

Regionale Lehrerfortbildung



Informationstechnologie
Modellierung

Ulrich Fischer
Wilhelm-Sattler-Realschule Schweinfurt

Der Modellbegriff

Was ist eigentlich ein Modell?

Überlegen Sie zunächst einmal, welche Begriffe Ihnen spontan zu Modell oder Modellierung einfallen.

Bei mir waren es folgende: Atommodell, Automodell, Teilchenmodell, Modell des Lichtstrahls, Modellflugzeug, Modelleisenbahn, Skulptur, Fotomodell (Heidi Klum, ...), Gipsmodell, ...



Bemüht man ein Lexikon, so fällt einem als erstes auf, das nicht eine, sondern eine Reihe unterschiedlicher Definitionen aufgeführt sind.

Einige Definitionen sollen uns den Modellbegriff näher bringen (nach Prof. Dr. Martin Glinz, Uni Zürich):

- Konkretes oder gedankliches **Abbild eines vorhandenen Gebildes** (ein Automodell, ein Modell einer Volkswirtschaft)
- Konkretes oder gedankliches **Vorbild für ein zu schaffendes Gebilde** (ein Modell eines geplanten Gebäudes, Musterbriefe für verschiedene Anlässe)
- Modell eines physikalischen Phänomens (Modell des Lichtstrahls, ...)
- Person oder Sache als **Gegenstand einer künstlerischen Abbildung** („Sie stand ihm Modell“)
- **Person, welche Modeschöpfungen** vorführt („Sie arbeitet als Fotomodell“)
- **Typ von etwas** („Dieses Auto ist das neueste Modell“)

Der wissenschaftliche Modellbegriff geht in folgende Richtung:

Konkretes oder gedankliches **Abbild eines vorhandenen Gebildes** oder **Vorbildes für ein zu schaffendes Gebilde**. Modelle sind **Abbildungen und Konstruktionen** der Realität.

Wozu brauchen wir überhaupt Modelle?

- zum **Verständnis** eines Gebildes
- um über ein Gebilde **kommunizieren zu können**
- als gedankliches **Hilfsmittel zum Gestalten, Bewerten oder Kritisieren**
- für ein geplantes Gebilde
- zur Durchführung von **Experimenten**, die am Original nicht durchgeführt werden sollen, können oder dürfen
- Aufstellen oder Prüfen von **Hypothesen** über beobachtete oder postulierte Phänomene

Modelle sind insbesondere dann notwendig, wenn das modellierte Original

- **nicht beobachtbar** ist
- **zu groß oder zu klein** ist
- zu **komplex** ist
- **nicht zur Verfügung steht**
- noch **nicht existiert**

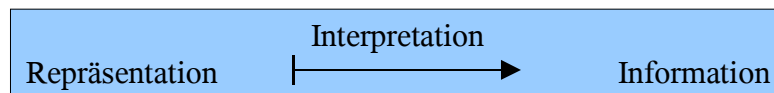
oder wenn die Arbeit am Original

- zu **gefährlich**,

- zu teuer,
- verboten,
- nicht möglich ist.

Informationen und Daten bedürfen für ihre Weiterleitung bzw. Weiterverarbeitung einer Repräsentation, also einer Darstellungsform, mit der Empfänger etwas anfangen und somit interpretieren kann. Man kann sagen:

Einer **Repräsentation** wird durch eine **Interpretation** eine **Information** zugeordnet:

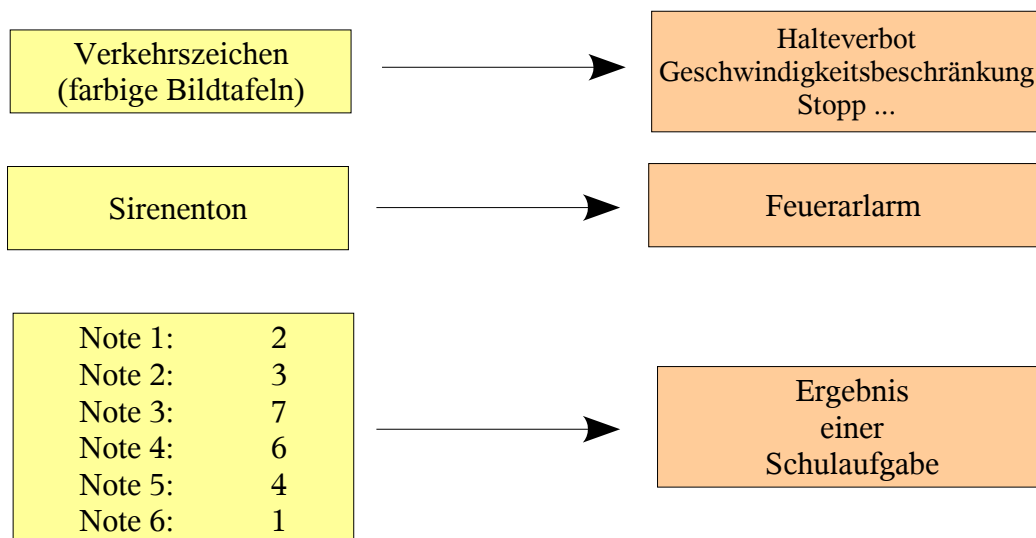


Eine Information zusammen mit ihrer Repräsentation bezeichnet man als Objekt. Somit sind Repräsentationen ebenfalls Träger von Informationen.

