgaben, zu denen anhand von Analysen Auswahlentscheidungen zu treffen und/oder Verfahren zu entwickeln sind.

Quelle: Balzert

Software-Definitionen

- **Software** (engl., eigtl. »weiche Ware«), Abk. SW, Sammelbezeichnung für Programme, die für den Betrieb von Rechensystemen zur Verfügung stehen, einschl. der zugehörigen Dokumentation (Brockhaus Enzyklopädie)
- **Software**, die zum Betrieb einer Datenverarbeitungsanlage erforderlichen nichtapparativen Funktionsbestandteile (Fremdwörter-Duden)
- **Software**: ... unter Software subsumiert man alle immateriellen Teile, d. h. alle auf einer Datenverarbeitungsanlage einsetzbaren Programme (Lexikon der Informatik und Datenverarbeitung [Schneider86])
- **Software**: Menge von Programmen oder Daten zusammen mit begleitenden Dokumenten, die für ihre Anwendung notwendig oder hilfreich sind
(Ein Begriffssystem für die Softwaretechnik [Hesse84]).
- **Software**: Computer programs, procedures, rules, and possibly associated documentation and data pertaining to the operation of a computer system
(IEEE Standard Glossary of Software Engineering Terminology [ANSI83]).

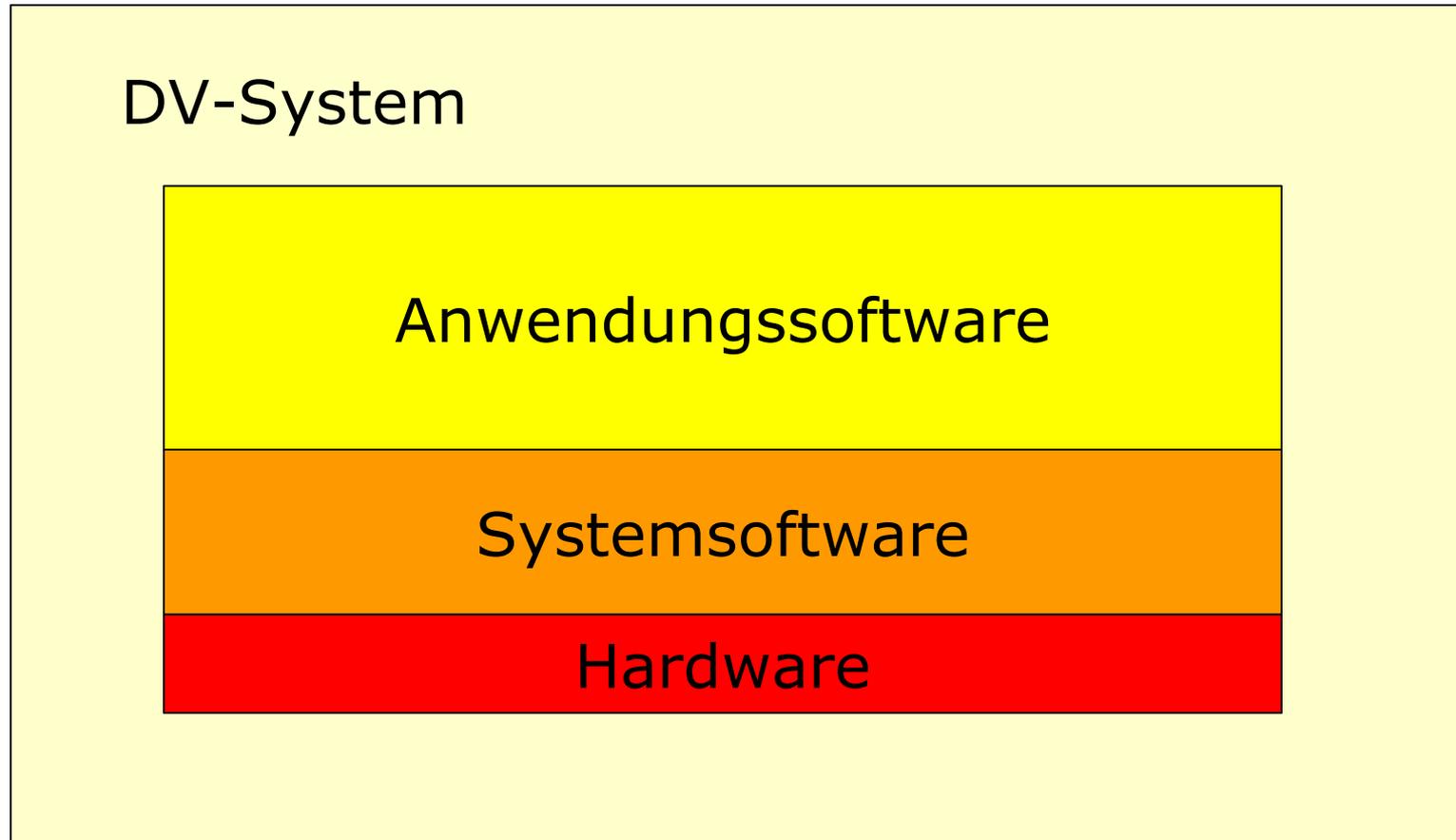
Software-Definitionen

- **Software-Produkt**
 - Ein Produkt ist ein in sich abgeschlossenes, i. A. für einen Auftraggeber bestimmtes Ergebnis eines erfolgreich durchgeführten Projekts oder Herstellungsprozesses. Als Teilprodukt bezeichnen wir einen abgeschlossenen Teil eines Produkts.
 - SW-Produkt: Produkt, das aus Software besteht.

- **Software-System**
 - Unter einem System wird ein Ausschnitt aus der realen oder gedanklichen Welt, bestehend aus Gegenständen (z. B. Menschen, Materialien, Maschinen oder anderen Produkten) und darauf vorhandenen Strukturen (z. B. deren Aufbau aus Teileinheiten oder Beziehungen untereinander) verstanden. [Hesse84].
 - Software-System ist dementsprechend ein System, dessen Systemkomponenten und Systemelemente aus Software bestehen.

