

Grundlagen der Programmierung

Dr. Christian Herzog
Technische Universität München

Wintersemester 2014/2015

Kapitel 1: Einführung

Ziel der Vorlesung Grundlagen der Programmierung

- ❖ **Aktive Beherrschung von Modellierungsmethoden**
 - Modellierung von Problemen aus der Realität
 - Konstruktion von Lösungen mit Hilfe der Informatik
 - Systematisches Entwickeln einer Lösung in Form eines Informatik-Systems mit Hilfe von Programmiersprachen
- ❖ **Aktive Beherrschung einiger theoretischer Grundlagen**
 - Algorithmus, Algebra, Verifikation, Automaten
- ❖ **Aktive Beherrschung verschiedener Programmierparadigmen**
 - Funktionale, objektbasierte/imperative und objektorientierte Programmierung
- ❖ **Aktive Beherrschung der Programmiersprache Java**
 - Die Vorlesung ist aber **kein Programmierkurs für Java**
- ❖ **Aktive Beherrschung von Abstraktion**

Überblick über das Einführungs-Kapitel

- ❖ Was ist Informatik?
- ❖ Was sind die Aufgaben der Informatik?
- ❖ Realität und Modell
- ❖ Beispiele von Informatik-Systemen

Was ist Informatik?

Informatik ist die Wissenschaft von

- ❖ der theoretischen Analyse und Konzeption von Informatiksystemen (**Theoretische Informatik**)
- ❖ der organisatorischen und technischen Gestaltung von Informatiksystemen (**Systembezogene Informatik**)
- ❖ der Realisierung von Informatiksystemen, insbesondere der technischen Komponenten (Hardware) (**Technische Informatik**)

Aufgabe des Informatikers

- ❖ **Analyse: Das Verständnis von Problemen aus der Realität**
 - Realität ist dabei ein
 - ◆ soziales System (z.B. eine Firma, eine Versicherungsgesellschaft),
 - ◆ ein physikalisches System (z.B. die Erde, das Universum),
 - ◆ oder ein künstliches System (z.B. ein System das von Informatikern geschaffen wurde).
- ❖ **Synthese (Design): Die systematische Entwicklung einer Lösung des Problems**
 - Die Evaluierung und Verwendung von (existierenden/neuen) Hardware-und Softwarebausteinen
- ❖ **Implementation: Die Konstruktion der Lösung**
 - Mit Hilfe eines Informatik-Systems (auch Datenverarbeitungssystem genannt)
- ❖ **Wartung: Die Bereitstellung und Betreuung des Informatik-Systems**

Welche Fähigkeiten benötigt der heutige Informatiker?

- ❖ Lösen von Problemen aus der Wirklichkeit
- ❖ Modellierung mit statischen und dynamischen Strukturen (Subsysteme, Klassen, Datenstrukturen und Algorithmen, Automaten)
- ❖ Formale Spezifikation von solchen Modellen
- ❖ Wiederverwendung von Wissen (Entwurfsmuster)
- ❖ Analyse und Synthese von Datenstrukturen und Algorithmen
- ❖ Beherrschung von
 - Methoden (z.B. Rekursion)
 - Werkzeugen (z.B. Compiler)
 - Sprachen (z.B. Java)
- ❖ Ein gutes Verständnis der theoretischen Grundlagen
- ❖ Fähigkeit zur Gruppenarbeit
- ❖ Interaktion mit Kunden (Reden, Schreiben, Verhandeln)

Wirklichkeit und Modell

- ❖ **Wirklichkeit W:** Dinge, Personen, Abläufe in der Zeit, Beziehungen zwischen diesen Gegenständen
- ❖ **Modell M:** Begriffe von (real existierenden oder nur gedachten) Dingen, Begriffe von Personen, Begriffe von Abläufen in der Zeit, Beziehungen zwischen diesen Begriffen.

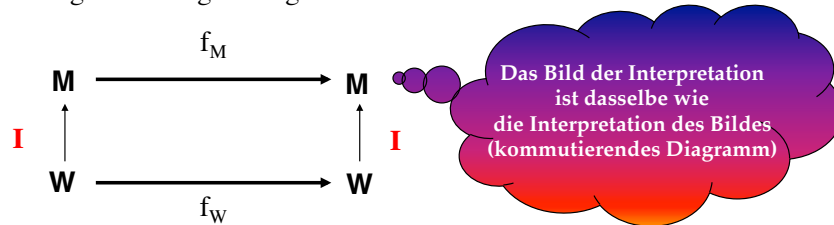
- ❖ Wirklichkeit nennen wir im folgenden auch oft Realität.

Warum Modelle?

