


Grundlagen der Softwaretechnik (SWT)

Vorlesung im Wintersemester 2015/2016
Leibniz Universität Hannover



Prof. Dr. Kurt Schneider
Fachgebiet Software Engineering
Kurt.Schneider@Inf.Uni-Hannover.de



Grundlagen der Softwaretechnik


1. Wieso Software Engineering?
2. Vom Einzelkämpfer zum Großprojekt

Systematische Softwareentwicklung

3. Anforderungen und Test: *Die Basis des Projekts*
4. Entwurf: *Strukturen und nicht-funktionale Eigenschaften*
5. Entwürfe notieren mit UML: *Modelle im Software Engineering*
6. Design Patterns: *Entwurfserfahrungen nutzen*
7. Management: *Technik und Projekt*

Ausblick auf „Software-Qualität“ im Sommersemester:
Agile Methoden, Testen für Fortgeschrittene, Prüfen und konstruktive Qualitätssicherung


Kurt Schneider Leibniz Universität Hannover SWT 2015/16 3



Anspruch der Veranstaltung

- Lernziele von SWT (Vorlesung und Übungen)
 - Sie kennen die Schlagworte des Software Engineering
 - Sie wissen, wie wichtige Verfahren im Prinzip funktionieren
 - Sie haben einige selbst ausprobiert
 - Sie wissen, wo sie genauer nachlesen können
 - Sie können vernünftig in SW-Projekten mitarbeiten
 - Sie achten beim Programmieren auf Verständlichkeit
 - Sie wissen, wie wichtig Testen ist und machen es auch
 - Sie setzen fortgeschrittene Techniken (Design Patterns) ein
 - Sie entscheiden auf Basis eines begründeten Urteils
- Sie lernen dagegen *nicht*
 - Grundlagen des Programmierens: Java (Basics) vorausgesetzt
 - Irgendein kommerzielles Werkzeug zu bedienen


Kurt Schneider Leibniz Universität Hannover SWT 2015/16 4



Organisatorisches

- Aktuelle Informationen, Folien etc.:
 - Wir verwenden StudIP zum Verteilen der Folien und für Nachrichten
 - Melden Sie sich aber bitte auch hier an: www.se.uni-hannover.de
 - Dort werden Bonuspunkte verwaltet. Das kann StudIP nicht.
- Klausur
 - Klausur SWT (direkt danach folgt SWQ):
Fr. 1.4.2016, 11:30h-12:30h, Räume E214+Audimax (Hauptgebäude).
Lesen Sie kurz vorher www.se.uni-hannover.de
- Übungen: Olga.Liskin@inf.uni-hannover.de
 - Beachten Sie die Übungszeiten!
- Irrtum und Änderungen vorbehalten!

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 5



Vorlesung – Übung – Praktikum

Vorlesung

- Konzepte vorstellen
- Erklären
- kurz zeigen

Übung

- Wichtiges noch mal zeigen
- Selbst anwenden/erklären
- Erfahrungen und Fragen diskutieren
- Übrigens: Korrelation Übung-Note


Programmier Praktikum (ProPra)

- Programmieren üben
- Scheu verlieren
- In größerem Zusammenhang anwenden

Klausur

Berufspraxis

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 6




Wo Sie die angegebene Literatur finden

- Bibliothek gegenüber dem Hauptgebäude
- „Systemstelle Software Engineering“ (L inf 430)
- Im Lesesaal oder zu bestellen
- Spezialliteratur haben wir am SE
 - Fragen Sie uns bei Interesse

Manche Bücher sind sehr gut zu lesen. Blättern Sie doch mal durch!

In Anlehnung an Cockburn, A.: Writing Effective Use Cases. Addison-Wesley, 2001

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 7



Ein paar Tipps, aus Erfahrung

- Das Wesen einer Vorlesung
 - Sie sind freie Menschen
- Laptop, Block, oder Gedächtnis?
 - Tipps zur Arbeitstechnik
- Auch hier achten wir auf freundlichen, professionellen Stil
 - Kommen Sie pünktlich
 - Seien Sie leise, stören Sie Ihre Kollegen nicht
 - Gehen Sie respektvoll miteinander um
 - Spielen Sie mit Handy und Laptop: **aber bitte nur draußen!**
 - Wie man hineinruft ...