Software – Definitionen

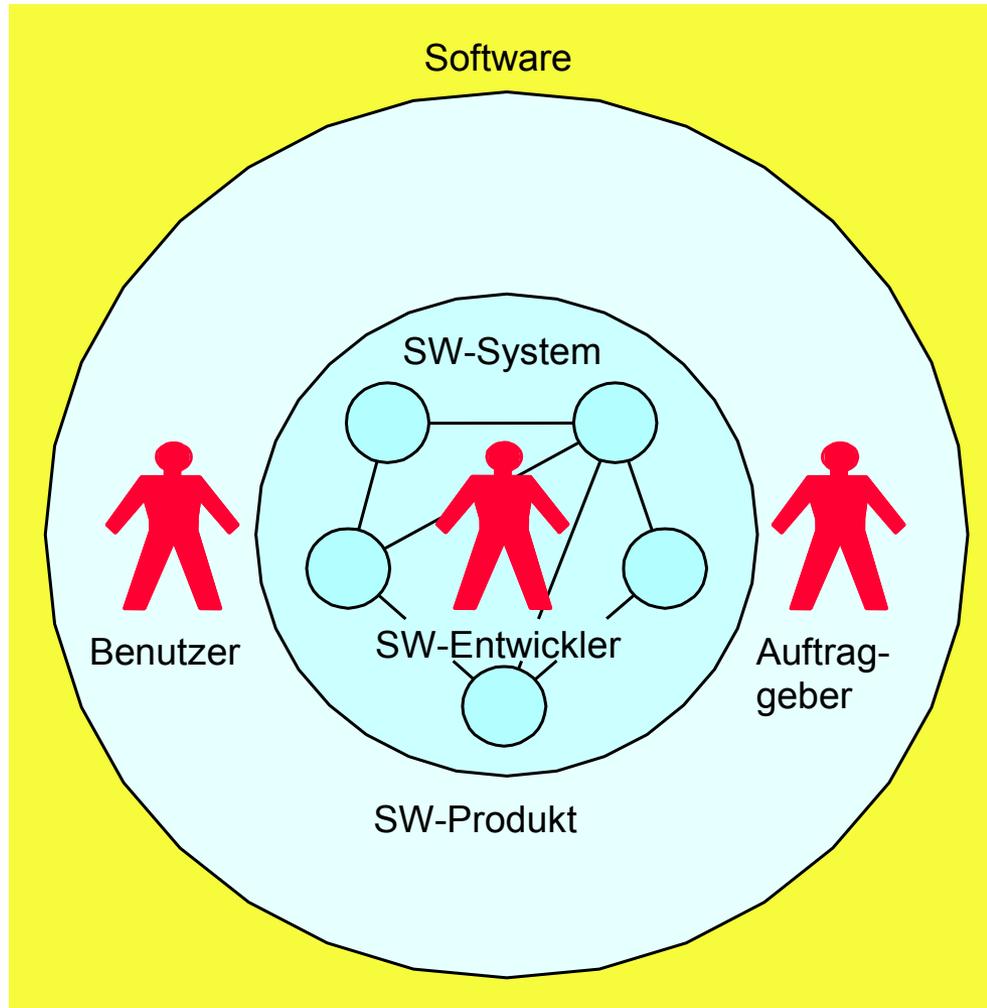
- **Systemsoftware**
 - Software die für eine spezielle Hardware oder Hardwarefamilie entwickelt wurde um den Betrieb und die Wartung dieser Hardware zu ermöglichen. Dazu gehören das Betriebssystem, Compiler, ...
 - Orientiert sich grundsätzlich an den Eigenschaften der Hardware, für die sie geschaffen wurde und ergänzt deren Fähigkeiten
- **Anwendungssoftware** (application software)
 - Software die Aufgabe des Nutzers mit Hilfe eines Computersystems löst.
 - Setzt in Regel auf der Systemsoftware der verwendeten Hardware auf bzw. benutzt sie zur Erfüllung der eigenen Aufgaben
- **Computersystem** (DV-System)
 - Anwendungssoftware + Systemsoftware + Hardware
- **Anwender**
 - Angehörige einer Institution oder organisatorischen Einheit die ein Computersystem zur Erfüllung ihrer fachlichen Aufgaben einsetzen
- **Benutzer**
 - Personen, die ein Computersystem unmittelbar einsetzen und bedienen
- **Technisches System**
 - Computersystem + technische Einrichtungen

