

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

Dieser kommentierte Foliensatz soll Ihnen helfen, sich besser an die Vorlesung zu erinnern. Auch die Prüfungsvorbereitung kann er unterstützen.

Grundlage der Prüfung ist jedoch ausschließlich die Vorlesung selbst, nicht die Folien oder Kommentare in diesem Skript! Sie können sich im Zweifel nicht auf dieses Dokument berufen.

Denn natürlich kann eine halbe Seite pro Folie nicht all das erfassen, was in der Vorlesung dazu gesagt wurde. Auch die Nachfragen und Diskussionen fehlen hier natürlich.

Wenn Sie diese Kommentierten Folien jedoch nach dem Besuch der Vorlesung zur Auffrischung des Gehörten einsetzen, dann werden Sie sicher von den Kommentaren profitieren.

Viel Spaß damit und viel Erfolg!

Kurt Schneider

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

The cover slide features the logo 'SOFTWARE ENGINEERING' with 'SE' in large letters. The title 'Grundlagen der Softwaretechnik' is at the top. Below it is a list of seven chapters: 1. Wieso Software Engineering?, 2. Vom Einzelkämpfer zum Großprojekt, 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, and 7. Management: Technik und Projekt. A green box at the bottom right contains the text: 'Ausblick auf „Software-Qualität“ im Sommersemester: Agile Methoden, Testen für Fortgeschrittene, Prüfen und konstruktive Qualitätssicherung'. The footer includes the names 'Kurt Schneider', 'Leibniz Universität Hannover', and 'SWT 2015/16 - 3'.

Die Vorlesung besteht aus sieben Kapiteln:

Zunächst wird begründet, wieso man Software Engineering braucht, und wieso es so heißt: Engineering von Software.

Das wird besonders deutlich, wenn man sich den Unterschied zwischen einem Einzelkämpfer-Programmierer und einem professionellen Entwickler in einem Großprojekt betrachtet. Schon beim Programmierstil ergeben sich wesentliche Unterschiede.

Im Hauptteil folgt die Vorlesung dem Ablauf in einem Softwareprojekt, von Anforderungen über Entwurf bis zu speziellen Strukturen, den Design Patterns. Danach kommt dann noch die Programmierung, die in Kapitel 2 vorgezogen wurde.

Querschnittlich ist ständig technisches Management (des Codes und der Dokumente) zu betreiben, und außerdem Projektmanagement.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Sie sollen wissen, was mit dieser Grundlagen-Vorlesung beabsichtigt ist, was Sie also nachher wissen und tun können – wenn das Ziel erreicht wird.

Davon zu unterscheiden sind zunächst die Inhalte, die ich Ihnen vermitteln will. Die finden Sie im Inhaltsverzeichnis.

Die vorliegende Vorlesung versucht vor allem einen guten Überblick über zentrale Themen zu verschaffen und die wichtigsten Zusammenhänge immer wieder darzustellen.

Dagegen kann die Vorlesung nicht eine starke Vertiefung einzelner Themen leisten oder intensiv auf die Programmierung eingehen. Dafür gibt es andere Veranstaltungen, im Bachelor und im Master-Studium.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Beachten Sie bitte:

Sie erfahren die theoretischen Grundlagen und Zusammenhänge am ehesten in der Vorlesung.

Die Übungen brauchen Sie aber auch! Nicht nur, weil Sie dort Bonuspunkte sammeln können, sondern vor allem, weil sie dort Fragen besprechen können, die bei Ihnen aufgetreten sind. Und weil sie Verfahren oder Techniken einmal, zweimal und vielleicht noch öfter ausprobieren, besprechen, wirklich ein-üben können.

In der Vorlesung selbst können diese „praktischen Dinge“ auch nur am Beispiel gezeigt und besprochen werden.

