

# 1.

# Organisatorisches und Motivation

Software Engineering ist die Entdeckung und Anwendung solider Ingenieur-Prinzipien mit dem Ziel, auf wirtschaftliche Art Software zu bekommen, die zuverlässig ist und auf realen Rechnern läuft, verhindert Schäden, senkt die Gesamtkosten und hält Fortschritte fest. Es sind Methoden, Sprachen und Werkzeuge durch Konzepte miteinander verbunden. Reine Programmierfehler, wie ein falsches Zeichen kann eine Katastrophe auslösen, wie zum Beispiel die Mariner-1-Rakete. Je früher ein Fehler in der Entwicklung entdeckt wird, desto günstiger ist dessen Beseitigung.

- Software
- Software Engineering

# Software

Software beschreibt sämtliche **nicht** physische Bestandteile eines Computers, Computernetzwerks oder mobilen Geräts. Gemeint sind Programme und Anwendungen (Betriebssystem), die den Computer für den Nutzer funktionstüchtig machen.

- Software ist ein **technisches Produkt**, dass *entwickelt* wird und *Eigenschaften* hat
- Software kann als **ideales technisches Produkt** angesehen werden, da es unter anderem *nicht verschleißt*
- *Eigenschaften:*
  1. **immateriell**: entwickelbar, kopierbar, wieder verwendbar, nur durch Fehler abnutzbar
  2. **keine natürliche Lokalität**
  3. **realisiert keine stetige Funktion**: Zusammenhang bei Eingabe und Ausgabe ist nicht von Funktion beschreibbar
  4. **Software-Systeme funktionieren autonom**, ohne menschliche Hilfe
  5. „**Werkstoffe**“ der Software sind **amorph und universell**

# Software Engineering

Ist die Herstellung und Anwendung einer Software, wobei mehrere Personen beteiligt sind oder Versionen entstehen.

1. **Werkzeuge** zur automatischen oder vom Benutzer gesteuerten Transformation und Speicherung von Information.
2. es werden **Programmier- und Modellierungssprachen** genutzt, um die *Syntax festzulegen*
3. **Methoden** werden genutzt, um die Software *herzustellen*
4. mit diesen drei Dinge entstehen **Konzepte**

## Was macht man im Software Engineering?

- **Analyse:** bestehende *Problem* zu *durchdringen und zu verstehen* (statisch oder dynamisch)
- **Spezifikation der Anforderungen:** Anforderungen müssen geordnet, dokumentiert, geprüft, ergänzt und korrigiert werden
- **Architekturentwurf, Spezifikation der Module** (System- und Softwareentwurf)
- **Codierung und Modultest:** Übersetzt den Code, implementiert und testet
- **Integration, Test, Abnahme**
- **Betrieb und Wartung**
- **Auslauf und Ersetzung**

SE als defensive Disziplin: Sie verhindert Schäden und sollte generell beachtet werden.

SE als globale Optimierungsmaßnahme: Separate Einsparungen an einzelnen SE-Aktivitäten

SE als Technologie für die Köpfe: Wissenschaftliche Erkenntnis und neue Technologie lassen sich gut in die Anwendung transferieren, wenn sie sich in Produkten niederschlagen