Beispiele:

konkretes Phänomen der realen Welt

abstrakte Idee, Konzept



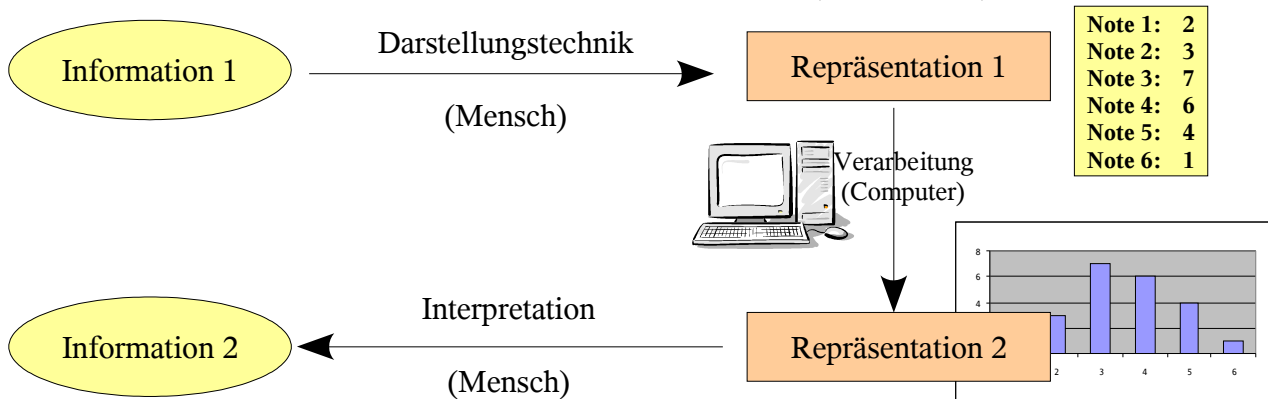
Dass die drei Begriffe Repräsentation, Interpretation und Information untrennbar miteinander verbunden sind, ist offensichtlich: Jede Information bedarf einer Repräsentation, sonst kann sie weder mitgeteilt noch gespeichert noch sonst wie verarbeitet werden. Umgekehrt ist jede Repräsentation ohne zugeordnete Information im wahrsten Sinn des Wortes sinnlos. Verarbeitungsprozesse operieren auf Repräsentationen.

Wie wichtig eine gut gewählte Repräsentation ist, wird am Beispiel der Zahlen besonders deutlich. Auf Grund der abstrakten Idee der Zahlen gilt z.B. 'einhundertzwölf' plus 'zweitausendsiebenundachtzig' ergibt 'zweitausendeinhundertneunundneunzig'. Es wäre völlig aussichtslos, sich solche Fakten abstrakt merken zu wollen. Erst auf der Basis geschickt gewählter Repräsentationen, hier dem Dezimalsystem, kann man mit solchen Problemen sinnvoll umgehen.

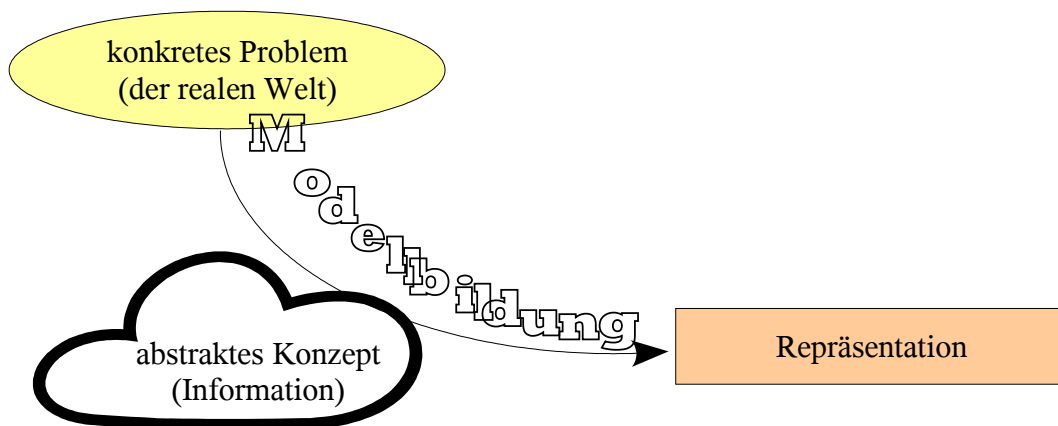
$$\begin{array}{r}
 112 \\
 + 2087 \\
 \hline
 2199
 \end{array}$$

Die Geschichte der Naturwissenschaften belegt eindrucksvoll die Bedeutung guter Repräsentationen. So war z.B. das römische Zahlensystem vor ca. 2000 Jahren ein wahrer Hemmschuh für herausragende naturwissenschaftliche Leistungen.

Informationen müssen vor ihrer Weiterverarbeitung dargestellt (repräsentiert) werden:



Die gewählten Repräsentationen stellen stets ein **Modell der Wirklichkeit** dar.



Beispiele:

konkretes Problem	abstraktes Konzept	Repräsentation
Wie sieht ein Festkörper in seinem Inneren aus?	Kugeln mit Abstoßungs- bzw. Anziehungskräften	
Wie lang ist die Seitenlänge eines Quadrats mit einer Fläche von 2 cm²?	Seite * Seite = 2 cm²	$a^2 = 2$ $a = \sqrt{2}$
Das zweite Beispiel zeigt, dass Verarbeitungsvorschriften, also auch Programme, ebenfalls Modelle darstellen.		