Wir wollen, dass Sie Erfolg haben
und Ihnen die Sache Spaß macht!

Kurt Schneider
Leibniz Universität Hannover
SWT 2015/16 8

Softwaretechnik

Kapitel 1

1. Wieso Software Engineering?



1. Wieso Software Engineering?
2. Vom Einzelkämpfer zum Großprojekt

Systematische Softwareentwicklung


3. Anforderungen und Test: *Die Basis des Projekts*
4. Entwurf: *Strukturen und nicht-funktionale Eigenschaften*
5. Entwürfe notieren mit UML: *Modelle im SE*
6. Design Patterns: *Entwurfserfahrungen nutzen*
7. Management: *Technik und Projektmanagement*

Leibniz Universität Hannover
SWT 2015/16 9

Softwaretechnik

Inhalt von Kapitel 1

1. Wieso Software Engineering?



- Anliegen und Ziele
- Herausforderungen an Software-Profis
- Grundbegriffe
- Geschichte und „Vorbild Ingenieur“

Leibniz Universität Hannover
SWT 2015/16 10





Software ist überall


- Bankomat
- Animierter Film
- Auto
- Zündschlüssel
- Spielzeug
- Gehaltsabrechnung
- PC-Spiel
- Luftraumkontrolle
- Medizin
- ...

Kurt Schneider
Leibniz Universität Hannover
SWT 2015/16 12



Mit SW ist jeder ständig konfrontiert

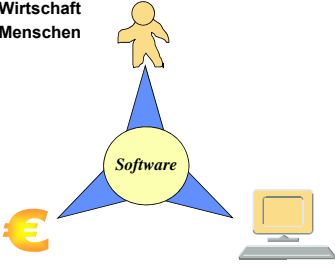
- jemand muss eine Idee für SW haben
- jemand muss sie entwickeln
- jemand muss sie kaufen
- jemand muss sie installieren
- jemand muss sie verwenden (wollen)
- jemand braucht die Resultate



Kurt Schneider
Leibniz Universität Hannover
SWT 2015/16 13

Fazit: Einfluss von Software

- SW beeinflusst Technik
- SW beeinflusst Wirtschaft
- SW beeinflusst Menschen



Kurt Schneider Leibniz Universität Hannover SWT 2015/16 14

Software ...

Ist der faszinierende „Werkstoff unserer Zeit“!

**Aber es kommt darauf an,
was man daraus macht**

... und wie man es macht!

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 15

Wenn es dagegen nicht funktioniert

- Katastrophenbeispiele
 - Flugzeug bremst nicht
 - Marssonde geht verloren
 - Börsenprogramme schaukeln sich auf
 - Fahrkartenautomat macht einen verrückt
 - Spiel ist nicht zu kapieren, Geburtstagskind heult
- Was hätten Sie besser gemacht?
- Sie werden Gelegenheit bekommen, das zu beweisen:
 - Auto
 - Banken, Börsen
 - Kernkraftwerke
 - Spielzeughersteller
 - Admin. Software
 - ...

Hier oder in vielen anderen Bereichen
... schon in Übungen und Softwareprojekt!

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 16

Software-Probleme in den Medien

Steuererklärung
Software-Problem lähmt Finanzämter

Samstag, 27.02.2010, 16:40

Google-Anzeigen Finanz Formular
Einfache und sichere Buchhaltung für kleine Unternehmen von Sage www.sage-small-business.de

An das Finanzamt
(soweit schon erhalten)

Tausende Bürger in allen Bundesländern müssen einem Zeitungsbericht zufolge bei der Bearbeitung ihrer Einkommensteuererklärung mit Verzögerungen rechnen.

