

Allgemeines zum Thema Software-Engineering

Historie

Zunächst:

- kleine DV-Anlagen und kleine Programme
- keine allgemeine Programmiertechniken oder Programmierrichtlinien
- individuelle Programmier-„Künste“

Später:

- leistungsfähigere DV-Anlagen führen zu (sehr) großen SW-Systemen
- Jedoch: keine neuen Software-Technologien

Folge:

So genannte **Software-Krise** (ca. 1965):

- sinkende HW- und steigende SW-Kosten (Wartung, Anpassung und Erweiterung großer Software-Systeme)
- Methoden und Techniken reichen nicht aus, um die komplexen Software-Systeme zu handhaben

Konsequenz:

- In zwei NATO-Konferenzen zur genannten Problematik wird der Begriff **Software Engineering** geprägt
- In Deutschland: Gründung der Informatikstudiengänge

Definitionen

Engineering:

„Die schöpferische Anwendung von wissenschaftlichen Prinzipien auf **Entwurf** und **Entwicklung** von Strukturen, Maschinen, Apparaten oder Herstellungsprozessen ... alles im Hinblick auf eine gewünschte Funktion, Handlungsökonomie und Sicherheit von Leben und Eigentum ... „ (Wikipedia u.a.)

Software Engineering:

„*Software Engineering* ist die Anwendung von Prinzipien, Fähigkeiten und Kunstfertigkeiten auf den **Entwurf** und die **Erstellung** von Programmen und Systemen von Programmen (Software).“

Ziel des Software Engineering:

Erstellung von Software-Systemen in ökonomischer Art und Weise, die bestimmten Qualitätsanforderungen genügt

Unterschied Programm – Software

=> Die Komplexität

Programm:

von nur einem Programmierer mit Methoden des
Programmierens im Kleinen erstellt

Software:

- von deutlich mehr als einer Person erstellt (typisch: mehr als 3 Personen)
- hoher Arbeitsaufwand (typisch: 5 bis 50 Personenjahre)
- Benutzer sind üblicherweise verschieden von den Programmierern
- Erstellung mit Methoden des *Programmierens im Großen*
(Problemanalyse, Projektplanung, Strukturierung, Modularisierung, usw.)
- Verwendung von CASE-Tools (Computer Aided Software Engineering)

Randbedingungen für die Erstellung eines SW-Produkts

Kundensicht:

- Wünsche an die SW
- Preisvorstellung
- gewünschte Lieferzeit
- Hardware/Betriebssoftware-Voraussetzungen
- Kenntnisse/Fähigkeiten der späteren Nutzer

Anbietersicht (zusätzlich):

- Wünsche an die SW
- Wiederverwendbarkeit der Software für andere Kunden
- eigene finanzielle Möglichkeiten
- Das für die SW-Entwicklung zur Verfügung stehende Personal (Anzahl, Motivation und Know-how)

Was sind Qualitätsanforderungen?

- Wodurch zeichnet sich Qualität allgemein aus?
- Wodurch zeichnet sich die Qualität einer Software aus?