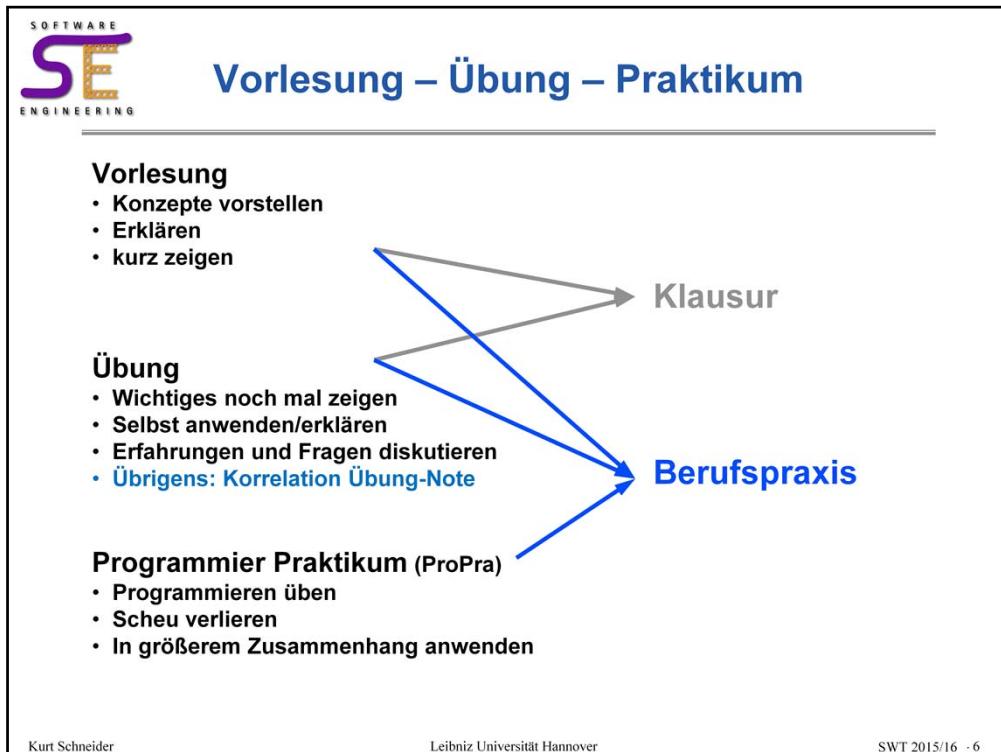
Richtig können werden Sie diese aber nur, wenn Sie sie selbst gemacht haben.

Unsere Webseite www.se.uni-hannover.de sollten Sie kurz vor dem Prüfungstermin noch einmal konsultieren.

Dort stehen immer Neuigkeiten, aber z.B. auch, wenn die Prüfung verschoben oder in einen anderen Raum verlegt werden muss.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Sie brauchen alle drei angebotenen Lehrveranstaltungen:

- Die Vorlesung liefert die Konzepte und zeigt sie jeweils an einem Beispiel. Hier lernen Sie, wie etwas im Prinzip funktioniert, wieso und warum man gewisse Dinge tut. Sie können damit erklären, was man tun muss und wieso.
- In den Übungen werden zwar auch Begriffe abgefragt. Hauptsächlich sollen Sie aber Verfahren selbst anwenden und feststellen, wo es Schwierigkeiten gibt.
- Sie brauchen Vorlesung (nicht nur die Folien davon!) und Übungen, um die Klausur gut zu bestehen.
- Für die Berufspraxis brauchen Sie zusätzlich die Fähigkeit, sich schnell und ohne Scheu in einer Programmiersprache auszudrücken. Dazu dient das Programmier-Praktikum. Für die Klausur brauchen Sie es nicht.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

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

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

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 7

Bitte sehen Sie sich die Bücher mal an, die hier auf den Folien zitiert werden. Vielleicht finden Sie eines, das Sie noch genauer durchblättern wollen. Das kann für Sie ein wertvolles Stück Extra-Wissen werden.

Generell ist diese Vorlesung als selbst-verständlich angelegt: Sie brauchen nicht unbedingt Sekundärliteratur zu lesen, wenn Sie Vorlesung und Übung regelmäßig besucht haben. Wenn Sie das aber aus irgendeinem Grund nicht getan haben, werden die Folien – und wahrscheinlich auch diese Erläuterungen – nicht in allen Fällen ausreichen. Schließlich versuchen die Dozenten in allen Vorlesungen ja, Ihnen den Inhalt möglichst nahe zu bringen.

Wenn Ihnen das nicht reicht (oder Sie eben einen Teil von Vorlesung und Übungen versäumt haben), dann können Sie zu den zitierten Büchern bzw. Papieren greifen.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Persönlich kann ich am meisten von einem Vortrag (auch einer Vorlesung) profitieren, wenn ich mir nicht alles mitschreiben muss, aber zu den vorliegenden Unterlagen Notizen mache. Ohne eigene Notizen wäre ich ja alleine auf mein Gedächtnis angewiesen – oder müsste davon ausgehen, dass alles, was gesagt wurde, auch in den Unterlagen (hier: auf den Folien) steht.