- ❖ **Der Mensch nutzt Modelle**
 - um Details in der Wirklichkeit wegzulassen, so dass man mit einfacheren Schritten zu Schlüssen kommen kann. Wir nennen das **Abstraktion**.
- ❖ **Der Mensch nutzt Modelle**
 - um Einsichten in Vergangenes und Bestehendes zu gewinnen;
 - um Vorhersagen über zukünftiges Verhalten zu bekommen.

Was ist ein "gutes" Modell?

- ❖ **I: $W \rightarrow M$** : Abbildung, die Gegenstände der Wirklichkeit W auf ihre Begriffe im Modell M abbildet (Interpretation) mit folgender Eigenschaft:
 - Ist $f_M : M \rightarrow M$ eine beliebige Beziehung zwischen Begriffen in M , dann soll es auch in W eine Beziehung $f_W : W \rightarrow W$ geben, so dass folgendes Diagramm gilt:



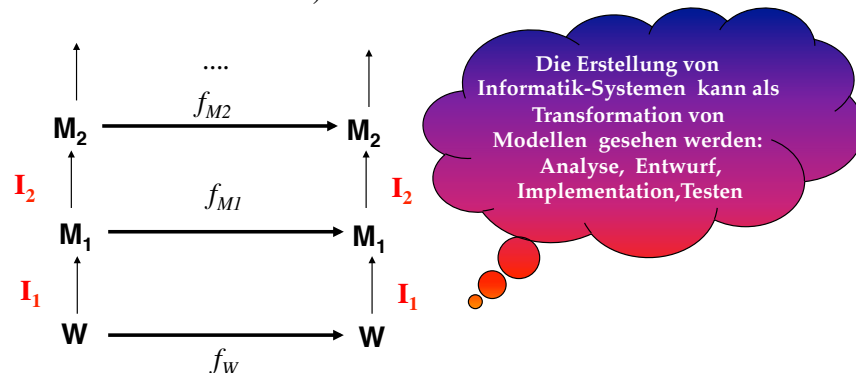
- ❖ Beziehungen, die im Modell M gelten, haben ihre Entsprechung in der Wirklichkeit W .

Modelle sind falsifizierbar

- ❖ **Der Wahrheitsgehalt eines Modells ist niemals völlig sicher**
 - “nach menschlichem Ermessen”
 - “nach dem heutigen Stand der Erkenntnis”
- ❖ **Popper (“Objective Knowledge”):**
 - Es gibt keine absolute Wahrheit bei der Modellierung der Wirklichkeit.
 - Wir können nur Modelle von der Wirklichkeit bauen, die solange “wahr” sind, bis ein Gegenbeispiel gefunden wird (“Falsifizierung”).

Modelle von Modellen von Modellen...

- ❖ **Modellierung ist ein relativer Begriff.** Man kann auch ein Modell wieder als reale Welt auffassen und davon ein weiteres Modell (mit noch mehr Abstraktionen) erstellen.



Systemarten

- ❖ **Wissenschaften die sich mit Systemen beschäftigen:**
- ❖ **Naturwissenschaften (Physik, Chemie, Biologie):**
 - Beschäftigen sich mit natürlichen Systemen (Kosmos, Mensch, Bienenstock).
 - Ziel ist das Verständnis der Natur.
- ❖ **Humanwissenschaften (Psychologie, Soziologie):**
 - Beschäftigen sich mit sozialen Systemen (Nation, Gemeinde, Gruppe, Studentenschaft, Unternehmen, Markt).
 - Ziel ist das Verständnis des Menschen.
- ❖ **Wissenschaften der künstlichen Systeme (Elektrotechnik, Maschinenbau, Informatik):**
 - Beschäftigen sich mit künstlichen Systemen (Regelkreise, Eisenbahnen, Informatik-Systeme).
 - Ziel ist das Verständnis des Verhaltens von künstlichen Systemen.

Was ist ein System?

- ❖ Ein System ist eine gedankliche Konstruktion.
- ❖ Ein System kann in vielerlei Hinsicht gesehen werden.
2 wichtige Ansichtsarten:
 - Statische Sicht: Ein System ist eine komplexe Ansammlung von Elementen zusammen mit den Beziehungen zwischen den Elementen.
 - Dynamische Sicht: Ein System ist eine Folge von Zustandsübergängen.
- ❖ Ein Informatik-System ist ein System, das auf einem Rechner ausgeführt wird.