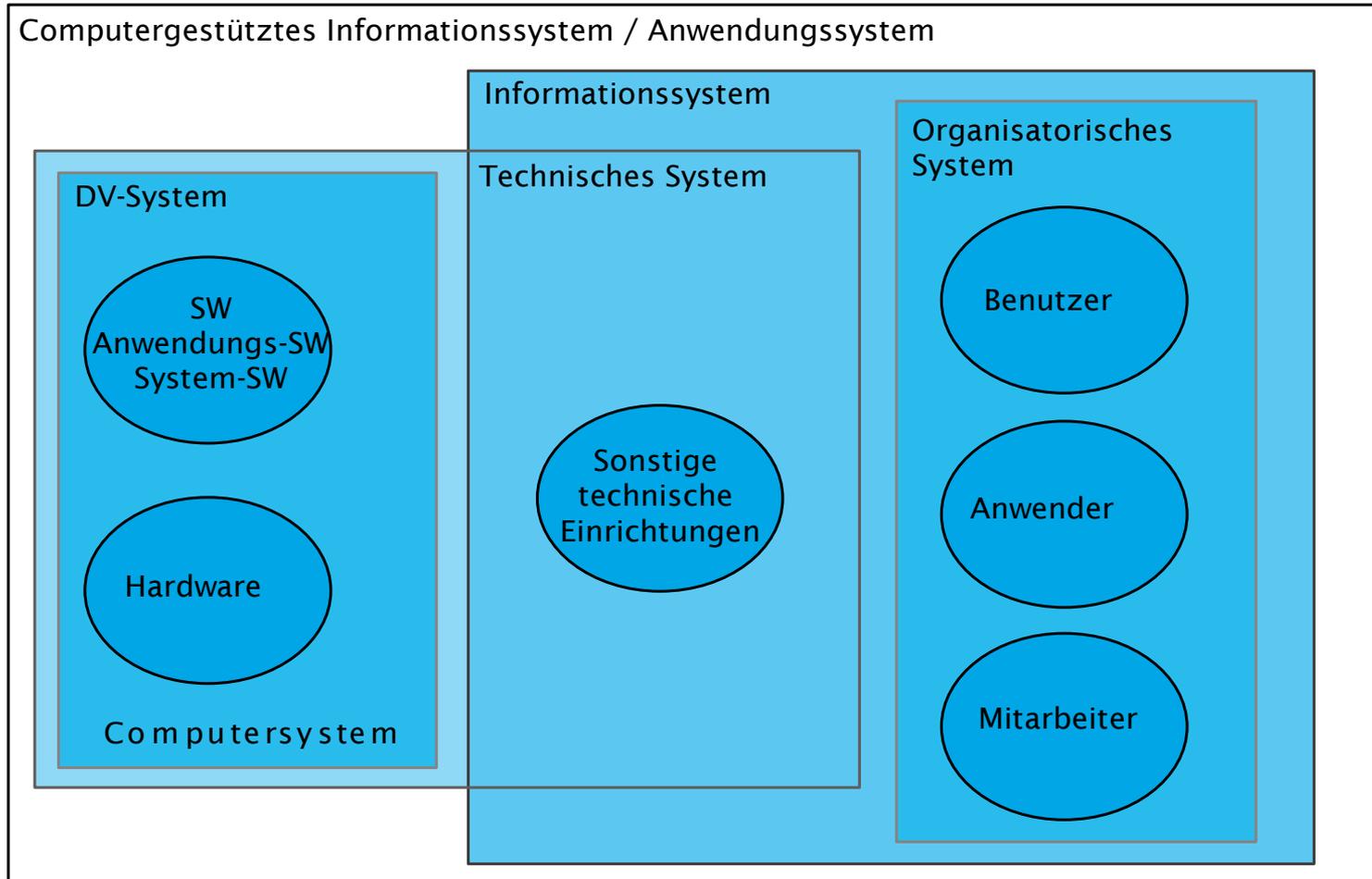


Software – Definitionen

- **Organisatorisches System**
 - Mitarbeiter in ihrer Rolle als Auftraggeber einschließlich Anwendern und Benutzern
- **Informationssystem**
 - Menschen und Maschinen die Information erzeugen und/oder benutzen und durch Kommunikationsbeziehungen verbunden sind. Enthält es mehrere Computersysteme, so spricht man von einem **computergestützten Informationssystem**
- **Computergestütztes Informationssystem**
 - System bei dem die Erfassung, Speicherung, Übertragung, Auswertung und/oder Transformation von Information durch Computersysteme teilweise automatisiert ist
- **Software-Entwicklung**
 - Ausschließliche Entwicklung von Software
- **System-Entwicklung**
 - Entwicklung eines Systems, dass aus Hardware und Softwarekomponenten besteht.

