

Softwaretechnik 1(A)

Einführung



Themen

- **Begriff: Softwaretechnik**
- **Ingenieurmäßige Entwicklung – Ingenieurmäßiges Vorgehen**
- **Softwaretechnische Kompetenz**

Begriff: Softwaretechnik

Begriffe

Was ist Software?

- Software

Menge von Programmen oder Daten zusammen mit begleitenden Dokumenten, die für ihre Anwendung notwendig oder hilfreich sind.

[Hesse et al. 84: Ein Begriffssystem für die Softwaretechnik]

Woher kommt der Begriff Softwaretechnik (SWT)?

- 1965: Der Begriff der Software-Krise etabliert sich in Industrie und Wissenschaft.
 - Fehler in Computersystemen sind fast immer auf Softwarefehler zurückzuführen.
 - Software wird nicht termingerecht und/oder zu höheren Kosten als geschätzt fertiggestellt.
 - Software entspricht oft nicht den Anforderungen ihrer Benutzer.
- 1968: Der Begriff der Software-Technik (“Software Engineering”) wird geprägt.
 - Er wird von F.L. Bauer im Rahmen einer Studiengruppe der NATO zum Thema Informatik geprägt.
 - Auslöser für Einführung des Begriffs Software-Technik war die Software-Krise.
 - Bahnbrechend war die NATO-Konferenz “Working Conference on Software Engineering” vom 7. - 10. Oktober 1968 in Garmisch.

Begriffe

Was ist Softwaretechnik (SWT)?

Einige Definitionen

- Software Engineering
The systematic approach to the development, operation, maintenance, and retirement of software.
[Ansi 83: IEEE Standard Glossary of Software Engineering Terminology]
- 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 et al. 84: Ein Begriffssystem für die Softwaretechnik]
- Softwaretechnik (syn.: Software Engineering) ist
 - die Entwicklung
 - die Pflege und
 - der Einsatz qualitativ hochwertiger Software unter Einsatz von
 - wissenschaftlichen Methoden
 - wirtschaftlichen Prinzipien
 - geplanten Vorgehensmodellen
 - Werkzeugen
 - mit quantifizierbaren Zielen

[B. Kahlbrandt 98: Software-Engineering: Objektorientierte Software-Entwicklung mit der UML]

Begriffe

Was ist Softwaretechnik (SWT)?

Einige Definitionen

- Software -Technik

Zielorientierte Bereitstellung und systematische Verwendung von

- Prinzipien,
- Methoden,
- Konzepten,
- Notationen und
- Werkzeugen

für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Software-Systemen.

Zielorientiert bedeutet die Berücksichtigung z.B. von Kosten, Zeit, Qualität.

[H. Balzert 96: Lehrbuch der Software-Technik (Band 1)]

Begriffe

Was ist Softwaretechnik (SWT)?

Einige Definitionen

- Software -Technik

Zielorientierte Bereitstellung und systematische Verwendung von

- Prinzipien,
- Methoden,
- Konzepten,
- Notationen und
- Werkzeugen

für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Software-Systemen.

Zielorientiert bedeutet die Berücksichtigung z.B. von Kosten, Zeit, Qualität.

[H. Balzert 96: Lehrbuch der Software-Technik (Band 1)]

Ingenieurmäßige Entwicklung

Ingenieurmäßiges Vorgehen

Ingenieurmäßiges Vorgehen

Softwaretechnik vs. Programmierung

- Softwaretechnik Programmierung
In den ersten beiden Semestern des Studiums haben Sie mindestens 2 programmier-intensive Veranstaltungen besucht
⇒ Sie können Programmieren, aber können Sie auch Software entwickeln?

Paradox

- Hundehütte → Wolkenkratzer
Wenn jemand mal eine Hundehütte gebaut hat, würden wir ihn noch lange keinen Wolkenkratzer bauen lassen.
- Programm → Softwaresystem
Wenn jemand dagegen programmieren kann, lässt man ihn heute aber oft bedenkenlos Software entwickeln.

Ingenieurmäßiges Vorgehen



Hundehütte

- kann von einer Person gebaut werden
- erfordert
 - minimalen Entwurf
 - einfachen Prozess
 - einfache Werkzeuge



Wolkenkratzer

- wird von einem Team erbaut
- erfordert
 - Entwurf (durch Architekt)
 - Analyse (durch Statiker)
 - Genehmigung (durch Bauaufsicht)
 - Realisierungsprozess
 - aufwendige Werkzeuge

Ingenieurmäßiges Vorgehen - Auftrag

- **Hamster-Programm**

Im ersten Semester haben Sie eine Hamster-Programmierumgebung entwickelt. Aufgrund der guten Ergebnisse werden Sie beauftragt ein E-Learning-Tool für die Einführung in die Programmierung (Java, Algorithmen, Datenstrukturen) für die Sekundarstufe II zu erarbeiten.

- **Wie** lange benötigen Sie, wenn Sie bis zu vier Mitarbeiter beschäftigen können?

- 4 →
- 3 →
- 2 →
- 1 →

- **Was** soll diese Anwendung kosten?

Ingenieurmäßiges Vorgehen

... wenn Software-Entwicklung Hausbau wäre ...