Informatik-Systeme

- ❖ **Diese Vorlesung beschäftigt sich mit Informatik-Systemen**
 - Informatik-Systeme werden oft auch Datenverarbeitungssysteme genannt.
- ❖ **Was ist ein Informatik System?**
 - Wir kategorisieren Informatik-Systeme in 4 Typen, die wir im folgenden vorstellen:
 - ◆ Typ 1: Berechnung von Funktionen
 - ◆ Typ 2: Prozessüberwachung
 - ◆ Typ 3: Eingebettete Systeme
 - ◆ Typ 4: Adaptive Systeme

Typ 1: Berechnung von Funktionen

- ❖ Ein System, das eine Funktion $f: A \rightarrow B$ berechnet, die einen Definitionsbereich A in einen Wertebereich B abbildet.
 - ❖ **Beispiel:**
 - Umrechnung zwischen Fahrenheit und Celsius :
- $$C = 5.0 * (F - 32.0) / 9.0 \quad \text{oder} \quad F = 9.0 * C / 5.0 + 32.0$$

```
public class Temperature {
    public Temperature() {}

    public double FahrenheitToCelsius( double temp ) {
        return 5.0 * (temp - 32.0) / 9.0;
    }
    public double CelsiusToFahrenheit( double temp ) {
        return 9.0 * temp / 5.0 + 32.0;
    }
}
```

Weitere Beispiele für die Berechnung von Funktionen

- ❖ Übersetzer (Compiler), Assembler
- ❖ Klimavorhersage, Wettervorhersage
- ❖ ...

Typ 2: Prozessüberwachung

- ❖ Ein endlos laufendes System, das Eingangsdaten von anderen Systemen (Prozessen, Menschen) empfängt und Ausgangsdaten an solche Systeme sendet.
- ❖ Die Ausgangsdaten können den Start oder den Halt anderer Systeme (Prozesse, Menschen) bewirken.
- ❖ Die Ausgangsdaten sind eine Funktion der Eingabedaten und der Systemgeschichte.

- ❖ Systeme, die während der Laufzeit Daten einlesen und ausgeben, so dass diese noch während der Laufzeit des Systems extern weiterverarbeitet werden können, nennen wir **interaktive Systeme**.

Beispiele für Prozessüberwachung

- ❖ System zur Beeinflussung des Verkehrsflusses an einer Straßenkreuzung
- ❖ Beispiel aus der Informatik: Betriebssystem

Typ 3: Eingebettetes System

- ❖ Ein System, das im Verbund mit Komponenten, die nicht der Datenverarbeitung dienen, eine Aufgabe löst.
- ❖ Die anderen Komponenten können technische Aparaturen, aber auch Menschen oder betriebliche Organisationen umfassen.
- ❖ **Beispiele:**
 - Elektronischer Bremskraftverstärker in einem Auto
 - Space Shuttle Kontrollsystem
 - Steuerung eines Mobiltelefons oder einer Digitalkamera

Adaptives System

- ❖ Ein (eventuell eingebettetes) System, das sich Veränderungen der Wirklichkeit anpasst, insbesondere auch solchen, die das System selbst hervorruft.
- ❖ Das ursprüngliche Modell ist nicht auf Dauer gültig.
- ❖ **Beispiele:**
 - Regelkreis
 - Soziales System: Ökosystem
 - Intelligentes Tutorsystem

Inhalte der Vorlesung Grundlagen der Programmierung

- ❖ **Informatik-Systeme**
 - statische Sicht auf Systeme
 - dynamische Sicht auf Systeme
- ❖ **Algorithmen und Datenstrukturen**
- ❖ **Information und Repräsentation**
- ❖ **Texterstellungs-Systeme**
- ❖ **Funktionale Programmierung**
- ❖ **Objektbasierte (imperative) Programmierung**
- ❖ **Objektorientierte Programmierung**
- ❖ **Automaten und formale Sprachen**
- ❖ ...

Zusammenfassung

- ❖ **Informatik ist Problemlösung**
 - Modellierung ist dabei sehr wichtig.
 - Das zu lösende Problem kann aus der Realität wie auch aus den Anforderungen eines Kunden kommen.
- ❖ **Informatiker ist Optimist:**
 - Das Modell beschreibt die Wirklichkeit (das Problem).
- ❖ **Ziel der Informatik ist das Verständnis von künstlichen Systemen.**
- ❖ **4 Arten von Informatik-Systemen:**
 - Berechnung von Funktionen, Prozessüberwachung, Eingebettete Systeme, Adaptive Systeme

In der Vorlesung verwendete Literatur

- ❖ **Broy: Informatik: Eine grundlegende Einführung**
 - Band 1: Programmierung und Rechnerstrukturen
Springer Verlag, ISBN 3-540-63234-4
- ❖ **Goos: Vorlesungen über Informatik**
 - Band 1: Grundlagen und funktionales Programmieren
Springer Verlag, ISBN 3-540-62880-0
 - Band 2: Objektorientiertes Programmieren und Algorithmen
Springer Verlag, ISBN 3-540-64340-0
- ❖ **Bernd Brügge:**
 - Vorlesungen „Einführung in die Informatik I und II“, Studienjahr 2003/04,
- ❖ **Weiterführende Literatur:**
 - Bernd Bruegge, Allen Dutoit:
Object-Oriented Software Engineering: Using UML, Patterns and Java, 3rd
Edition. Prentice Hall, NJ, 2009; ISBN: 0-13-606125-7