Software-Produkt und Software-System



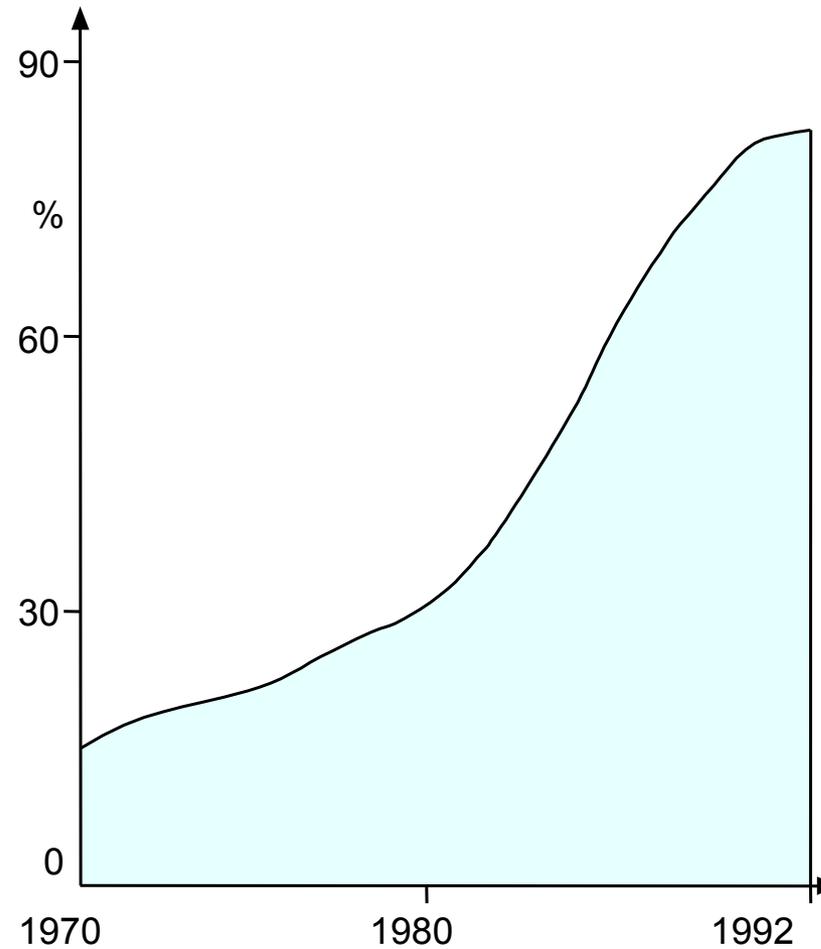


Legende: ● Systemkomponente bzw. Systemelement

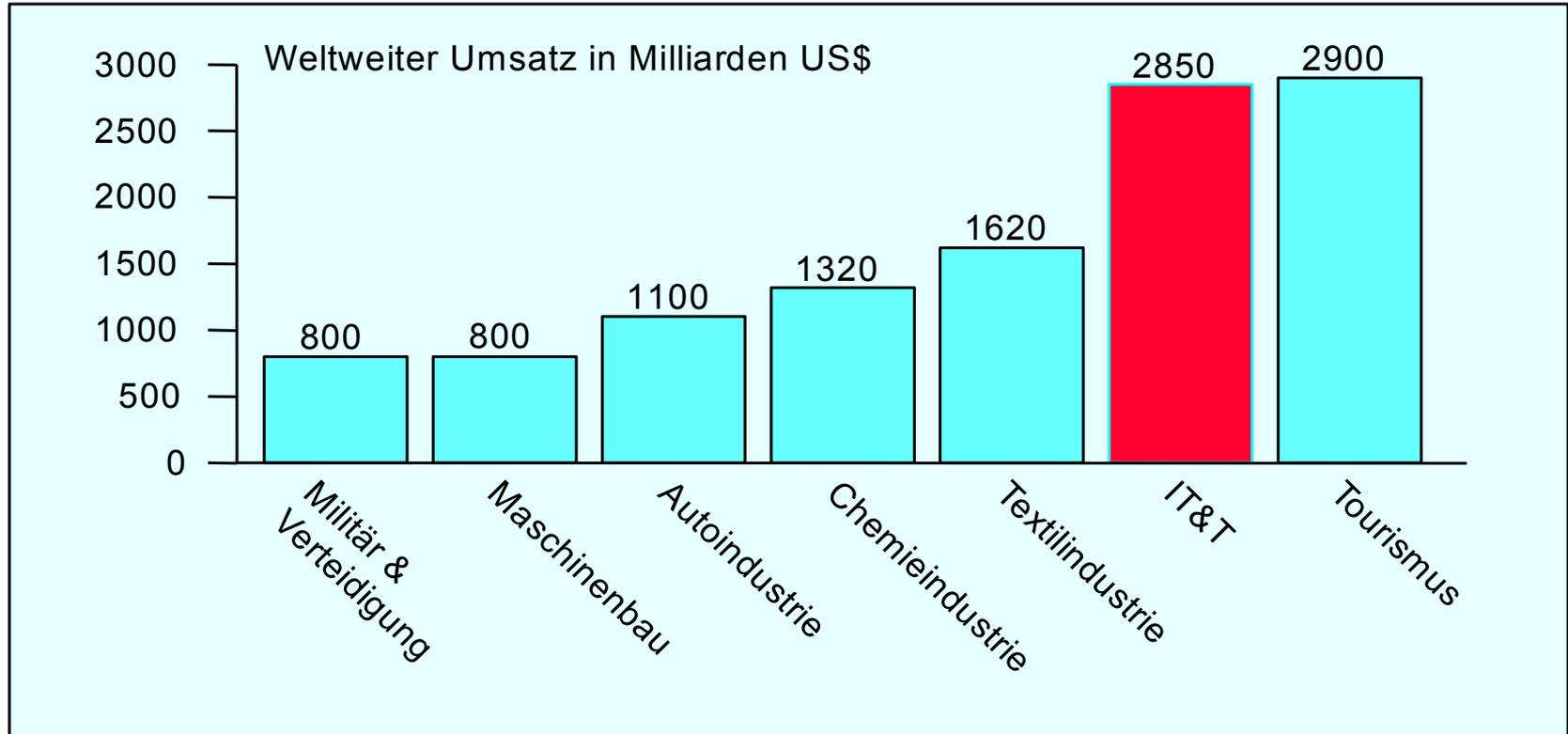
Problematik der Softwareentwicklung

- Software ist ein immaterielles Produkt;
- Software unterliegt keinem Verschleiß;
- Software wird nicht durch physikalische Gesetze begrenzt;
- Software ist im Allgemeinen leichter und schneller änderbar als ein technisches Produkt;
- Für Software gibt es keine Ersatzteile;
- Software altert;
- Software ist schwer zu vermessen.