Das aber ist in dieser Vorlesung natürlich NICHT beabsichtigt: Die Folien sind so gestaltet, dass man den Stoff am besten versteht, wenn man sie erklärt bekommt.

Fazit: Schreiben Sie sich etwas mit. Das ist dann Ihr ideales Skript!

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

Softwaretechnik

Kapitel 1



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

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

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

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Es gibt viele Programme, die gut gemeint, aber schlecht gemacht sind. Die Oberfläche oben zeigt eine Kasse, bei der man Bilder antippen kann. Es kann aber für einen Bediener viel schwieriger sein, die Gerichte zu erkennen, als sich eine Bezeichnung zu merken. Außerdem gibt es viele Zusatzfunktionen wie Summen, übersichten, Farbcodes; das Programm ist nicht einfach zu bedienen, obwohl sicher genau das Gegenteil erreicht werden sollte. Software hat viele Möglichkeiten – leider auch, zum Falsch-Machen. Wie kann man professionell und systematisch gute Software entwickeln?

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Software ist überall

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

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 12

Weil Software zunehmend alle Bereiche unseres Lebens durchdringt, wird sie immer wichtiger. Und es wird immer wichtiger, sich die Konsequenzen schlechter Software (von schlechter Qualität) vorzustellen.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

The slide features a cartoon illustration of a man in a blue suit running towards the left, with a lightbulb above his head, symbolizing an idea. In the background, there are several other people and computer monitors, suggesting a busy office environment. The title 'Mit SW ist jeder ständig konfrontiert' is displayed in large blue letters, with the 'SE' logo of the FG Software Engineering at the top left.

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

Ein relativ neues Phänomen ist, dass nicht nur Informatiker und Fachleute, sondern jeder heutzutage auf Schritt und Tritt mit Software konfrontiert ist.

Die Konfrontation findet auf verschiedene Weise statt (siehe Folie).

In jedem Schritt sind Menschen involviert, die auf andere Weise mit SW zu tun haben.

Sie könnten sich an jeder Stelle später wiederfinden. Dann sollten Sie stets im Hinterkopf behalten, dass es auch alle anderen Schritte (und die dort involvierten Personen) geben muss, damit Software nützlich eingesetzt wird. Als professionelle Software Ingenieure sollten Sie diese gesamte „Wertschöpfungskette“ bedenken, damit sie an keiner Stelle unterbrochen wird.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

Fazit: Einfluss von Software

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

The diagram features a central yellow circle labeled "Software". Three blue triangles point from this center to three icons: a yellow computer monitor on the right, a yellow euro symbol on the left, and a yellow human figure at the top. This visualizes how software influences these three interconnected areas.

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 14

Ihre große Verbreitung in allen Lebensbereichen, für wichtige Unternehmungen und in den verschiedensten Hinsichten führt dazu, dass Software vieles beeinflusst:

- Sie ist ein Katalysator für moderne Technik geworden,
- Sie treibt oder bremst bedeutende Teile der Wirtschaft und
- sie wirkt auf die Menschen, die sie verwenden, die sie erstellen.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Wenn es dagegen nicht funktioniert

- Katastrophenbeispiele
 - Flugzeug bremst nicht
 - Marssonde geht verloren
 - Börsenprogramme schaukeln sich auf
 - Fahrkartautomat 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

Als Informatiker sind wir nahe am „Puls der Zeit“.

Wir sind aber auch verantwortlich, wenn unser Produkt zu Katastrophen oder Pannen führt.

Und keiner sollte sich einreden, man könne SW ganz ohne Verantwortung entwickeln. Nicht bei dieser starken Durchdringung aller Lebensbereiche.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

The screenshot shows a news article from the website [Software-Engineering.de](http://www.software-engineering.de). The title is "Software-Probleme in den Medien". Below it, a sub-headline reads "Steuererklärung Software-Problem lädt Finanzämter". The date is Saturday, 27.02.2010, 16:40. The article discusses software issues at tax offices, mentioning the Vitruvian Man logo and a yellow SIM card. It also covers the "Elektronische Gesundheitskarte" (Electronic Health Card) and the "Toll-Collect" system for toll collection.