Für den Modellbegriff wollen wir uns eine Definition zu eigen machen, die einerseits leicht nachvollziehbar, andererseits aber auch allgemein genug ist, um den Modellbegriff für die Informatik abzugrenzen:

Ein Modell ist eine vereinfachte Beschreibung eines realen, geplanten oder gedachten Systems, das jedoch nicht alle Eigenschaften des Vorbilds aufweisen muss.
Das Erstellen eines Abbildes eines gedachten oder realen Objekts bzw. Systems bezeichnet man als Modellierung.

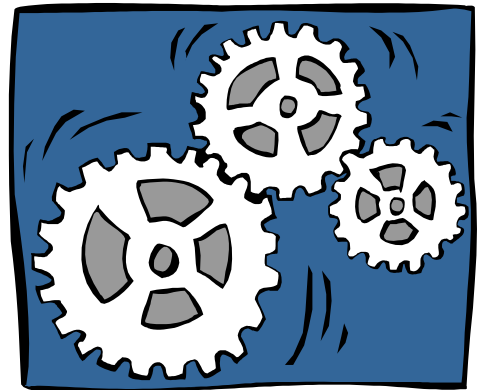
Der Systembegriff

Damit müssen wir einen weiteren Begriff – den des Systems – abklären:

Als System bezeichnen wir eine Menge von Komponenten (Systembestandteilen), die miteinander in Beziehung bzw. Wechselwirkung stehen.
Ein System ist stets ein Ausschnitt aus der realen Welt.

Beispielsweise kann man den **Motor eines Autos** als System bezeichnen oder auch das ganze Auto. Dabei muss es sich aber nicht unbedingt um materielle Gegenstände oder konkrete Dinge handeln (Gleissystem der Deutschen Bahn), auch unser **Sternen- oder Planetensystem**, unser **Zahlensystem**, das **Zahlungs- bzw. Währungssystem** oder **Anwenderprogramme am Computer** sind Beispiele für Systeme.

Betrachten wir Modelle nur als Abbilder oder Vorbilder, so wird jedes Modell und jedes modellierte Original als Menge von **Objekten mit Attributen und Methoden (Operationen)** beschrieben. Dabei gehört jedes Objekt einer bestimmten **Klasse** an. Die Methoden sind in der Regel die Folgen von bestimmten **Ereignissen**.



- **Ein Objekt (Exemplar, Instanz) ist ein individuelles Exemplar von Dingen, Personen oder Begriffen.** Es besitzt eine Objektidentität.
- **Klassen** fassen Objekte mit gleichen Attributen und Methoden zusammen. Klassenattribute beschreiben die Eigenschaften von Klassen. Klassenoperationen sind den Klassen zugeordnet.
- **Attribute (Eigenschaften)** beschreiben die Merkmale eines Objekts. Die aktuellen Werte, die ein Objekt gerade besitzt heißen Attributwerte (Eigenschaftswerte).
- **Methoden (Operationen)** beschreiben das Verhalten eines Objekts, d.h. die Dienstleistungen, die es seiner Umwelt oder sich selbst zur Verfügung stellt. Methoden kommunizieren mit der Umwelt über Ein-/Ausgabeparameter.
- Durch **Botschaften (Ereignisse)** kommunizieren Objekte und Klassen untereinander.



Natürlich müssen wir uns auch überlegen, wie wir Modelle beschreiben und welche **Notationsform** wir wählen. Eine Notation ist ein **System von Zeichen** zur Darstellung eines Modells.

Dabei müssen wir uns einig sein, welcher Zeichenvorrat zur Verfügung steht, welche Regeln für die Bildung von Zeichenstrukturen (**Syntax**) bestehen und welche Bedeutungen die verwendeten Zeichen besitzen (**Semantik**). Notationen werden in diesem Zusammenhang auch **Sprachen** genannt.

Ziel eines Modells ist

- Analyse eines Systems unter **festen Randbedingungen**
- Analyse eines Systems bei der Eingabe von **unterschiedlichen Szenarien** (Parameter)
- **Optimierung** eines bestehenden Modells (zeitliches Verhalten, Ressourcennutzung, ...)

Beispiele:

	Bankschalter	Anzahl der wartenden Kunden Anzahl der geöffneten Schalter Länge des Beratungsgespräches	
	Telefonauskunft	Anzahl der Serviceplätze Anzahl der aktiven Mitarbeiter Auskunftsrate	
	Aufzug (Lift)	Anzahl der Wartenden pro Stockwerk Position des Liftes	

Simulation

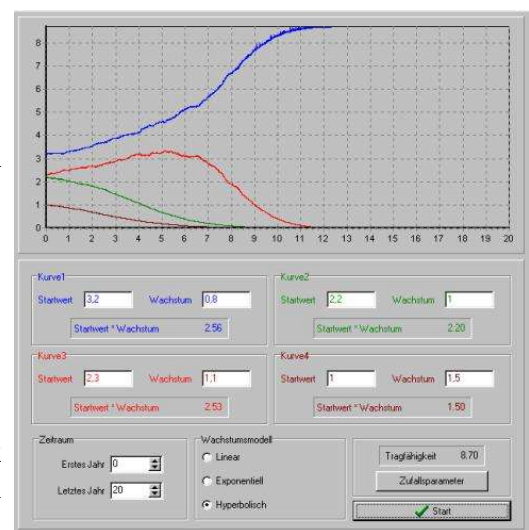
Vorwiegend bei dynamischen Systemen wird mit dem Modell eine Simulation erzeugt.