Zunehmender Wertanteil der Software beim Kauf eines Computersystems

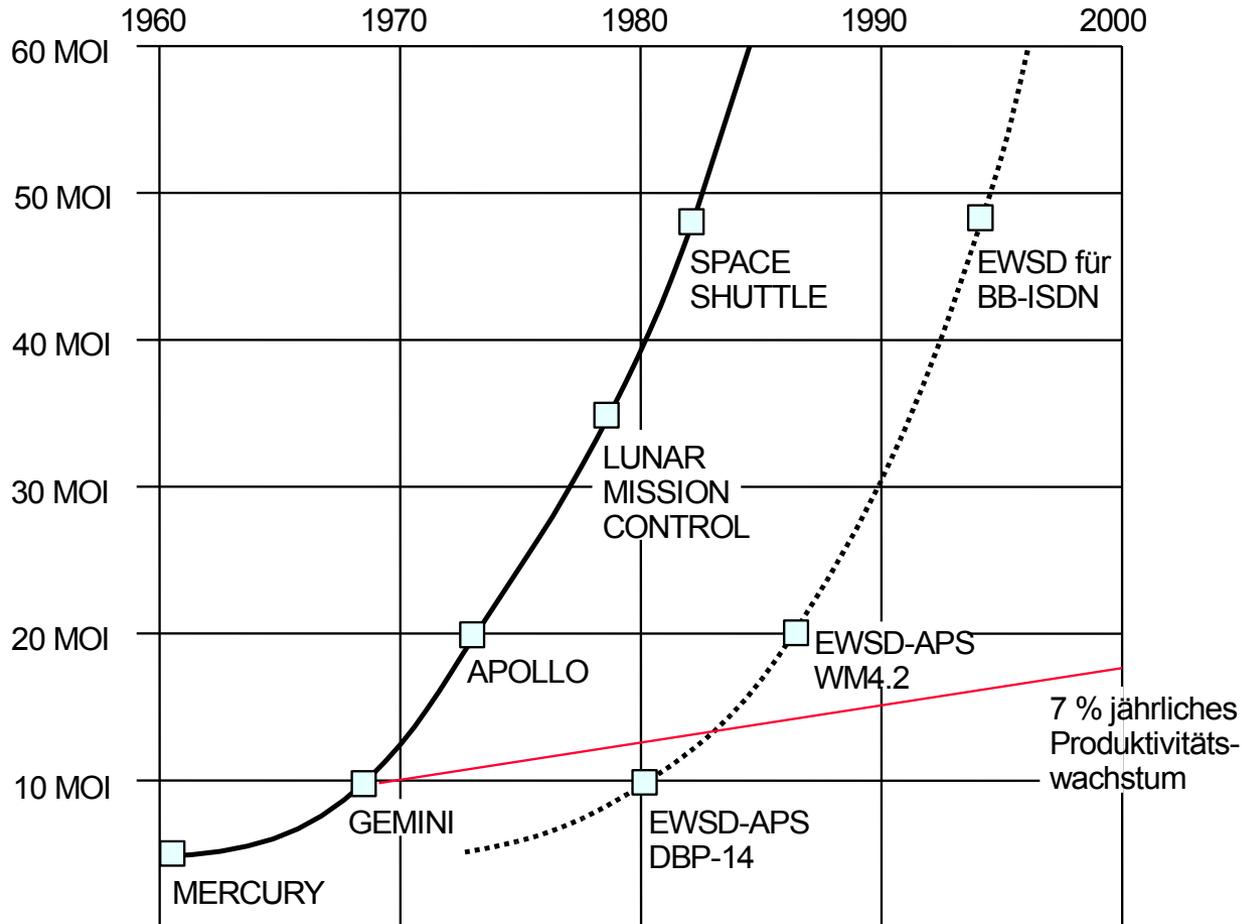


Quellen: Hughes AircraftGEI
(zitiert nach Scharf88)



Quelle: G. Koch, European IT Conference (EITC), Brüssel, Juni 1994

Wachsende Komplexität

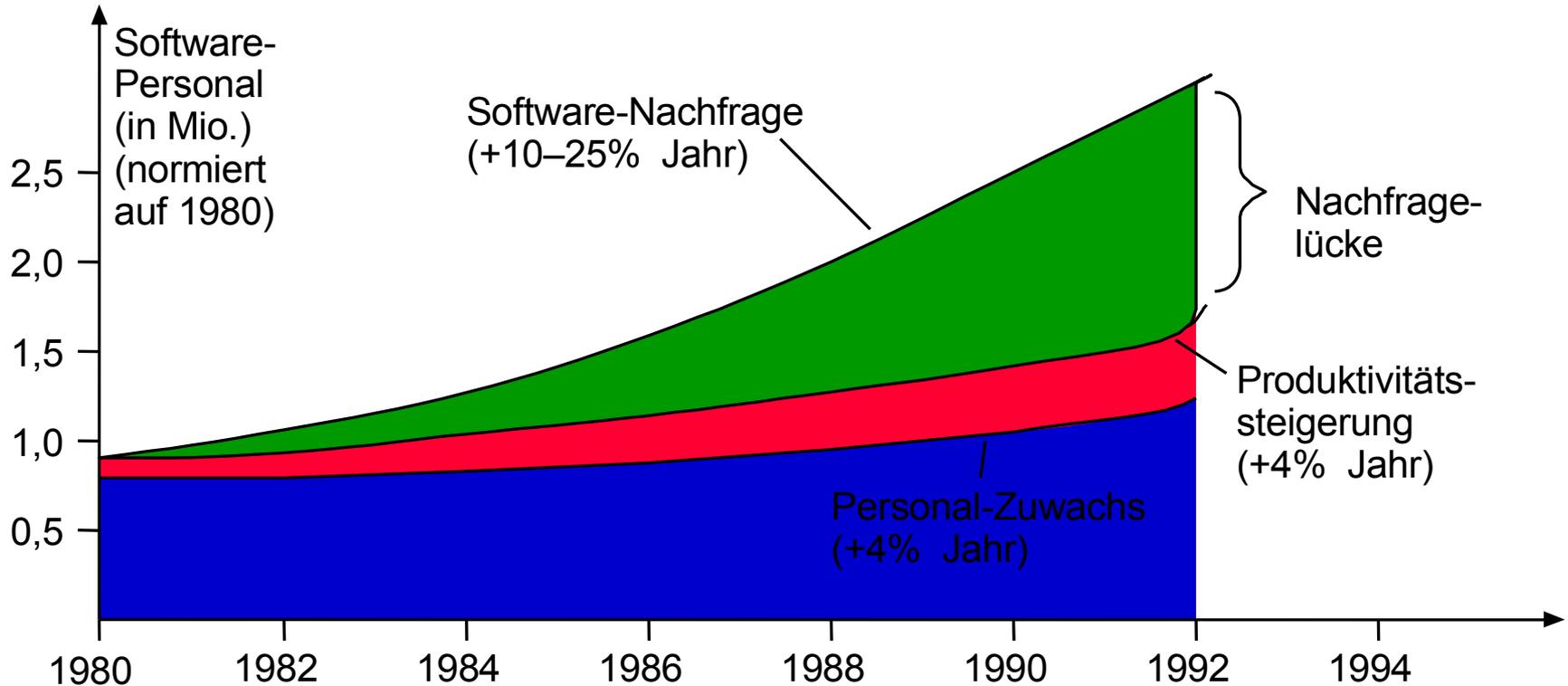


MOI: Millionen Objektcode-Instruktionen

EWSD: Elektronisches Wählsystem Digital

Quellen: Boehm 87, S.45 und Siemens (Unterlagen zum Seminar Industrielle Software-Technik, Deutsche Informatik-Akademie Bonn 588)

