



Karlsruher Institut für Technologie

Vorlesung Softwaretechnik I

Einführung und Überblick

SWT I – Sommersemester 2021

Walter F. Tichy, Christopher Gerking, Tobias Hey



Kurzdefinition Softwaretechnik

Softwaretechnik (engl. *software engineering*) ist die Lehre von der Softwarekonstruktion: der systematischen Entwicklung und Pflege von Softwaresystemen

Diese Vorlesung gibt einen Überblick über die Methoden der Softwarekonstruktion und behandelt essentielles Grundwissen für jede/n ingenieurmäßig arbeitende/n Informatiker/in.

Grobübersicht

- Die Softwaretechnik gibt/sucht Antworten auf die folgenden Fragen:
 - Wie **findet man** heraus, welche **Eigenschaften** zu konstruierende Software haben soll (Anforderungsermittlung)?
 - Wie **beschreibt** man dann diese Eigenschaften (Spezifikation)?
 - Wie **strukturiert** man die Software so, dass sie sich leicht bauen und flexibel verändern lässt (Entwurf)?
 - Wie **verändert** man Software, die keine solche Struktur hat oder deren Struktur man nicht (mehr) versteht (Pflege, Restauration, *Reengineering*)?
 - Wie **vermeidet** man Mängel in Software oder **deckt** sie **auf** (Qualitätssicherung, Test)?
 - Wie **organisiert** man die Arbeit einer Softwarefirma oder -abteilung, um regelmäßig kostengünstige und hochwertige Resultate termingerecht zu erzielen (Prozessmanagement)?
 - ...und weitere Fragen

Softwaretechnik ist wesentlich mehr als Programmieren

- **Planung und Kostenschätzung** sind unerlässlich.
- **Anforderungsermittlung, Spezifikation, Entwurf** geschehen ebenfalls vor der eigentlichen Programmierung.
- **Qualitätssicherung** erstreckt sich auf alle Tätigkeiten (sie wird z.B. mittels Testen, Durchsichten oder Inspektionen der erstellten Arbeitsprodukte durchgeführt).
- **Organisation** von kleinen und großen Entwicklerteams kommt hinzu.
- Programmiersprachen wählt/wechselt man, je nach Anforderung.
(Programmiersprachen hier nicht behandelt)

Inhalt

- Was ist Software?
- Warum ist Software so schwer zu entwickeln?
- Warum ist marktreife Software so schwer zu entwickeln?
- Was ist Softwaretechnik?

Was ist Software?

- Software (engl., „Weichware“, Programmatur, im Gegensatz zur Apparatur), Abkürzung: „SW“
- Sammelbezeichnung für Programme und Daten, die für den **Betrieb** von **Rechensystemen** zur Verfügung stehen, einschl. der zugehörigen Dokumentation (Brockhaus Enzyklopädie)
- die zum **Betrieb** einer **Datenverarbeitungsanlage** erforderlichen nichtapparativen Funktionsbestandteile (Fremdwörter-Duden).

Was ist Software?

- *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 /ANSI 83/).

Was ist Software?

- Beispiele für Programme:
 - Quellprogramme
 - Bibliotheken
 - Testprogramme
- Beispiele für zugehörige Daten:
 - Initialisierungsdaten
 - Dialogtexte (evtl. mehrsprachig)
 - Testdaten
- Beispiele für Dokumentation:
 - Anforderungsdokumentation
 - Funktionsbeschreibungen der einzelnen Klassen/Methoden
 - Test- und Abnahmeprotokolle.
- Mit Software **nicht** gemeint sind *reine Daten*, z.B.
 - Musikstücke, statische Webseiten, Inhalt von Datenbanken.

Was ist Software? (ausführlich)

- Beispiele für Programme: Quellprogramme, Zwischencode, Objektcode, Bibliotheken, Rahmenarchitekturen, Installationsprogramme, Testprogramme, lauffähige Beispiele.
- Beispiele für zugehörige Daten: Initialisierungsdaten, Hilfsinformation, Fehlermeldungen und Dialogtexte (evtl. mehrsprachig), Zeichensätze, Testdaten, erwartete Testausgaben.
- Beispiele für Dokumentation: Anforderungsdokumentation, Architekturbeschreibung, Funktionsbeschreibungen der einzelnen Klassen/Methoden, Test- und Abnahmeprotokolle, Anwendungsbeispiele, Benutzerhandbücher, Fragen und Antworten, interaktive Hilfe.
- Mit Software nicht gemeint sind *reine Daten*, z.B. Musikstücke, Bücher, statische Webseiten, Inhalt von Datenbanken.

Was ist Software?

■ Softwareprodukt

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

Was ist Software?