In der Informatik ist die Simulation die Nachbildung eines realen Vorgangs oder Ablaufs mit Hilfe des Computers.

Dabei hängt die Aussagekraft der Simulation sehr stark von eventuell getroffenen Vereinfachungen, Annahme von Idealzuständen und Voraussetzungen ab.

Hubwieser (1996/2000) betont die Modellierung als einen Themenbereich der Informatik, der eine „immense“ Bedeutung für die Allgemeinbildung hat und verweist auf stützende Aussagen von Brauer und Brauer (1973), Koerber und Peters (1988) und die GI-Empfehlungen (Gesellschaft für Informatik) von 1993 (Schulz-Zander et al.). Ebenso wie Schubert betrachtet er den Prozess der Modellbildung und Simulation nicht nur als Lerninhalt, sondern als ein durchgängiges Prinzip der Unterrichtsgestaltung.

Betrachtet man den Lehrplan sowohl für das Fach Informatik als auch Informationstechnologie in der bayerischen Realschule näher, so kann man erkennen, dass Modelle und **Modellierung** ab der ersten Unterrichtsstunde bis zum Ende der 10. Jahrgangsstufe tatsächlich als **durchgängiges Unterrichtsprinzip** vorzufinden sind – nicht nur an den Lehrplanstellen, an denen der Modellbegriff ausdrücklich erwähnt wird. **Wichtig ist, dass die Komplexität der Modelle der Altersstufe der Schüler gerecht wird.**



Simulation eines chaotischen Zustandes mit Hilfe eines Tabellenkalkulationsprogramms

Ein Modell gehorcht festen Regeln. Bei der Chaosforschung untersucht man, welche minimalen Änderungen eines Faktors die Änderungen eines Ergebnisses bewirken können. "Der Flügelschlag eines Schmetterlings in Spanien kann einen Tornado in Texas bewirken."

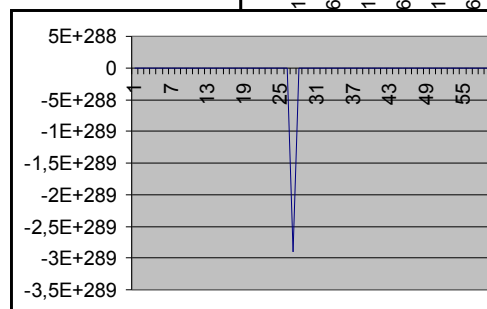
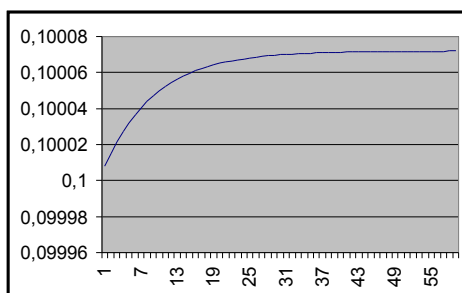
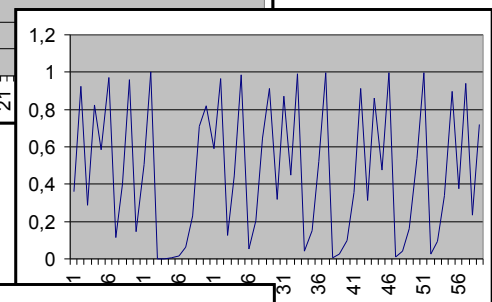
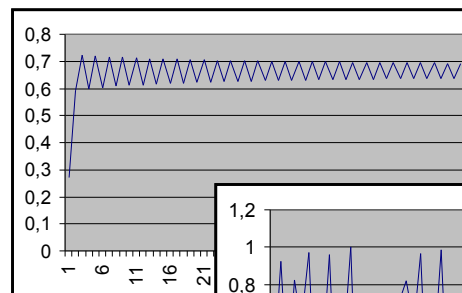
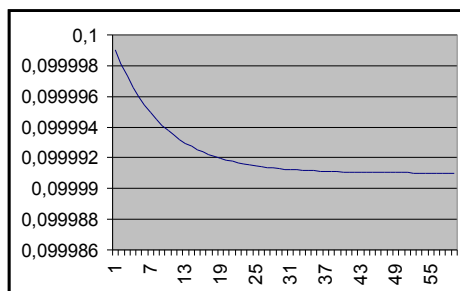
Mit Hilfe einer rekursiven Formel wird in einem Tabellenkalkulationsprogramm eine Wertetabelle erzeugt.

	A	B	C
1	x	$a \cdot x - a \cdot x^2$	Parameter a
2	0,1	0,099999	1,1111
3	0,099999	0,099998111	
4	0,099998111	0,099997321	

Der Zellinhalt von C2 wird in B3 übernommen (Rekursion). Die Zellinhalte von C3 und B2 werden bis Zeile 60 nach unten kopiert. Der Bereich B2:B60 muss nun markiert und grafisch dargestellt werden. Es empfiehlt sich eine Liniengrafik.