Software nachfrage und -angebot



Quelle: G. Koch, European IT Conference, Brussels, June 1994

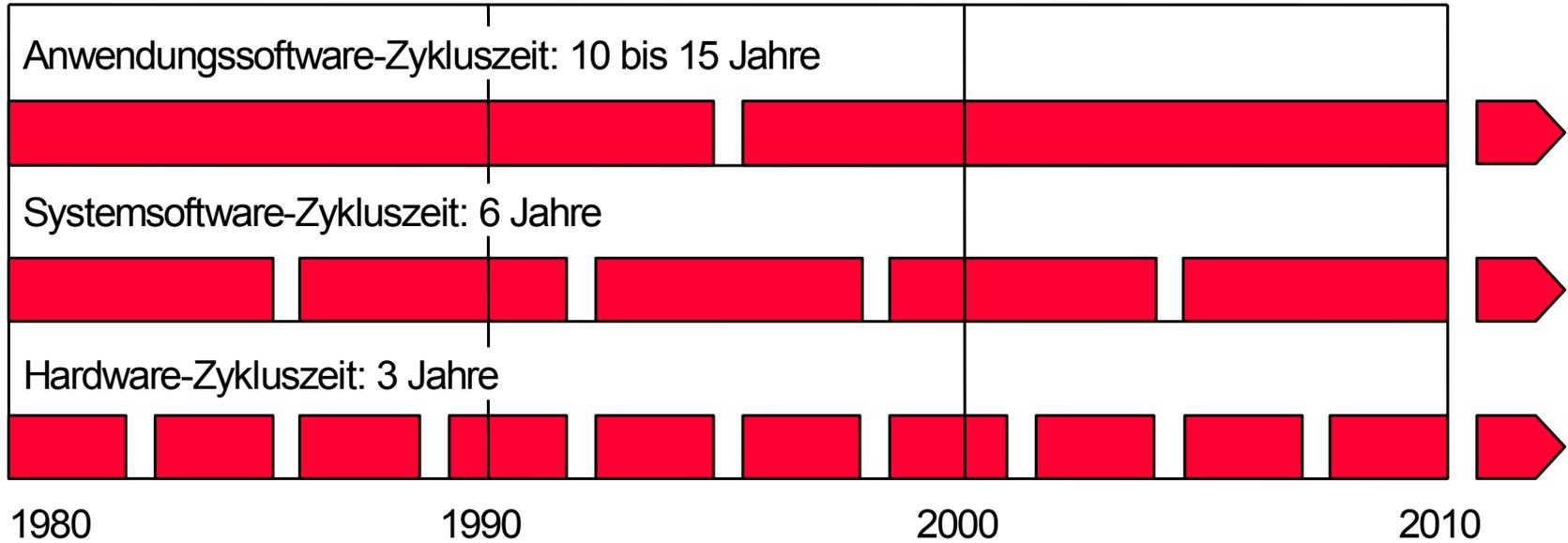
Art der Softwarelösung	Auftraggeber-Firmen (Personalgröße)
Individual-Spezialsoftware	50.000 – 100.000
Standardsoftware & individuelle Anpassungen	3.000 – 50.000
Lösungen zusammengesetzt aus Standard-Softwarekomponenten	500 – 3.000
Vollständige Lösungen durch Standardsoftware	500<

Quelle: G. Koch, European IT Conference, Brussels, June 1994

Schwierigkeiten bei der Entwicklung

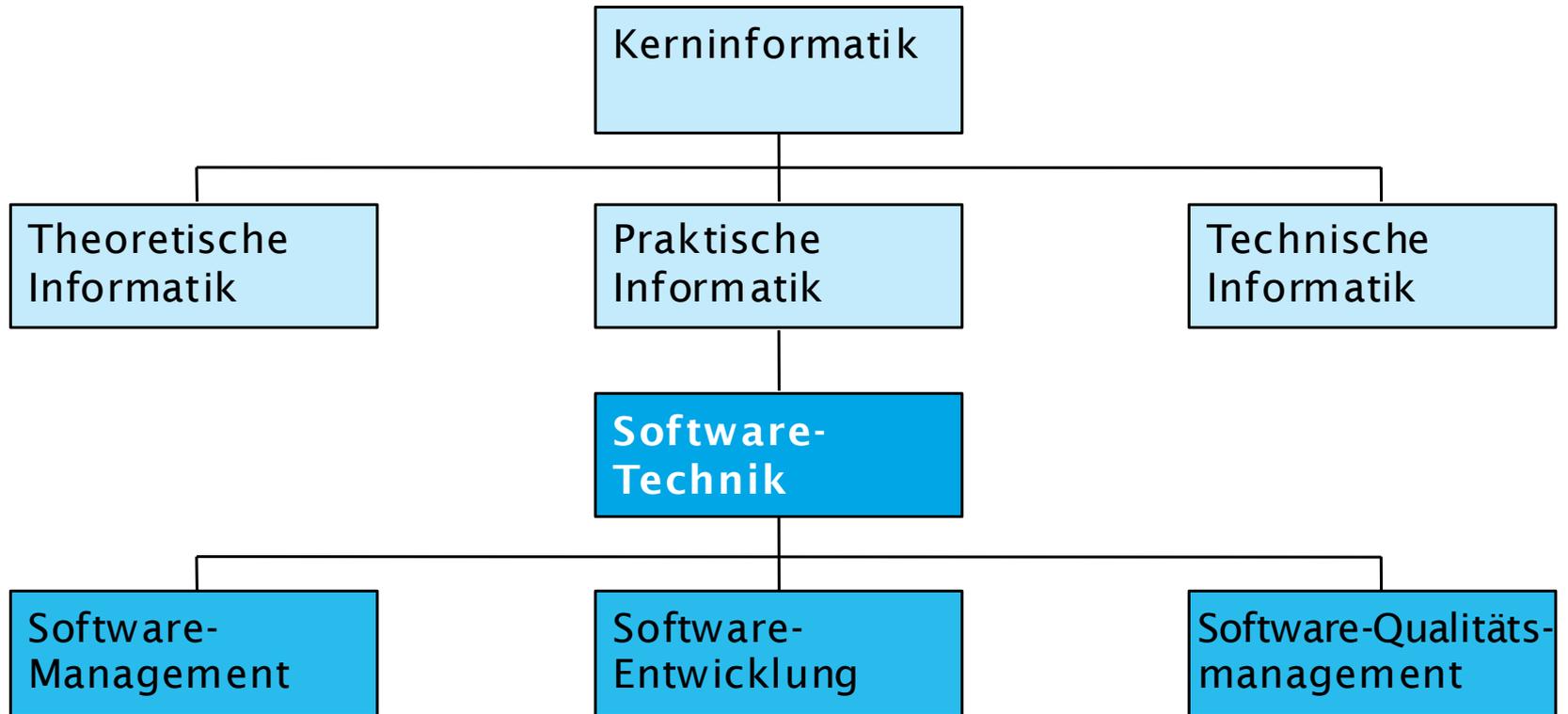
- Zunehmend Außer-Haus-Entwicklung
 - Trend:
 - o Software nicht selbst zu entwickeln, sondern Auftragsentwicklung
 - Prognose:
 - o Von den Software-Produkten und den zugehörigen Dienstleistungen werden generell etwa **55% intern** und **45% extern** erbracht werden
 - o Durch die zunehmende Produktintegration von Software (eingebettete Systeme) wird der Prozentsatz intern erstellter Software nicht drastisch zurückgehen.
- Zunehmend Altlasten
 - Anwendungssoftware wird oft 20 Jahre und länger eingesetzt
 - Da sich die Einsatzumgebung dieser Anwendungssoftware ständig ändert, muss diese Software ebenfalls ständig angepasst werden
 - Diese permanenten Anpassungsprozesse verursachen oft 2/3 aller Software-Kosten