■ System

- „Ausschnitt aus der realen oder gedanklichen Welt“
 - bestehend aus **realen Gegenständen** (z.B. Menschen, Materialien, Maschinen oder anderen Produkten) und Beziehungen darunter, z.B.
 - das Verkehrssystem
 - das Nervensystem
 - das Sonnensystem
 - oder bestehend aus **Konzepten** und darauf vorhandenen Strukturen, z.B. deren Aufbau aus Teileinheiten oder Beziehungen untereinander.
 - vgl. die Taxonomie der Lebewesen aus der Biologie (Reich, Stamm, Klasse, Ordnung, Familie, Gattung, Spezies)
 - ein Betriebssystem
 - oder einer **Mischung aus realen Gegenständen und Konzepten**, sowie Beziehungen darunter,
 - z.B. das Gesundheitssystem (Beziehungen zw. Versicherungsnehmern, Versicherungsgesellschaften, Ärzten, Krankenhäusern, Verträgen, Konventionen, Pharmaindustrie, usw.)

Was ist Software?

■ System

- Systeme sind bei dieser Betrachtungsweise aus Teilen (**Systemkomponenten** oder **Subsystemen**, gegenständlich oder konzeptionell) zusammengesetzt, die untereinander in verschiedenen **Beziehungen** stehen und wechselwirken können.
- Die Systemkomponenten bilden eine aufgaben-, sinn- oder zweckgebundene **Einheit**.
- Ein System hat eine **Grenze**, die bestimmt, was zum System gehört und was nicht, sowie meist eine **Schnittstelle**, über die das System mit dem Rest der Welt interagiert.
- Systemkomponenten, die nicht weiter zerlegbar sind oder zerlegt werden sollen, werden als **Systemelemente** bezeichnet.

Was ist Software?

- Softwaresystem

- System, dessen Systemkomponenten und Systemelemente aus Software bestehen

Was ist Software?

■ Systemsoftware

- ... auch Basissoftware genannt
- Für eine **spezielle Hardware** oder eine Hardwarefamilie entwickelt, um den Betrieb und die Wartung dieser Hardware zu ermöglichen und zu erleichtern
- Betriebssystem, Compiler, Datenbanksysteme, **Kommunikationsprogramme** und spezielle **Dienstprogramme** (z.B. zur SW-Installation, Benutzerverwaltung, Datensicherung)
- Orientiert sich grundsätzlich an den Eigenschaften der Hardware, für die sie geschaffen wurde und ergänzt normalerweise die funktionalen Fähigkeiten der Hardware

Was ist Software?

■ Anwendungssoftware

- *application software*, Applikationssoftware, „App“
- Löst die Aufgaben des Anwenders mit Hilfe von Rechenanlagen
- **Setzt** in der Regel **auf der Systemsoftware** der verwendeten Hardware auf und benutzt sie zur Erfüllung der eigenen Aufgaben

■ Computersystem bzw. Datenverarbeitungs(DV)-System

- Anwendungssoftware +
- Systemsoftware +
- Hardware

Was ist Software?

■ System- vs. Softwareentwicklung

■ Softwareentwicklung

- Ausschließliche Entwicklung von Software

■ Systementwicklung

- Entwicklung eines Systems, das aus Hardware- und Softwarekomponenten besteht
- Bei der Entwicklung müssen zusätzliche Randbedingungen berücksichtigt werden

Charakteristiken von Software

- Software ist ein **immaterielles Produkt**
- Software unterliegt **keinem Verschleiß**, und man benötigt keine Ersatzteile
- Software wird nicht durch physikalische Gesetze begrenzt
- Software ist **nicht** leichter zu ändern als ein physisches Produkt vergleichbarer Komplexität
- Software ist einfacher und schneller zu verteilen und installieren als ein physisches Produkt (bei SW ist nach der Entwicklung keine Produktion erforderlich, sondern nur eine Kopieroperation oder Datenübertragung)
- Software scheint zu altern (Umgebung ändert sich)
- Software ist schwer zu „vermessen“

Änderungen in der Software in den letzten Dekaden

- Zunehmende Bedeutung – Software wird zur **Wertsteigerungstechnik** in vielen Bereichen
- Wachsende **Komplexität**
- Wachsender Anteil an Software in **mobilen Geräten**
- **Vernetzung und geographische Verteilung**
- Zunehmende **Qualitätsanforderungen**
- Nachfragestau und Engpasstechnik
- Zunehmende Standardsoftware
- Zunehmend „Altlasten“
- Zunehmend „Außer-Haus“-Entwicklung

Bedeutung von Software?

■ Zunehmende Bedeutung

- „In zunehmendem Maße entwickelt sich Software zu einem eigenständigen Wirtschaftsgut und spielt eine entscheidende Rolle in der Gesellschaft. Software ist Bestandteil der meisten hochwertigen technischen Produkte und Dienstleistungen geworden. In einigen Bereichen, wie Banken und Versicherungen, werden nahezu alle Dienstleistungen durch Softwareeinsatz realisiert“ /BMFT 94, S. 3/

Bedeutung von Software?

■ Zunehmende Bedeutung (Forts.)