Bei Änderung des Parameters a auf den Wert 1,1112 (Erhöhung des Parameters um ein Zehntausendstel !!) ergibt sich die zweite Grafik.

Experimente mit Parametern $a = 2,8; 3,0; 3,2 \dots 4,2$ ergeben interessante Zustände eines Modells, die so nicht voraussehbar waren. (Grenzen der Modellierung)



Gefahren bei der Modellbildung

Bei der Modellbildung und Simulation muss man sich grundsätzlich mit folgenden **vier Fehlerquellen** auseinandersetzen:

- **Vereinfachungsfehler**
- **Verfahrensfehler** bei der Auswahl und Anwendung eines Rechenverfahrens
- **Rechenfehler**, da Computer aufgrund ihrer zwangsweise endlichen Speicherkapazität nur mit endlich vielen Stellen rechnen können, also zum Runden gezwungen sind. Dabei können sich die evtl. auftretenden Rundungsfehler, ohne dass man dies hinreichend kontrollieren könnte, sowohl gegenseitig verstärken als auch gegenseitig vermindern.
- **Interpretationsfehler** beim Schließen vom errechneten zukünftigen Modellzustand auf den realen zukünftigen Systemzustand, denn hier kann es sich aufgrund des Zusammenwirkens der o. g. Fehler - die sich gegenseitig vermindern, aber auch potenzieren können - in der Regel allenfalls um eine "spekulative Vorhersage" handeln.

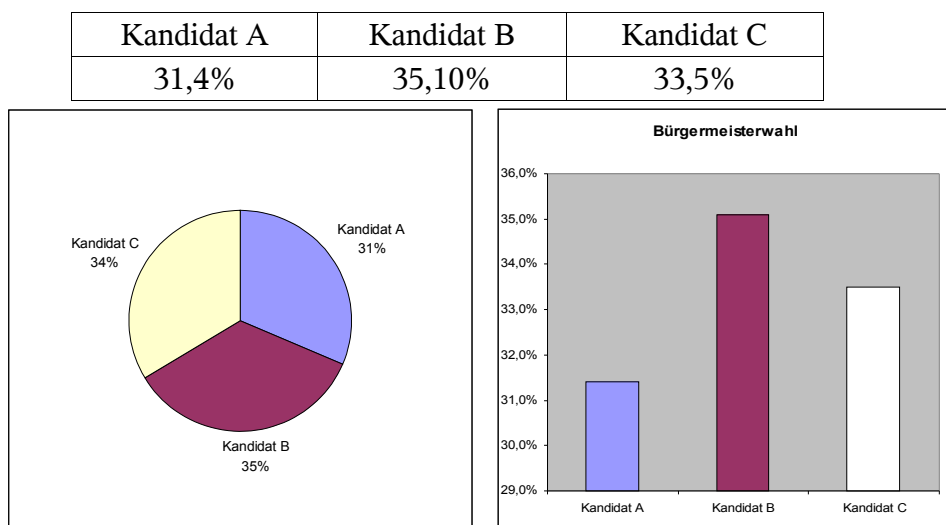
Außerdem besteht aber auch die **Gefahr**, Modelle ohne Rücksicht auf die beim Modellbildungsprozess begangenen Fehler mit der Realität gleichzusetzen und Simulationsergebnisse als zukünftige Realität statt als evtl. sehr fehlerhafte Rechenergebnisse anzusehen.

Eine in diesem Sinne falsche Computer-Gläubigkeit kann sachgerechte Beurteilungen und Entscheidungen eher gefährden, statt sie zu unterstützen.

Modellbildungsprozesse und Simulationsergebnisse haben keine Beweiskraft, sondern allenfalls eine gewisse Plausibilität, deren Stärke durch möglichst unvoreingenommene und sorgfältige Fehlerabschätzungen geprüft werden muss. Dabei muss auch das Ausmaß der Unsicherheit offen gelegt werden.

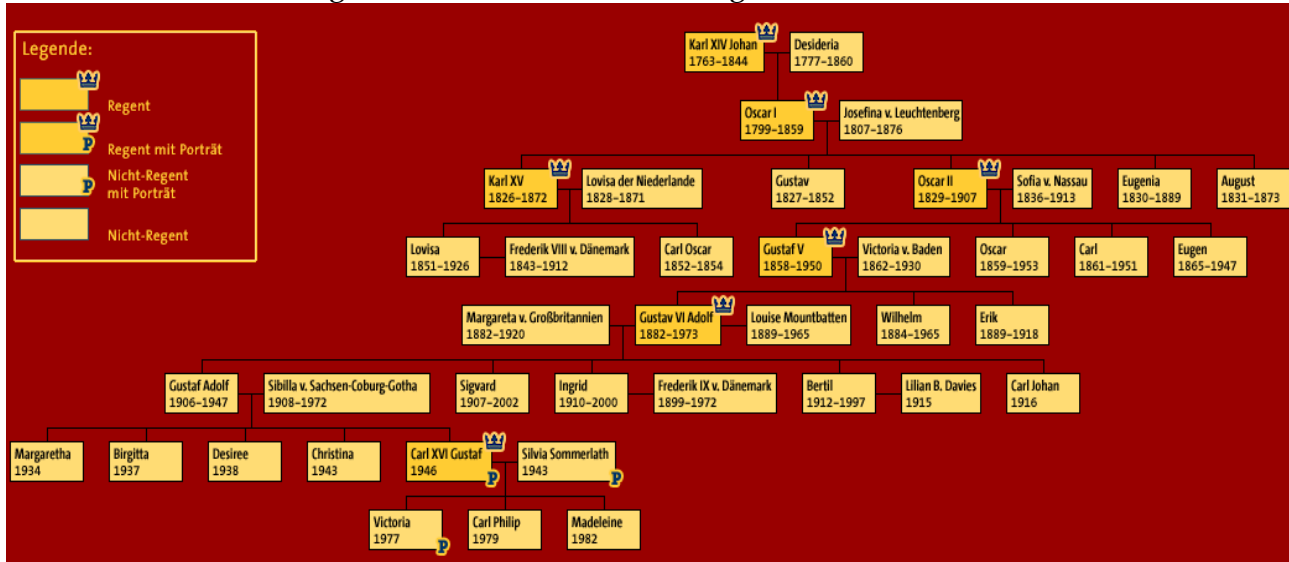
Dies muss den Schülerinnen und Schülern unbedingt bewusst gemacht werden!