8. 10. 000 €



⇒ 80. 000 €

Ingenieurmäßiges Vorgehen

Die Tacoma Narrows Bridge (TNB)

→ Spitzname "Galloping Gertie"



im Film
vom
7. November 1940



<http://youtu.be/j-zczJXSxnw>

Ingenieurmäßiges Vorgehen

- Ingenieurwesen ist auf erfahrungsgeleitetes Vorgehen angewiesen
→ "Normales Vorgehen"
- Das Vorgehen wird immer irgendwo von vorherigen Erfahrungen abweichen
→ "Radikales Vorgehen"
- Ingenieurmäßiges Vorgehen besteht darin, solche Abweichungen zu vermeiden/gering zu halten
→ Radikales Vorgehen nur, wo unbedingt nötig!
- Ansätze für normales Vorgehen:
 - Was sind die typischen **Probleme in der SW-Entwicklung?**
 - Was sind die dazu passenden **Lösungsansätze der SWT?**

Ingenieurmäßiges Vorgehen

Fallbeispiel:

Die Tacoma Narrows Bridge (TNB)

- Am 1. Juli 1940 wurde bei Tacoma im US-Bundesstaat Washington eine neue Hängebrücke über dem Puget Sound eröffnet.
- Baubeginn: 1938
Chefberater: Leon Moisseiff (Experte für die Konstruktion von Hängebrücken)
Erfahrung: Manhattan Bridge, Bay Bridge, George Washington Bridge und die Golden Gate Bridge



Golden Gate Bridge
1970m Länge
1280m Spannweite



Bay Bridge (west)
2820m Länge
700m Spannweite



Tacoma Narrows B.
1520m Länge
850m Spannweite



Leon S. Moisseiff
(1872 - 1943)



Ingenieurmäßiges Vorgehen

Die Tacoma Narrows Bridge (TNB)

hatte überwiegend ganz übliche Parameter für die Einzelteile:

- übliche Pylonen
bezügl. Bauart, Größen, Materialien etc.
- übliche Seile
bezügl. Bauart, Größen, Materialien etc.
- eine übliche Verankerung
bezügl. Bauart, Größen, Materialien etc.
- eine übliche Fahrbahn
bezügl. Bauart, Größen (2 Fahrspuren, 2 Gehwege), Materialien
- einen weitgehend üblichen Unterbau
bezügl. Größen, Materialien



Eröffnungsfeier am 1. Juli 1940

unüblich war lediglich die **Schlankheit**: \Rightarrow niedrige Steifigkeit und niedriges Gewicht

- Tragwerk nur 12m breit und 2,40m hoch
- Das Verhältnis Tragwerkhöhe: Spannweite war 1:350 (üblich war eher bis zu 1:90)
- Das Verhältnis Tragwerkbreite: Spannweite war 1:72 (Golden Gate Bridge: 1:47)

[http://www.bernd-nebel.de/bruecken/index.html?/bruecken/4_desaster/tacoma/tacoma.html]

Ingenieurmäßiges Vorgehen – Ursachen für den Zusammenbruch

- Keine Ursache: Die Windstärke
Sie war mit ca. 75 km/h noch weit im zulässigen Bereich
- Unfallursache (Bedingungen):
Die Längsschwingungen, für die die Brücke nicht konstruiert war
- Unfallherkunft (Urgründe):
 - Herkunft:
Das Rechenmodell von Moisseiff berücksichtigte nur Querschwingungen
 - Urgrund:
Längsschwingungen waren vor TNB nie relevant gewesen, weil vergleichbar lange Brücken stets viel schwerer und steifer gebaut worden waren

⑧ Moisseiffs Konstruktion lag zu weit außerhalb des Bereichs, über den es Erfahrungen gab!

Ingenieurmäßiges Vorgehen

Erforderliche kognitive Kompetenzen in der SWT

1. Kennen	Gleich wiedergeben	vorausgesetzt
2. Verstehen	Anders wiedergeben	
3. Anwenden	Folgern in konkreten Fällen	
4. Analysieren	Zerlegen; Verstehen der Struktur	
5. Synthetisieren	Neue, ähnliche Strukturen bilden	
6. Beurteilen	Brauchbarkeit bewerten	

[nach Bloom et al]

Softwaretechnische Kompetenz

Kompetenz

- Sie werden nach dem Studium **nicht dafür bezahlt**
 - dass Sie einen Hochschulabschluss haben
 - dass Sie etwas wissen
 - dass Sie etwas tun (z.B. programmieren)
 - dass Sie irgendwelche Probleme lösen
- sondern dafür
 - dass Sie solche Probleme lösen, bei denen der Nutzen der Lösung höher ist als die Kosten der Lösung - am besten sogar viel höher
- Dazu müssen Sie lernen:
 1. Probleme und ihre Wichtigkeit zu verstehen
 2. Probleme zu lösen
 3. Kosten und Nutzen von Technologie und Methoden abzuschätzen

Kompetenz

- Nutzenmaximierung
 - **Problem** des Kunden und der Nutzer **verstehen**
 - **Kompetenzen** und **Kontext** der Stakeholder **einordnen können**
 - **Technologische Kompetenz einbringen**
 - **wertvolle Anforderungen aufdecken** und **einbringen**
 - Software **hoher Passgenauigkeit produzieren**
- Kostenminimierung
 - Software mit möglichst wenig Arbeitsaufwand fertigstellen
Schlüsseltechniken: **Wiederverwendung, Qualitätssicherung, Risikomanagement →**
Technologiekompetenz
 - Künftige Kosten vermeiden (durch Qualität)
 - keine hohen Kosten für **Fehlerkorrekturen =>** hohe Zuverlässigkeit und Robustheit
 - keine hohen Kosten bei **späteren Änderungen =>** gute Wartbarkeit

Fragen



Aufgabe



Fachkompetenz im Vorgespräch

Sie haben einen Kundentermin in der Projektanbahnungsphase.

Der Kunde erklärt Ihnen, ein Problem mit der Kugelproduktion zu haben.

Produzierte Kugeln (3 kleine und 2 große) werden in 5er-Sets gruppiert.

Dann muss die schwerste Kugel mit möglichst wenig Schritten ermittelt werden,
wobei nur eine einfache Balkenwaage ohne Gewichte zur Verfügung steht....

Können Sie ihm helfen?