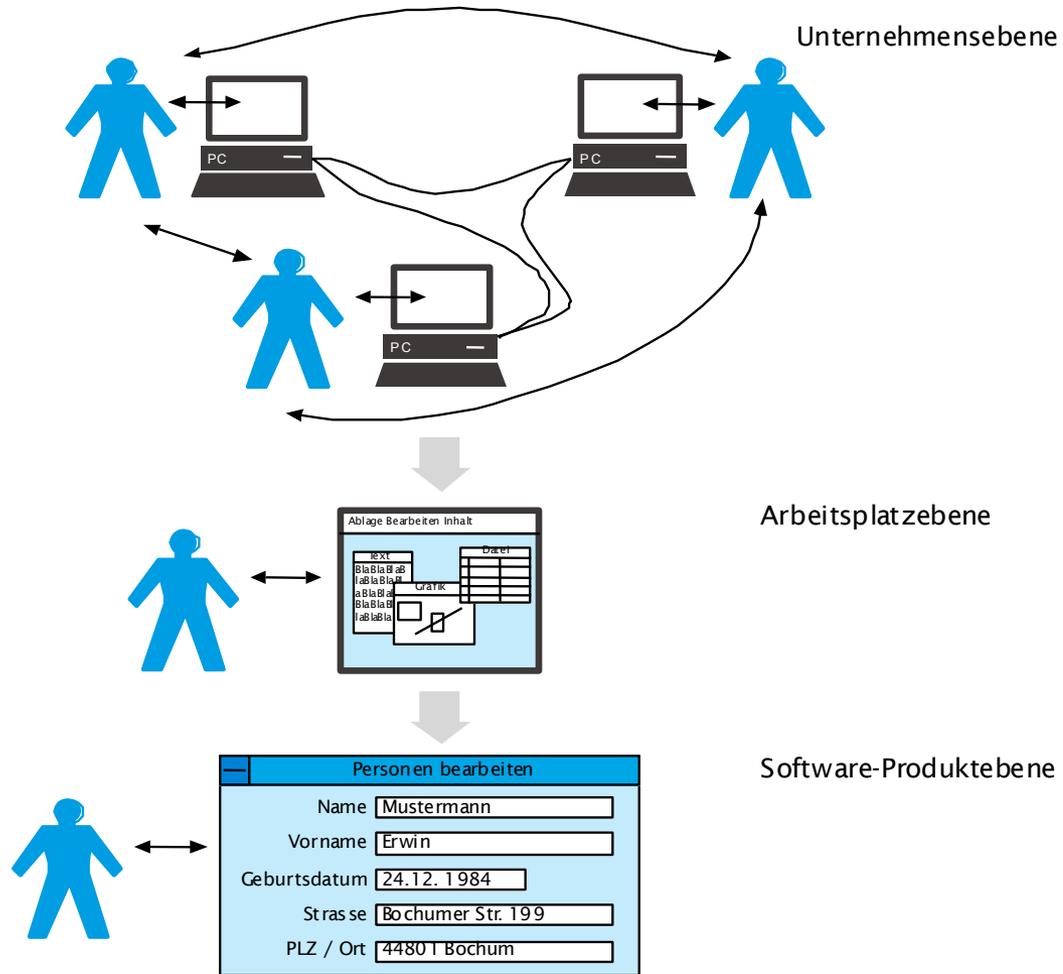
Schwierigkeiten bei der Entwicklung

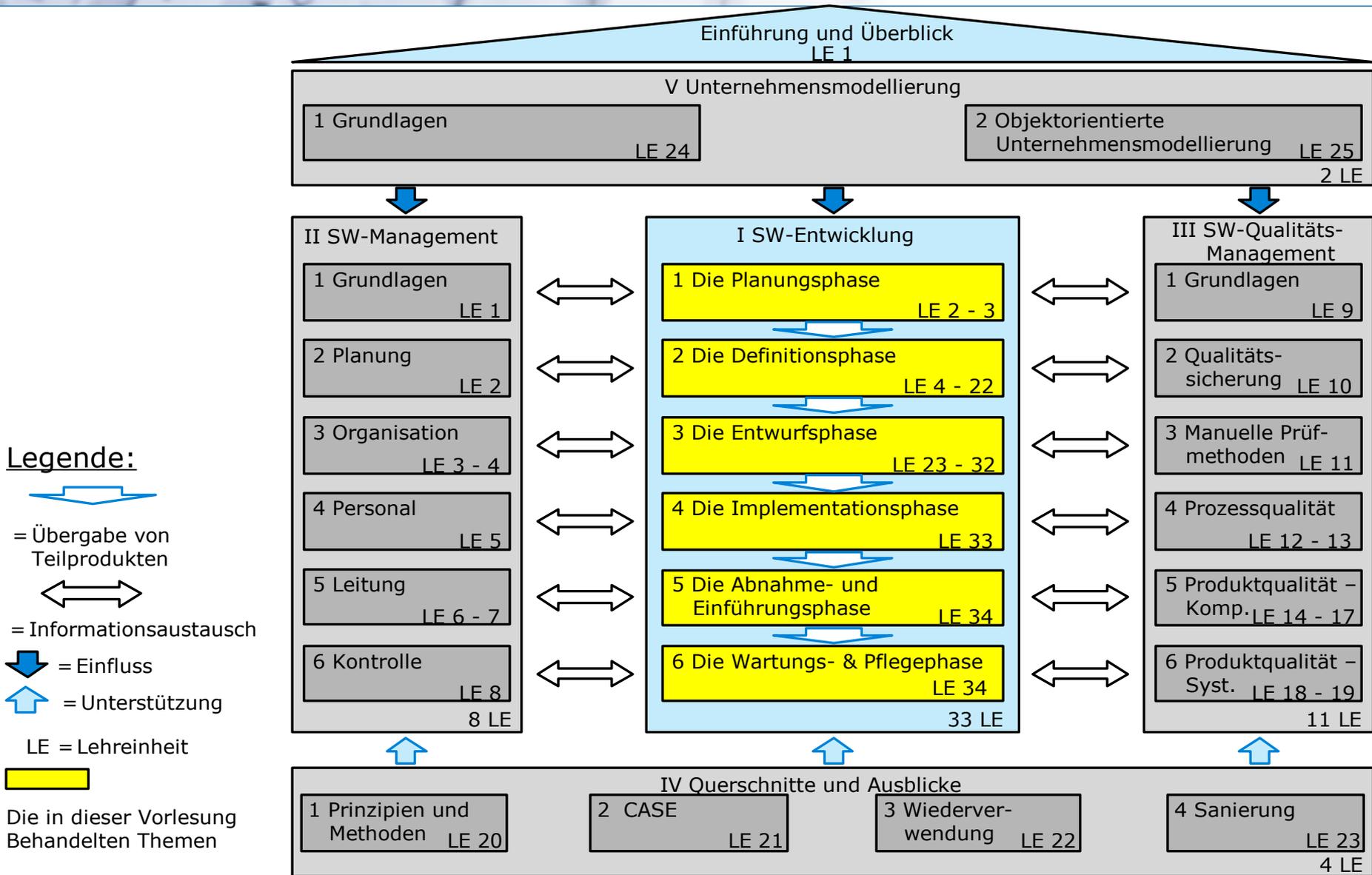


Softwaretechnik

- **Softwaretechnik** (syn.: Software-Engineering): Fachgebiet der Informatik, das sich mit der Bereitstellung und systematischen Verwendung von Methoden und Werkzeugen für die Herstellung und Anwendung von Software beschäftigt [Hesse 84]
- **Software-Engineering**
das ingenieurmäßige Entwerfen, Herstellen und Implementieren von Software sowie die ingenieurwissenschaftliche Disziplin, die sich mit Methoden und Verfahren zur Lösung der damit verbundenen Problemstellungen befasst (Brockhaus Enzyklopädie).
- **Software-Technik**
Zielorientierte Bereitstellung und systematische Verwendung von Prinzipien, Methoden und Werkzeugen für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Software-Systemen.







Zusammenfassung (1)

Die **Software-Technik (Software-Engineering)** als Teildisziplin der Informatik befasst sich mit der Herstellung und Anwendung von **Software (SW)**. Dazu ist eine **Software-Entwicklung**, ein **Software-Management** und ein **Software-Qualitätsmanagement** erforderlich. Nach der Inbetriebnahme eines **Software-Produkts** erfolgen die **Wartung** und die **Pflege**.

Viele Aktivitäten, die im Rahmen der Software-Technik anfallen können heute durch **Werkzeuge (tools)**, genauer gesagt **CASE-Werkzeuge**, unterstützt werden. Der Begriff **CASE (Computer Aided Software Engineering)** drückt aus, dass für die Herstellung von Software selbst wieder Software eingesetzt wird und zwar in Form von CASE-Werkzeugen.