Grundsätzlich müssen wir uns außerdem bewusst machen, dass das Erzeugen eines Modells keine wertfreie Tätigkeit ist. Jedes Modell stellt das Original aus einer bestimmten Sicht heraus dar und **verändert** damit die Wahrnehmung des Originals. So kann der Erzeuger eines Modells die Sichtweise des Betrachters in eine von ihm gewollte Richtung lenken. Als Beispiel soll das Ergebnis einer Bürgermeisterwahl dienen, bei der 3 Kandidaten gegeneinander angetreten sind. Die Ergebnisse lassen sich durchaus unterschiedlich darstellen – beurteilen Sie selbst:



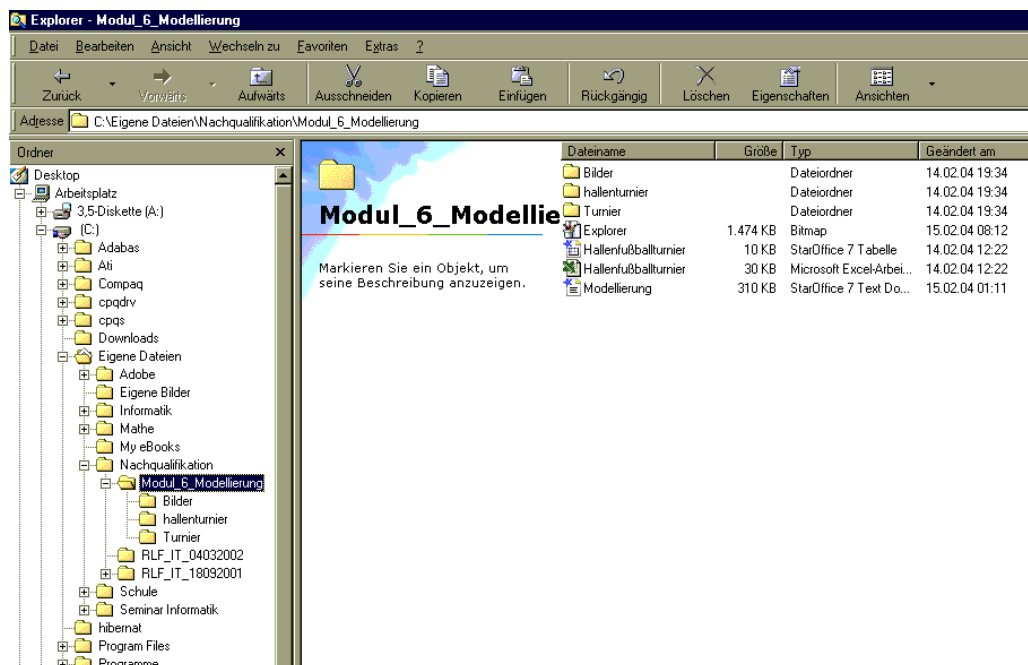
Modellierung

Damit sind wir bereits mitten in der Modellbildung. Ein nächstes Beispiel – wohl eher aus dem Erfahrungsbereich von Erwachsenen – ist Ihnen allen sicher bekannt: der Stammbaum einer Familie oder wie hier dargestellt des schwedischen Königshauses.