- Der relative Wertanteil der Software an den Gesamtkosten eines Systems ist in den letzten zwei Jahrzehnten ständig gestiegen.
- ... dass schon jetzt mehr als die Hälfte der Wertschöpfung von Siemens auf Softwareleistungen entfällt. Diese Entwicklung geht weiter.
Heinrich von Pierer, 1992 bis 2005 Vorstandsvorsitzender der Siemens AG

Bedeutung von Software

■ Bedeutung der Softwareindustrie

- „In exportorientierten Branchen der deutschen Wirtschaft übersteigt der Softwareanteil an der Wertschöpfung der Produkte häufig die 50%-Marke. Bei Anlagen der digitalen Vermittlungstechnik entfallen bis zu 80% der Entwicklungskosten auf Software“
/BMFT 94, S. 3/
- Beispiel: Ein Oberklasse-Fahrzeug enthält 50 bis 70 Mikroprozessoren und drei bis vier Kommunikationsbusse

Bedeutung von Software

- Nennen Sie Beispiele, wo Sie von Software in Ihrem täglichen Leben abhängen



Softwarequalität

- Zunehmende Qualitätsanforderungen
 - Für 50% der Ausfälle im industriellen Sektor sind Softwarefehler verantwortlich
 - Nach Cusumano haben sich die gefundenen Defekte in jeweils 1000 Zeilen Quellcode folgendermaßen entwickelt:
 - 1977: durchschnittlich 7 – 20 Defekte
 - 1994: durchschnittlich 0,2 – 0,05 Defekte
 - In 17 Jahren konnte die Defektdichte also um ungefähr das 100fache gesenkt werden

Softwarequalität

- 0,1%-Defektniveau bedeutet:
 - pro Jahr:
 - 20.000 fehlerhafte Medikamente
 - 300 versagende Herzschrittmacher
 - pro Woche:
 - 500 Fehler bei medizinischen Operationen

Softwarequalität

- 0,1%-Defektniveau bedeutet: (Forts.)
 - pro Tag:
 - 16.000 verlorene Briefe in der Post
 - 18 Flugzeugabstürze
 - pro Stunde:
 - 22.000 Schecks nicht korrekt gebucht
- Auch in Zukunft:
 - Massive **Anstrengungen** zur Qualitätssicherung **notwendig**

Softwarequalitätsprobleme

■ Beispiel 1: Gepäcktransportsystem des Flughafens Denver

- Gigantischer Flughafen:
 - 140 km², doppelte Fläche von Manhattan,
 - 10 Mal so breit wie Heathrow.
 - 3 Flugzeuge können gleichzeitig landen, in schlechtem Wetter.

- Gepäcktransportsystem:

34 km	lang
4000	automatische Fahrzeuge
100	Steuerungsrechner
5000	optische Sensoren
400	Funkgeräte
56	Strichcode-Leser

Softwarequalitätsprobleme



Gepäcktransportsystem des Flughafen Denver



Softwarequalitätsprobleme (nicht nur)

- Flughafen Denver
 - Geplante Eröffnung: 31 Okt. 1993
 - Automatisches Gepäcktransportsystem verzögert wegen Softwareproblemen (Koffer zerdrückt und zerfetzt)
 - Erste Verzögerung bis Dez. 1993, dann März 1994.
 - Juni 1994: System noch immer nicht betriebsbereit, keine Schätzung für Zeitpunkt der Inbetriebnahme
 - Inbetriebnahme: Feb. 1995 (16 Monate Verspätung)
- Kosten: 1.1 Million \$ / Tag (Zinsen plus Betrieb des leeren Flughafens)
- Insgesamt: 560 Millionen \$ Verlust.
- (siehe Artikel in Scientific American, Sept. 1994)
- Das Gepäcksystem wurde 2005 verschrottet und durch ein manuelles mit Schleppern und Anhängern ersetzt. Es hatte nie ordentlich funktioniert.

Softwarequalitätsprobleme

■ Beispiel für zu Hause:



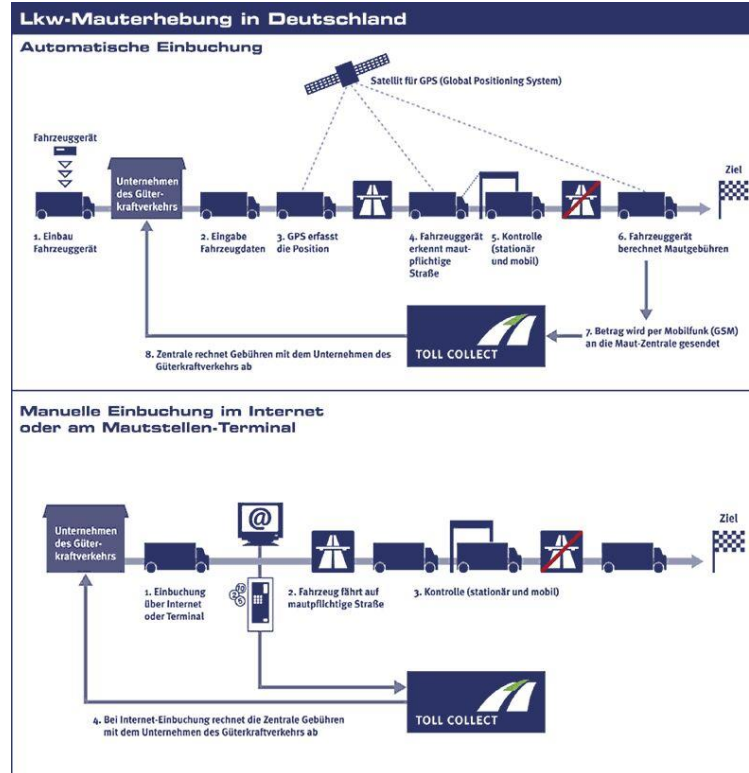
TOLL COLLECT
service on the road

Softwarequalitätsprobleme

■ Beispiel 2: Deutschlands TollCollect

- TollCollect ist ein System zur Mauterhebung für Schwerlasten auf deutschen Autobahnen.
- Neu: Mauterhebung ohne Anhalten oder Verlangsamung der Lasten; es gibt keine Mautstellen!
- Statt dessen hat jeder Lasten einen Bordrechner mit GPS-Sensor, der die gefahrenen Strecken erfasst, die fällige Maut berechnet und diese Information periodisch via Mobilfunk (SMS!) an eine zentrale Abrechnungsstelle sendet. Die Abrechnungsstelle erstellt Rechnungen und belastet Kredit- und Flottenkarten.
- Lasten ohne Bordrechner können ihre Maut an Web-Terminalen an Raststätten vorausbezahlen.
- Überwachungsbrücken an Autobahnen prüfen korrekte Mautzahlung.

Softwarequalitätsprobleme

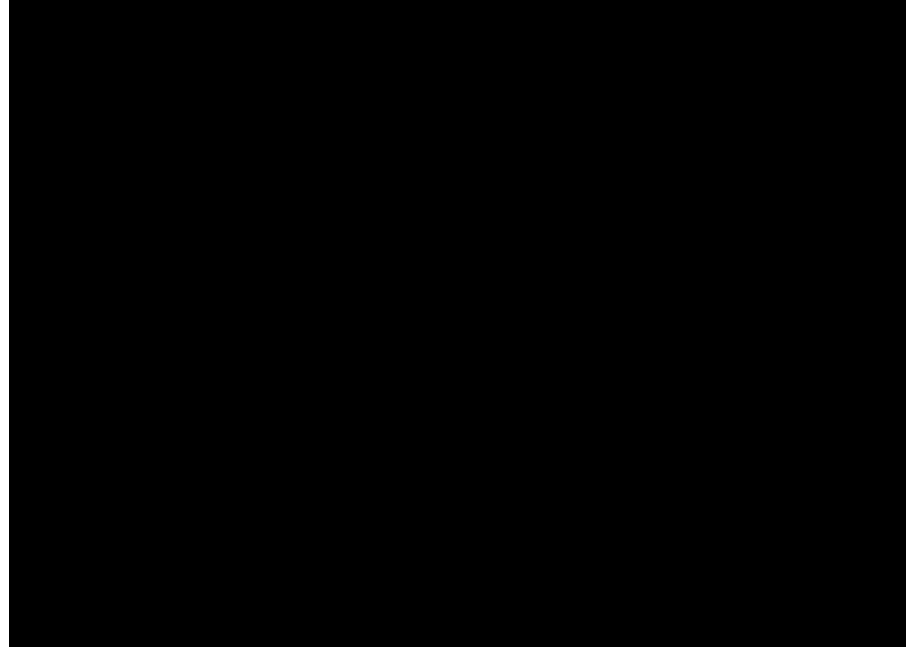


Softwarequalitätsprobleme

■ TollCollect:

- Geplante Inbetriebnahme: 31. August 2003
- Software Probleme verzögerten den Start (Bordrechner stürzten ab, manche konnten nicht mehr gestartet werden; Zentralsysteme überlastet mit Anfragen nach neuen Autobahnstrecken und aktuellen Tarifen)
- Alternative: Eurovignette war abgeschafft worden.
- Reduziertes System ging im Januar 2005 in Betrieb (ohne Aktualisierung von Strecken und Tarifen per Funk)
- Gesamtsystem ging am 1. Januar 2006 in Betrieb
- Gesamtverlust: 5600 Millionen Euro plus Zinsen.

Beispiel 3:
Die Ariane 5
Rakete
explodierte 40 s
nach dem Start.
(1996, \$500
Mio. Schaden).



Details: <http://www.around.com/ariane.html>

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

Fallstudie des Absturzes

- Eine Überlauf ereignete sich bei der **Konvertierung** einer 64-Bit-Gleitkommazahl in eine 16-Bit-Ganzzahl in einem Programm genannt „**Inertial Reference System**“.
- Dieser Überlauf wurde **nicht abgefangen**, daher hielt die ganze **Lagesteuerung** an, was die Rakete in eine unzulässige **Schräglage** brachte, worauf sie sich selbst sprengte.
- Das Programm, das den Überlauf verursachte, war während des Fluges **gar nicht benötigt** (nur bis -9s Startzeit).
- Es lief aber 50 s lang nach dem Start weiter, um bei einem Abbruch des Starts eine Re-Initialisierung von mehreren Stunden zu vermeiden.
- Es war für die Ariane 4 Rakete entwickelt worden, bei der dieser Überlauf nicht auftreten konnte.
- Der Defekt war also ein **Wiederverwendungsfehler**, verursacht durch **fehlende Spezifikation** der Bedingungen, unter denen die Software richtig arbeitet.