Software-Problem lädt Finanzämter

Samstag, 27.02.2010, 16:40

★★★★★ 0 Empfehlen 0 Twitter 0 1 0

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

Identifikationsnummer (soweit schon erhalten)
An das Finanzamt
Bei Wohnsitzwechsel

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 funktioniere nicht, die

Kurt Schneider Leibniz Universität Hannover

Elektronische Gesundheitskarte

Toll-Collect

SWT 2015/16 · 17

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Es geht hier darum, viele Aufgaben professioneller zu machen. Auch wenn sie im Prinzip schon Programmieren können, fehlt Ihnen vielleicht noch die Erfahrung und die Fähigkeit, sich wirksam mit anderen abzustimmen und ihre Ergebnisse zu abzulegen, dass andere sie leicht finden und weiter nutzen können.

Einerseits soll die Sache weiterhin Spaß machen. Andererseits ist die Softwareentwicklung viel mehr als reine Programmierung. Und auch professionelle Programmierung muss auf Aspekte achten, an die man beim „Hobbyprogrammieren“ nie denken würde.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Über die Dinge links kann man sich leicht und mit Recht aufregen.

Für angehende Informatiker reicht das aber nicht. Sie sollten sich vor allem fragen: Wie kann ich es besser machen?

Dazu sind die Hinweise rechts gedacht.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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.)

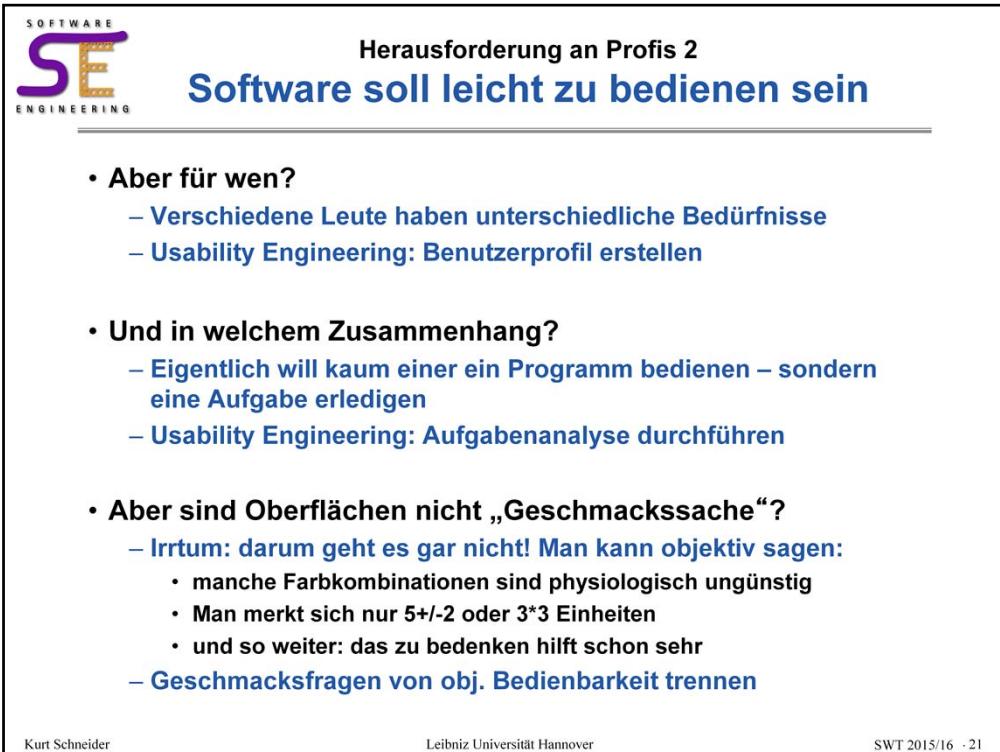
Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 20

Wie können also Profis an die Software-Entwicklung herangehen?

Welche Herausforderungen müssen sie bewältigen

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Nächster Schritt:

Nehmen wir an, die SW täte nun in etwa, was von ihr verlangt wird.

Dann kann es sein, dass sie es nicht auf die Weise tut, wie man es sich wünschen würde.

Bedienbarkeit ist eine wichtige Eigenschaft, die SW oft fehlt.

Systematisch kann man bedienbare Software durch Usability Engineering schaffen; dessen Grundlagen fehlen aber den meisten Entwicklern.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

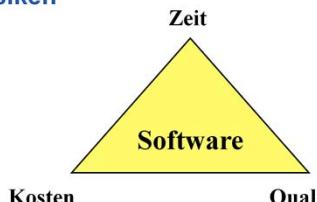
Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

**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



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

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 22

Nächster Schritt:

Selbst hervorragend bedienbare SW kann noch „schlecht“ sein:

Nämlich dann, wenn sie zu spät kommt oder zu teuer ist.