Von den Problemen betroffen sollen Bürger sein, die eine Steuererklärung mit der Anlage KAP (Einkünfte aus Kapitalvermögen) beim Finanzamt einreichen, berichtet das Bielefelder „Westfalen-Blatt“ unter Berufung auf das nordrhein-westfälische Finanzministerium. Die Software zur Bearbeitung der Anlage funktioniert nicht, die



Elektronische Gesundheitskarte



Toll-Collect

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 17

Softwareentwicklung „richtig machen“ ein kleiner Unterschied

Annahme: Sie wollen nicht Hobbyprogrammierer werden, sondern Softwareprofi – dazu gehört die richtige Einstellung

Nach dieser Vorlesung sollen Sie

- Fähigkeiten
- Fertigkeiten und
- Überzeugungen gewonnen haben,

die die **Wahrscheinlichkeit für erfolgreiche Projekte** erhöhen.

„Programmieren ist nicht schwer, professionell Entwickeln sehr“

- Technische Seite: Programmstruktur und Methoden
- Ökonomische Seite: Kosten und Nutzen
- Menschliche/psychologische Seite: Kunden und Mitarbeiter

Nehmen Sie die Softwaretechnik **ernst**, dann macht sie Spaß

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 18

Kritik ist einfach

Unsere Frage muss sein: was können wir (als Informatiker) dagegen tun?

- Bluescreen
- Googles Datenerhebung
- Zu komplizierte Oberflächen
- Langsame Verarbeitung
- Unlogische/doppelte Abfragen
- Störende Releasewechsel
- Sicherheitsbedenken

- Anforderungen klären
- Benutzer verstehen
- Neueste Technik einsetzen
- Seriös planen und verfolgen
- Qualität und Risiken prüfen

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 19

Software Engineering

Herausforderung an Profis 1

Software soll tun, was Kunden erwarten

- Aber woher wissen wir, was Kunden erwarten?
 - Anforderungen erfassen und umsetzen
 - Irrglauben ablegen, wir wüssten selbst am besten, was richtig ist
- Und wer genau sind diese „Kunden“?
 - Initiator des Auftrags, Käufer, Anwender, Marketing
 - Nicht nur den Erstbesten befragen
- Aber wenn die es nicht sagen?
 - dann können sie es vielleicht einfach nicht!
 - Wir müssen helfen – dafür gibt es Techniken (Requirements Eng.)

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 20

Software Engineering

Herausforderung an Profis 2

Software soll leicht zu bedienen sein

- Aber für wen?
 - Verschiedene Leute haben unterschiedliche Bedürfnisse
 - Usability Engineering: Benutzerprofil erstellen
- Und in welchem Zusammenhang?
 - Eigentlich will kaum einer ein Programm bedienen – sondern eine Aufgabe erledigen
 - Usability Engineering: Aufgabenanalyse durchführen
- Aber sind Oberflächen nicht „Geschmackssache“?
 - Irrtum: darum geht es gar nicht! Man kann objektiv sagen:
 - manche Farbkombinationen sind physiologisch ungünstig
 - Man merkt sich nur 5+/-2 oder 3*3 Einheiten
 - und so weiter: das zu bedenken hilft schon sehr
 - Geschmacksfragen von obj. Bedienbarkeit trennen

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 21

Software Engineering

Herausforderung an Profis 3

Software soll im Zeit- und Kostenplan bleiben

- Realistisch planen trotz vieler Unsicherheiten
 - Alle relevanten Aspekte methodisch planen (Projektmgmt.)
 - Systematisch aus Erfahrung lernen
- Neueste Technik einsetzen, dennoch gut schätzen
 - Typische Informatikerfehler vermeiden, Schätztechniken
- Auswirkungen von Zeit- und Kostenüberziehung bedenken
 - Risiken für den Kunden abschätzen
 - Damit eigene Risiken

Zeit
Kosten
Qualität

Software

ACHTUNG!
„Qualität“ umfasst hier Funktionalität und nicht-funktionale Eigenschaften