Genaue Ursache für den Absturz

- 37 Sekunden nach Zündung der Ariane 5 lieferte der Sensor für die horizontale Geschwindigkeit in einer Flughöhe von 3.700 Metern den Wert 32.768,0
- Es trat ein Überlauf bei der Umwandlung des Wertes 32.768,0 in eine 16-Bit-Ganzzahl auf.
- Die maximale Zahl bei 16 Bit einschl. Vorzeichen ist $2^{15} - 1 = 32.767$; beachte dass 16. Bit als Vorzeichen dient.
- Dieser Fehler wurde jedoch nicht abgefangen.
- Die Software schaltete daher auf den Ersatzrechner um, welcher sich auf Grund des gleichen Problems 72ms zuvor bereits abgeschaltet hatte.
- Die deshalb an den Hauptrechner geschickten Diagnose-Daten wurden als Flugbahndaten interpretiert.
- Daraus resultierten unsinnige Steuerbefehle für die Triebwerke, was zu einer extremen Lage der Rakete führte.
- Da die Ariane 5 auseinanderzubrechen drohte, sprengte sie sich selbst (39 Sekunden nach der Zündung).

Ariane 5 – Navigationssystem (Ada)

```

...
declare
  vertical_veloc_sensor: float;
  horizontal_veloc_sensor: float;
  vertical_veloc_bias: integer;
  horizontal_veloc_bias: integer;
...
begin
  declare
    pragma suppress(numeric_error, horizontal_veloc_bias);
  begin
    sensor_get(vertical_veloc_sensor);
    sensor_get(horizontal_veloc_sensor);
    vertical_veloc_bias := integer(vertical_veloc_sensor);
    horizontal_veloc_bias := integer(horizontal_veloc_sensor);
    ...
  exception
    when numeric_error => calculate_vertical_veloc();
    when others => use_irs1();
  end;
end irs2;

```

16-Bit Ganzzahl mit
Vorzeichen.

Gleitkommawert:
32768.0

Schalte auf
Ersatzrechner um.

Überlauf bei Umwandlung,
welcher nicht abgefangen
wurde.

Warum ist Software so schwer zu entwickeln?

■ Zunehmend **Außer-Haus-Entwicklung**

■ Trend:

- Software nicht selbst entwickeln
- Sondern: Bei Softwarehäusern (auch im Ausland) in Auftrag geben
- Softwareentwicklung und zugehörige Dienstleistungen werden zunehmend in Niedriglohnländer verlagert
- Durch die zunehmende Produktintegration von Software (eingebettete Systeme) wird der Prozentsatz intern erstellter Software nicht drastisch zurückgehen
- Typische Geschäftsanwendungen müssen heute in ca. 60 Arbeitstagen erstellt werden

Warum ist Software so schwer zu entwickeln?

■ 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 Softwarekosten

Probleme bei der Erstellung marktreifer Software

- Was sind die Probleme bei der Erstellung marktreifer Software?
- Nennen Sie Beispiele!



Probleme bei der Erstellung marktreifer Software

- Anforderungen:
 - Funktionstreue
 - Qualitätstreue
 - Termintreue
 - Kostentreue

Probleme bei der Erstellung marktreifer Software

- Spezifische Randbedingungen:
 - Während der Entwicklung ändern sich...
 - die **Produktanforderungen**
 - neue oder geänderte Funktionen
 - geänderte Benutzungsoberfläche usw.
 - neue Schnittstellen (z.B. Webschnittstelle, Sprachsteuerung, App)

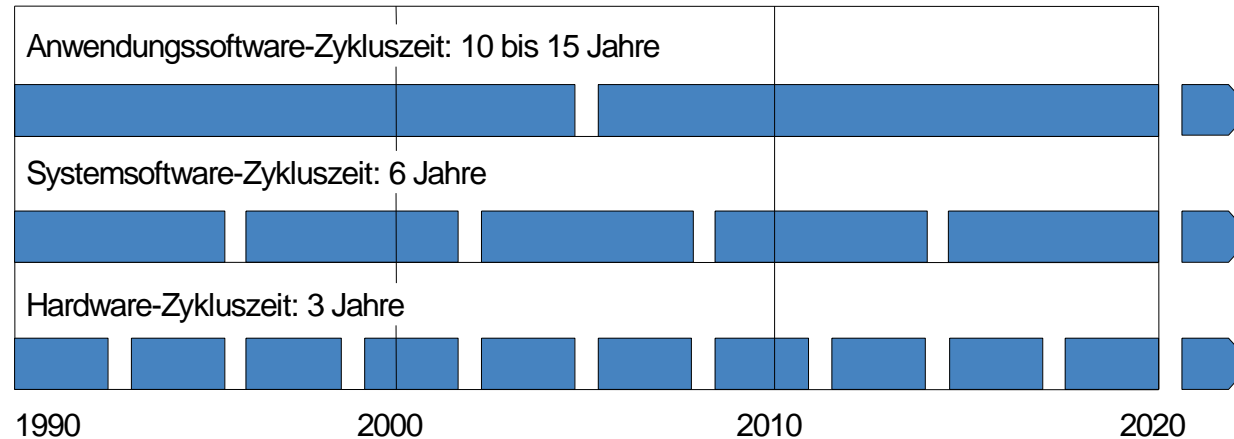
Probleme bei der Erstellung marktreifer Software