Durch Werkzeuge wird der Einsatz von Methoden unterstützt und automatisiert. **Methoden** umfassen in der Software-Technik dabei **methodische Vorgehensweisen, Verfahren, Konzepte** und **Notationen**. Methoden selbst helfen, **Prinzipien** zu verwirklichen.

Ein **Software-System** ist ein System, das aus **Systemkomponenten** bzw. **Subsystemen** aufgebaut ist, die wiederum letztlich aus **Systemelementen** bestehen.

Zusammenfassung (2)

Software kann man in **Anwendungssoftware (application SW)** und **Systemsoftware** gliedern. Beide zusammen mit der Hardware bilden ein **Computersystem** bzw. ein **DV-System**. **Benutzer** benutzen Computersysteme direkt, **Anwender** liefern Information für Computersysteme und nutzen ihre Ergebnisse. Anwender, Benutzer und sonstige Mitarbeiter bilden ein **organisatorisches System**, Computer und sonstige technische Einrichtungen ein **technisches System**. Beide zusammen ergeben ein **computergestütztes Informationssystem** auch **Anwendungssystem** oder kurz **Anwendung** genannt. Fehlen die Computersysteme, dann spricht man von einem **Informationssystem**.

- ANSI/IEEE Std. 729-1983, IEEE Standard Glossary of Software Engineering Terminology, IEEE Inc., New York, 1983.
- Balzert H., Lehrbuch Grundlagen der Informatik, Spektrum Akademischer Verlag, Heidelberg, 1999.
- Böhm B.W., Software Engineering in IEEE Transactions on Computers S.1226-1241, Dezember 1976.
- Böhm B.W., Improving Software Productivity in Computer S. 43-57, Sept. 1987.
- Ludewig J., Softwaretechnik in Stuttgart – ein konstruktiver Informatikstudiengang in Informatik-Spektrum, Februar 1999.
- Schneider H.-J., Lexikon der Informatik und Datenverarbeitung. Oldenburg Verlag, München, 1986.