Die wirtschaftliche Dimension im SE ist wichtig:

Ein scheinbares Optimum einer Lösung kann aufgrund hoher Kosten oder weil sie lange dauert durchaus „schlecht“ sein.

Beispiel: Toll Collect

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

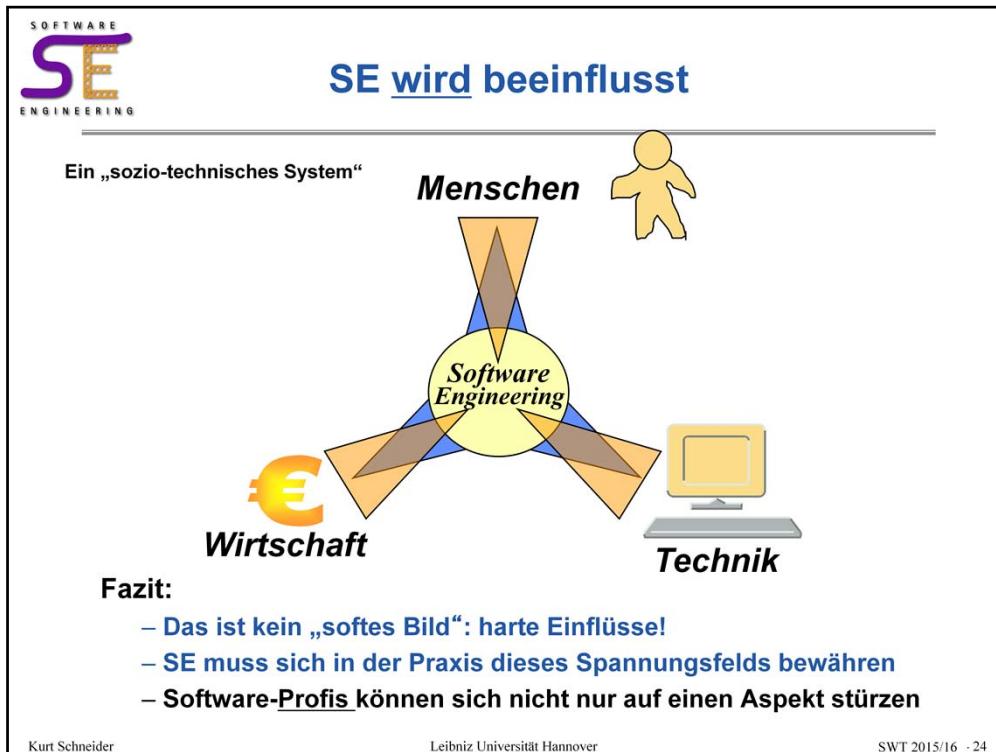
Nun kommen wir dazu, wie Software Engineering einen Teil der Probleme lösen soll.

Zum SE gehören einige Techniken, die selbst mit –management oder –engineering enden. SE ist der Überbegriff.

Übrigens geht es nicht nur um die Entwicklung neuer Programme auf der grünen Wiese. Viele häufiger wird Bestehendes weiterentwickelt.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Technik ist Rahmenbedingung unseres Handelns: sie muss man beherrschen. Übrigens kommen Sie nur mit der Technik nicht zu Rande, täuschen Sie sich nicht!

Sie müssen immer auch die beiden anderen Aspekte bedenken. Das gehört zum „Engineering“.

Die Wirtschaft beschränkt oft die Möglichkeiten, das als optimal erkannte auch umzusetzen. Der Mensch kann und will nicht alles tun, manches kann er aber besonders gut

-wer das berücksichtigt, hat große Vorteile

Fazit: wir werden uns immer wieder explizit an alle drei Einflüsse erinnern. Denn nur wenn etwas in SE unter diesen drei Einflüssen funktioniert, funktioniert es wirklich

alles andere ist „graue Theorie“

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Die beiden Begriffe sind interessant und zentral:

Software schließt nach dieser Definition auch Dokumente und „Daten“ ein, die zum Betrieb eines Computersystems gebraucht werden. Nicht nur Code!

Software Engineering hat den praktischen Aspekt (1) und den auf die Forschung gerichteten Aspekt (2). Auch hier ist SW in der breiten Bedeutung gemeint.