- Spezifische Randbedingungen: (Forts.)
 - Während der Entwicklung ändern sich...
 - Die **Komponenten** von **Hardware- und Systemsoftware**
 - neuer Prozessor
 - andere Peripherie
 - anderes Fabrikat
 - technische Innovationen.
 - die Softwaremethoden und -werkzeuge, mit denen das Produkt erstellt wird (Produktionsmittel)

Probleme bei der Erstellung marktreifer Software

- Spezifische Randbedingungen: (Forts.)
 - Hohe **Portabilitätsanforderungen** sind einzuhalten...
 - da Softwareprodukte eine meist wesentlich längere Lebensdauer besitzen wie die zugrunde liegende Hardware
 - oft auf mehreren Plattformen (HW & Systemsoftware) eingesetzt werden

Probleme bei der Erstellung marktreifer Software



Probleme bei der Erstellung marktreifer Software

- Entwickelt man heute Anwendungs-SW, dann bedeutet dies, dass...
 - während der Lebenszeit der Anwendungssoftware **mindestens einmal** die zugrunde liegende System-SW und **mindestens zweimal** die HW ausgetauscht wird
 - die Zielsysteme, für die Software entwickelt wird, zur Entwicklungszeit evtl. noch nicht vorhanden sind
 - weltweit vertriebene Softwareprodukte länderspezifische Varianten erlauben müssen (muttersprachliche Benutzungsoberflächen)

Softwaretechnik (engl. **software engineering**) ist die technologische und organisatorische Disziplin zur systematischen Entwicklung und Pflege von Softwaresystemen, die spezifizierte funktionale und nicht-funktionale Attribute erfüllen.

Hinweis: Definition auswendig lernen!

Was ist Softwaretechnik?

- Organisatorische Aspekte:
 - **Planung:** Wie, wo, wann ist was von wem zu tun; was sind die Kosten?
 - **Personalfindung,** Stellenbesetzung
 - **Arbeitsorganisation:** Regeln zur Zusammenarbeit (Prozesse, Aufgaben, Zuständigkeiten, Verantwortung, Berichtspflicht)
 - **Leitung:** Motivation und Kommunikation von Zielen
 - **Kontrolle:** prüfen, ob Arbeit nach Plan fortschreitet, Anforderung erfüllt werden
- Zu Arbeitsorganisation noch mehr gegen Ende der Vorlesung:
 - Wasserfall, iterative Modelle, agile Modelle (XP, Scrum, SyncStab)

Was ist Softwaretechnik?

- **Systematisch:** Es wird nicht herumprobiert oder geraten. Methoden und Werkzeuge werden zielgerichtet und dem Stand der Kunst entsprechend eingesetzt
- **Entwicklung und Pflege** von Softwaresystemen: Bei erfolgreicher Software ist nach der Entwicklung Pflege erforderlich (Korrekturen, Anpassungen, Erweiterungen). Pflege kann bis zu 2/3 der Gesamtkosten ausmachen

Was ist Softwaretechnik?

- **Funktionale Attribute** spezifizieren die Funktionen der Software
- **Nicht-funktionale Attribute** (auch Qualitätsattribute) spezifizieren, wie gut die Software ihre Funktionen erfüllt (wie zuverlässig, wie schnell, wie benutzerfreundlich, wie sicher) aber auch innere Qualitäten wie Änderbarkeit, Dokumentationsgrad.

Was ist Softwaretechnik?

- Gesellschaftlicher Auftrag an die Softwaretechnik: Software erstellen
 - in ausreichender Menge
 - von zufrieden stellender Qualität
 - termingerecht
 - zu wettbewerbsfähigen Kosten

- Nicht erforderlich: umsonst, Perfektion, sofort

Woher weiß man, wie das geht?

Softwareforschung (engl. **Software Engineering Research, Software Research**) ist die Bereitstellung und Bewertung von *Methoden, Verfahren* und *Werkzeugen* für die Softwaretechnik.

Was ist Softwaretechnik?

■ Methoden

- **Planmäßig** angewandte, begründete **Vorgehensweisen** zur Erreichung von festgelegten **Zielen**
 - Enthalten den Weg zu etwas hin
 - Machen Prinzipien anwendbar
 - Planmäßig: Beim Einsatz einer methodischen Vorgehensweise wird nicht herumprobiert
 - Geben an, welche Konzepte wie und wann verwendet werden, um festgelegte Ziele zu erreichen
 - Kann durch eine oder mehrere Verfahren unterstützt werden

Was ist Softwaretechnik?

■ Verfahren

- sind ausführbare Vorschriften oder Anweisungen zum **gezielten Einsatz** von **Methoden**
- Beschreiben konkrete **Wege zur Lösung** bestimmter Probleme oder Problemklassen
- Ein Verfahren ist im Allg. wegen seiner Beschränkung stärker einsatzbezogen als eine Methode
- enthalten meist **formale Vorschriften** und bilden den Inhalt von Standards

Was ist Softwaretechnik?