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 22

Software Engineering

- Hilft, diese Herausforderungen anzugehen
- Bietet systematische Methoden und Techniken
 - Projektmanagement
 - Risikomanagement
 - Requirements Engineering
 - Usability Engineering
 - ... und viele Techniken mehr
- Bezieht sich auf SW-Entwicklung, -Akquisition und -Wartung
- Bietet Konzepte, praktische Tipps für den täglichen Bedarf
... aber auch ungelöste Forschungsfragen

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 23

Software Engineering

SE wird beeinflusst

Ein „sozio-technisches System“

Menschen

Wirtschaft

Technik

Software Engineering

Fazit:

- Das ist kein „softes Bild“: harte Einflüsse!
- SE muss sich in der Praxis dieses Spannungsfelds bewähren
- Software-Profis können sich nicht nur auf einen Aspekt stützen

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 24

Software Engineering

Grundbegriffe

- Software (SW)
 - Computer programs, procedures and possibly associated documentation and data pertaining to the operation of a computer system (IEEE Std 610.12 - 1990)
 - Bemerkungen: „Computer system“ im weiteren Sinn: embedded
- Software Engineering (SE)
 - (1) the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software
 - (2) the study of approaches as in (1) (IEEE Std 610.12 - 1990)
 - offenbar ist also Engineering: „a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of X“
 - Parnas: „SE erst ab 2 Personen und 2 Versionen“
- Softwaretechnik (SWT): hier als Synonym für „Software Engineering“

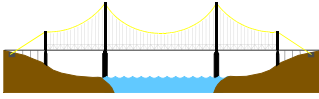
Kurt Schneider Leibniz Universität Hannover SWT 2015/16 25

„Ingenieursprinzipien“

„A systematic, disciplined, quantifiable approach to the development, operation, and maintenance of X“

Konsequenzen

- **Kostendenken**
 - Kein Perfektionismus
- **Qualitätsbewußtsein**
 - „es läuft“ reicht nicht
- **Anwendung von Normen und Regeln**
 - keine Künstler
- **Probleme durch Zerlegung lösen**
 - „divide et impera“
- **Baugruppen und Wiederverwendung**
 - statt „not invented here“



Kurt Schneider Leibniz Universität Hannover SWT 2015/16 26

Wieso „Software Engineering?“

- Hardware wird relativ immer billiger und besser
 - Speicher ist größer, schneller, billiger
 - Rechner auch
 - Netzwerke auch
- Software kann nicht mithalten → „Produktivitätsgap“
 - Denn Software entsteht im wesentlichen „im Hirn“
 - Das wächst nicht so schneller wie Chips und Speicher
- Verzahnung mit Wirtschaft und Gesellschaft
 - Immer mehr Software in allen Bereichen
 - steigende Komplexität
 - wachsende Lebensdauer
 - größere Technologiesprünge

SW=Risiko+Chance

Muss man systematisch angehen, nicht zufällig

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 27

Kurze Geschichte des Software Engineering

- Sehnsucht: ingenieurmäßig auch bei Software vorgehen!
 - **Verklärung des Ingenieurs**
 - Bei Hardware klappt es so gut, und bei Brücken und Häusern ?
 - **Aber Software ist anders**
 - SW ist immateriell → nur Entwicklung, keine Fertigung
 - SW hat unstetiges Verhalten → Testen viel schwieriger
 - universelles Material → für alle möglichen Anwendungen
 - amorphes Material → keine implizite Struktur
 - Entwickler überschätzen sich regelmäßig
 - **Trotzdem möchte man SW genauso gut in den Griff kriegen**
- **F.L. Bauer, NATO Science Conference, Garmisch-Partenkirchen 1968**

„Anwendung von Ingenieursprinzipien für zuverlässige SW, die auf realen Rechnern läuft und mit wirtschaftlichen Mitteln erstellt wird.“

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 28

NATO Science Conference Garmisch-Partenkirchen 1968

<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/N1968/>



Prof. F.L. Bauer



Der Begriff „Software Engineering“ wird geprägt

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 29

Professionelle Softwareentwicklung?



Kurt Schneider Leibniz Universität Hannover SWT 2015/16 30