Lernen Sie diese Definitionen auswendig.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover

The slide features the logo for Software Engineering (SE) in the top left corner. The main title is '„Ingenieursprinzipien“'. Below it is a subtitle: '„A systematic, disciplined, quantifiable approach to the development, operation, and maintenance of X“'. A horizontal line separates this from a section titled 'Konsequenzen' (Consequences). This section lists five points with associated bullet points:

- **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“

At the bottom of the slide, there are three small text elements: 'Kurt Schneider', 'Leibniz Universität Hannover', and 'SWT 2015/16 · 26'.

Die hier dargestellten Prinzipien sind z.B. beim Brückenbau selbstverständlich.

In der Informatik aber nicht:

- Oft fehlt das Kostenbewusstsein, man versucht, „die beste Lösung“ umzusetzen, vergisst aber, dass sie zu teuer wird. Das ist nicht ideal.
- Qualität als Leitgedanke hat sich in der SW noch nicht durchgesetzt. Es gibt viele, die ungetestete Software abliefern.
- Normen und Regeln kennen die wenigsten. Sie leiten nicht unser Handeln, vielmehr wird vieles ständig neu erfunden.
- Wiederverwendung ist zwar versucht worden, der große Durchbruch steht aber in der SW noch aus.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

$SW = \text{Risiko} + \text{Chance}$
Muss man systematisch angehen, nicht zufällig

Kurt Schneider Leibniz Universität Hannover SWT 2015/16 · 27

Man hat sich mit dem Problem der relativ geringeren Produktivität der Software-Entwicklung auseinandergesetzt.

- Diese Entwicklung kommt nicht durch eine Verschlechterung der SW, sondern durch eine noch viel stärkere Verbesserung der HW-Situation zu Stande.
- Da nun aber SW immer wichtiger wird, musste die Gesellschaft etwas tun, um den Spalt zu schließen. Sonst bleibt die SW in ihrer berühmt-berüchtigten „Software-Krise“ stecken. Davon geht ein großes Risiko für die Gesellschaft aus.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



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

Die Idee ist einfach:

Wie Häuser oder Brücken einfach auch SW behandeln.

Von Ingenieuren wird das Ingenieurstum in dieser Metapher verklärt. Es wird kaum beachtet, dass SW die obigen Unterschiede aufweist. Dadurch können nicht einfach die gleichen Praktiken eingesetzt werden.

F.L. Bauer hat das Wort Software Engineering eingeführt und erläutert, was er darunter versteht. Die Ingenieursprinzipien waren vorne aufgeführt worden (Kostendenken usw.).

Was man auch sagen muss:

- Im Software Engineering gibt es viel Neues; es wird in dieser Vorlesung zum Teil angesprochen, Vieles aber gibt es in Fortgeschrittenen Veranstaltungen wie „Moderne Software-Entwicklungsmethoden (MEM)“ oder „Requirements Engineering“ oder im XP-Labor (Agile Methoden).

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



The banner for the NATO Science Conference in Garmisch-Partenkirchen, 1968, features the logo "SOFTWARE ENGINEERING" with "SE" in large letters. The main title is "NATO Science Conference Garmisch-Partenkirchen 1968". Below the title is the URL <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/N1968/>. A black and white photograph of Prof. F.L. Bauer speaking at a podium is on the left, with the caption "Prof. F.L. Bauer". To his right is another black and white photograph showing a conference room with many people seated around a long table, with the caption "Der Begriff „Software Engineering“ wird geprägt". The footer includes the names "Kurt Schneider", "Leibniz Universität Hannover", and "SWT 2015/16 · 29".

Dies ist ein wenig SE-Geschichte:

F.L. Bauer auf der berühmten Konferenz 1968, auf der Software Engineering aus der Taufe gehoben wurde.

Das ist ein Fakt, den jeder Informatiker kennen muss, der eine SE-Vorlesung gehört hat.

Vorlesung „Grundlagen der Softwaretechnik (SWT)“ im WS 2015/2016

Prof. Dr. Kurt Schneider, FG Software Engineering, Leibniz Universität Hannover



Kennen Sie solche Situationen?

Sie sind sicher auch reizvoll, aber dazu brauchen Sie kein Informatikstudium.

Im Studium geht es darum, wie man tagein, tagaus systematisch und mit guter Qualität Software entwickeln kann.

Bei normalen Arbeitsstunden und geregelten Essenszeiten.