- Beispiel: OOA (Objektorientierte Analyse)
- Methoden (Vorgehensweisen):
 - Orientierung an den Daten des Systems (statisches Modell)
 - Orientierung an der Funktionalität des Systems (dynamisches Modell)
- Verfahren:
 - Beschreiben Schritte zum Erstellen des statischen (oder dynamischen) Modells
 - Beispielsweise: Klassen, Assoziationen, Attribute, usw. identifizieren

Was ist Softwaretechnik?

■ Werkzeuge (engl. *tools*)

- Dienen der **automatisierten Unterstützung** von **Methoden**, Verfahren, Konzepten oder bieten Notationen/Sprachen an
- Beispiel
 - Verfahren: Festgelegte Kontrollstrukturen in einer Programmiersprache
 - Werkzeuge: Überprüfung, ob keine *label*-Vereinbarung und kein *goto* verwendet wurde

Apple SSL-Fehler (Secure Socket Layer for https) 2014

```
static OSStatus
SSLVerifySignedServerKeyExchange(SSLContext *ctx, bool isRsa, SSLBuffer signedParams,
                                uint8_t *signature, UInt16 signatureLen)
{
    OSStatus      err;
    SSLBuffer      hashOut, hashCtx, clientRandom, serverRandom;
    uint8_t        hashes[SSL_SHA1_DIGEST_LEN + SSL_MD5_DIGEST_LEN];

    ■ ■ ■ ■ ■

    if ((err = ReadyHash(&SSLHashSHA1, &hashCtx)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &clientRandom)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &signature)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
        goto fail;

    err = sslRawVerify(ctx,
                      ctx->peerPubKey,
                      dataToSign,
                      dataToSignLen,
                      signature,
                      signatureLen);
    /* plaintext */
    /* plaintext length */

    if(err) {
        sslErrorLog("SSLDecodeSignedServerKeyExchange: sslRawVerify "
                   "returned %d\n", (int)err);
        goto fail;
    }

fail:
    SSLFreeBuffer(&signedHashes);
    SSLFreeBuffer(&hashCtx);
    return err;
}
```

Erkennen Sie den Fehler?
Eine automat. Überprüfung
hätte ihn gefunden.

Was ist Softwaretechnik?

■ Werkzeuge (Forts.)

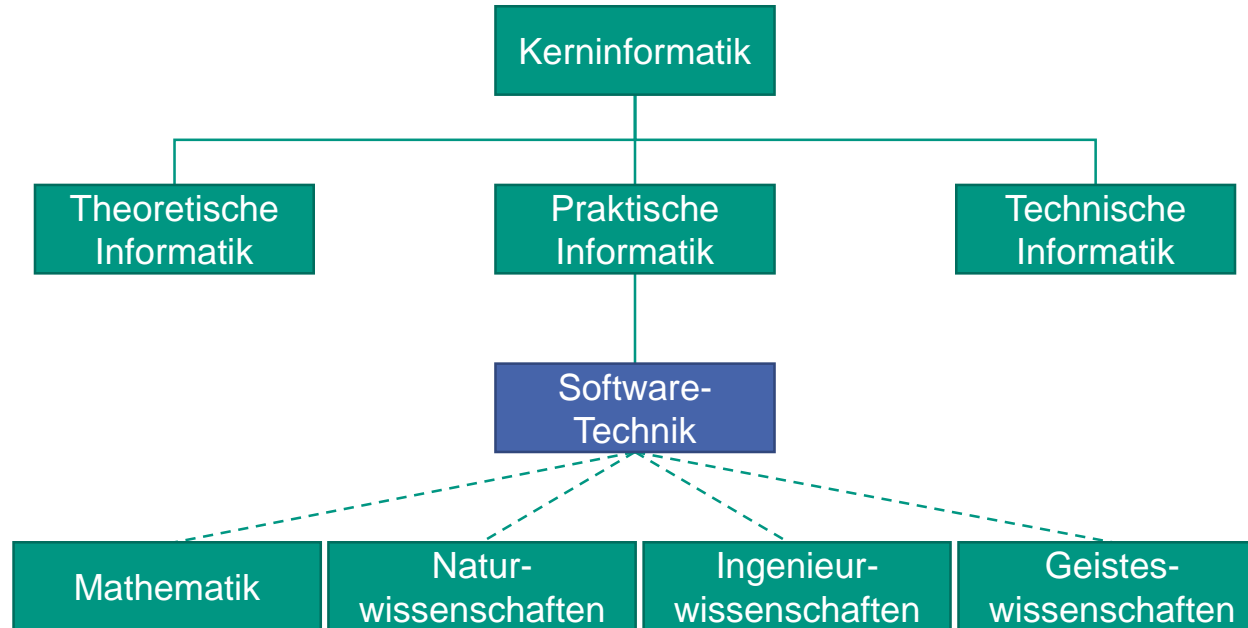
- Durch Werkzeuge kann die Einhaltung von Methoden, Verfahren, Standards, und Notationen erzwungen und die Produktivität durch Automatisierung erhöht werden
- Werkzeuge automatisieren Tätigkeiten, die...
 - hohe Präzision erfordern
 - oft wiederholt werden müssen
 - eine Überprüfung erfordern

Was ist Softwaretechnik?

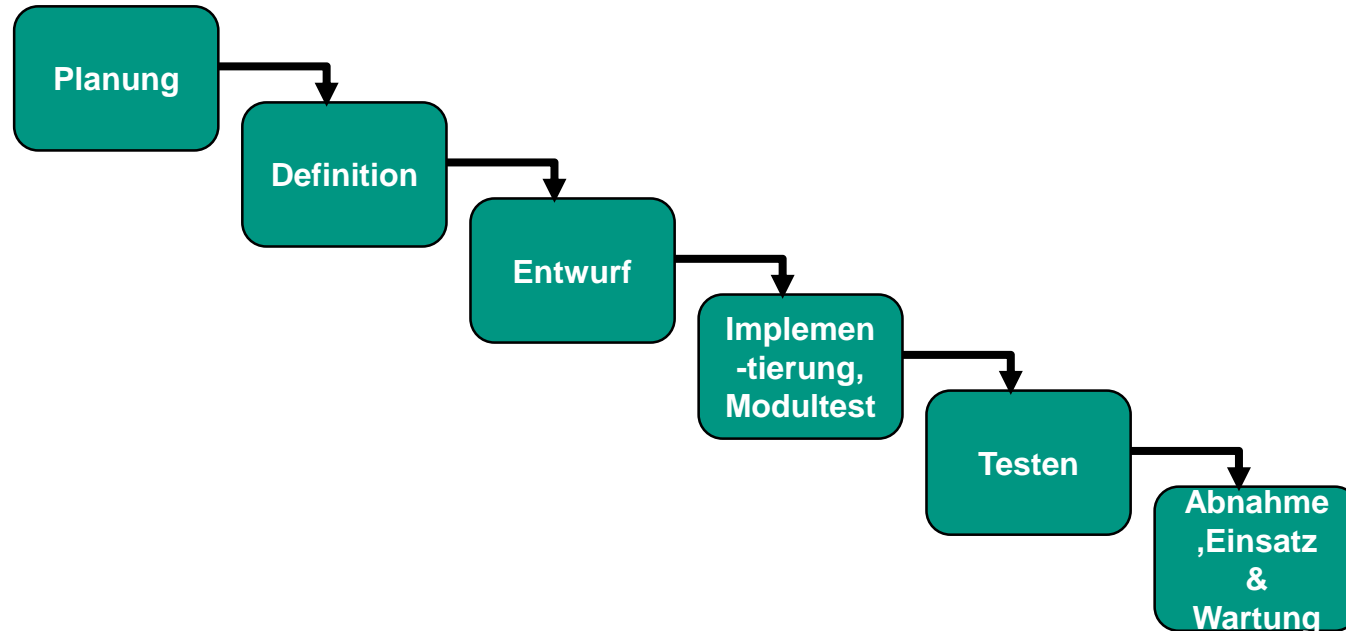
■ Werkzeuge (Forts.)

- Um Software zu entwickeln, ist der Einsatz von Werkzeugen **notwendig**, aber **nicht hinreichend**
- CASE
 - *Computer Aided Software Engineering*
 - Softwareentwicklung erfolgt mit Hilfe von Softwarewerkzeugen
- Beispiel: Werkzeuge zum Gestalten von graphischen Benutzeroberflächen

Was ist Softwaretechnik?

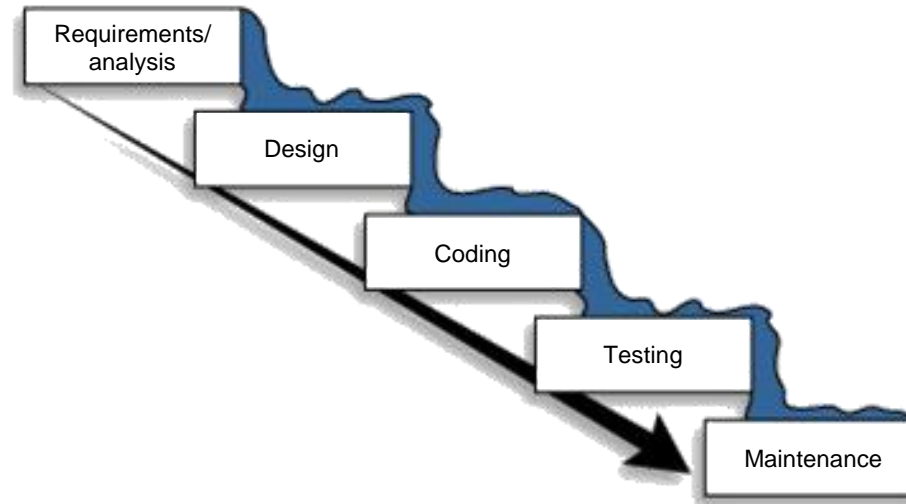


Überblick über den Aufbau der Vorlesung



Die Idee hinter dem Wasserfallmodell

The classic waterfall development model



Wasserfall