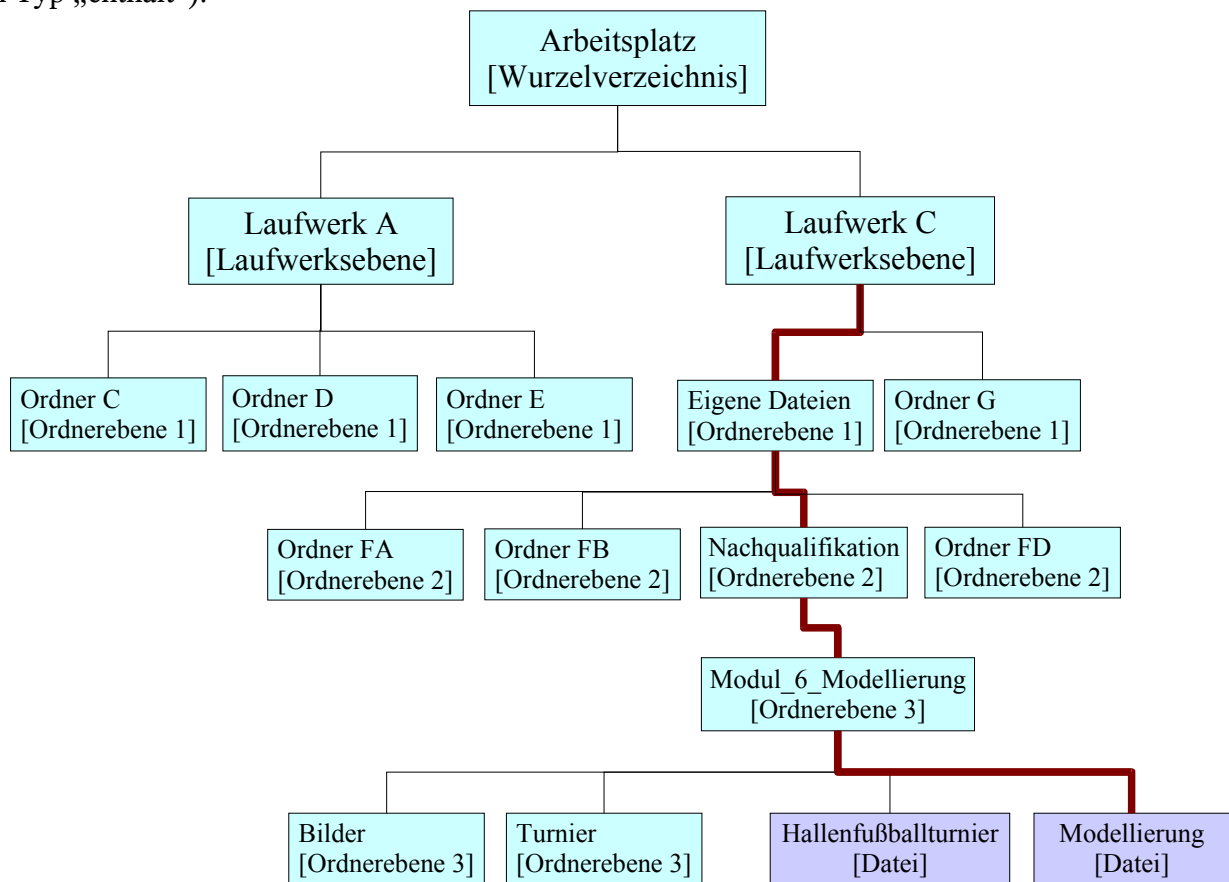
Quelle: <http://www.ndrtv.de/tv/royalty/koenigs/schweden/stammbaum.html>

Dieses Modell zeigt eine Struktur, die wir bei informatischen System häufiger antreffen: es wird eine Hierarchie aufgezeigt. Hierarchische Modelle finden wir auch z.B. bei Datenbanken oder noch näher liegend: Das Ordner- und Dateiensystem auf einem Computer. Unter dem Betriebssystem 'Windows' wird es mit Hilfe des Explorers modelliert.



Im linken Teil des Explorers erkennt man die hierarchische Struktur, in diesem Fall mit dem 'Arbeitsplatz' als Wurzelverzeichnis. Die nachfolgende Ebene beinhaltet die Laufwerke, danach die

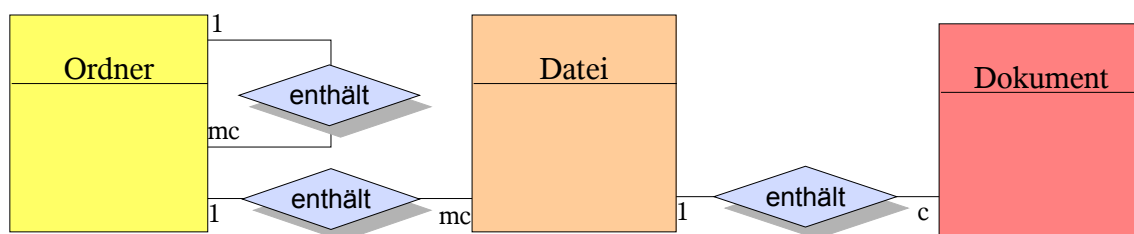
erste Ordnerebene, die zweite usw. Im rechten Teil ist der Inhalt des 'geöffneten' Ordners zu sehen: weitere Unterordner und Dateien. In einer anderen Modellierungsansicht – hier ist die Hierarchie besser zu erkennen – lässt sich dieses System folgendermaßen darstellen (alle Beziehungen sind vom Typ „enthält“):



Hervorgehoben in obiger Abbildung ist der Pfad von Laufwerk C bis zu der Datei 'Modellierung'. Die Schreibweise für diesen Pfad im Betriebssystem 'Windows' ist folgende:

[C:\Eigene Dateien\Nachqualifikation\Modul_6_Modellierung\Modellierung](#)

Das gesamte Ordner- und Dateiensystem eines Computers lässt sich auch mit Hilfe eines **Klassen-diagramms** modellieren:



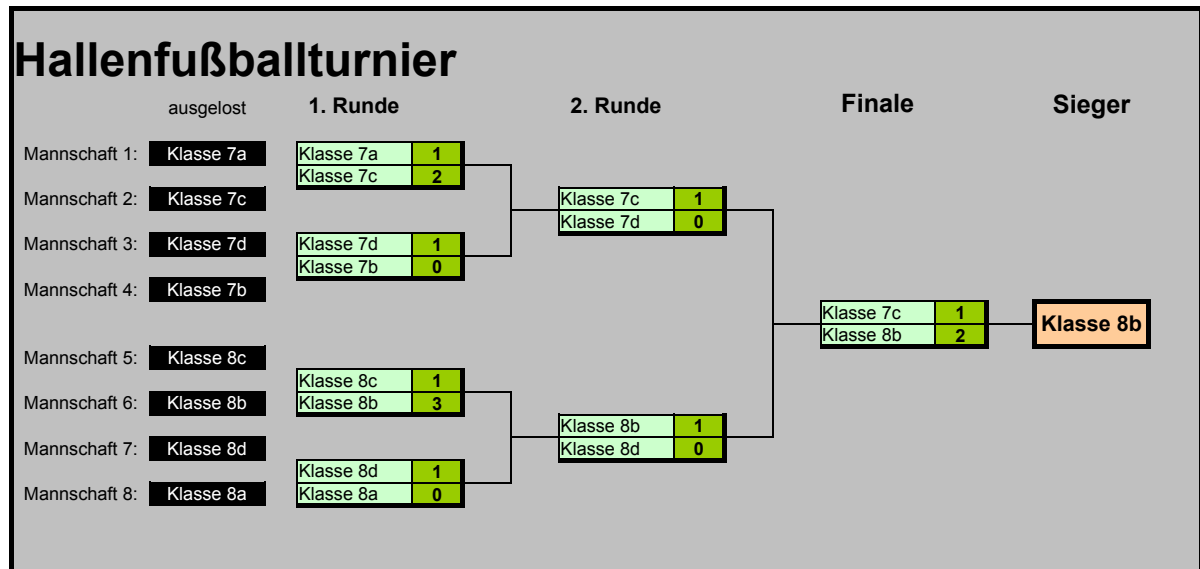
Die Attribute und Methoden der einzelnen Klassen sind hier weggelassen worden, da sie für den Zweck, der mit diesem Modell verfolgt wird, unerheblich sind.

Dies gilt für alle Modelle: es werden nur die für den jeweiligen Zweck notwendigen Teile ausge-

führt. Unerhebliches wird einfach weggelassen (siehe auch Definition von Modell!).

Ein weiteres, für den Unterricht sicher gut geeignetes Beispiel mit hierarchischer Struktur:

Wir wollen ein Hallenfußballturnier modellieren, wobei die letzten 16 Teilnehmer im K.O.-Prinzip den Turnierbesten ermitteln. Dies können wir durch ein Tabellenkalkulationsprogramm realisieren.



Modellierungsarten und Sichtweisen

Die Erstellung eines Modells erfolgt in folgenden Schritten:

- Analyse des Systems
- Entwurf
- Implementierung
- Realisierung

Je nach Art des existierenden Systems muss bei der Modellierung ein unterschiedlicher Ansatz bzw. eine unterschiedliche Sichtweise verwendet werden.

Nach H. Balzerts 'Lehrbuch der Softwaretechnik' werden folgende Grundkonzepte bzw. Sichten zu einer möglichen Problemlösung in der Informatik dargestellt:

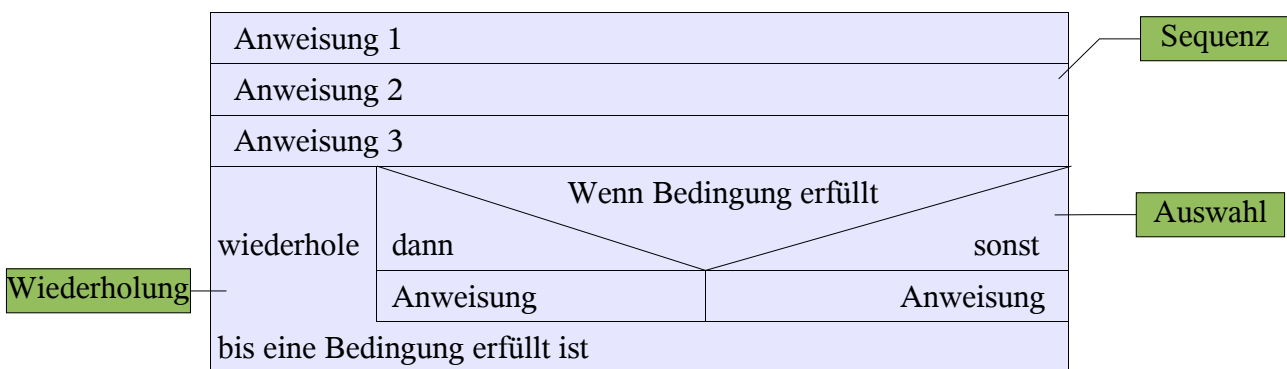
Konzepte und Sichten	Notationsformen
Algorithmische Sicht	Pseudocode, Kontrollstrukturen, Struktogramm, Programmablaufplan
Objektorientierte Sicht	Klassenstrukturen, Klassendiagramm, Objektdiagramm
Datenorientierte Sicht	Entitäten und Beziehungen, Datenstrukturen
Regelbasierte Sicht	Wenn-Dann-Regeln, Entscheidungstabelle
Zustandsorientierte Sicht	Zustandsdiagramm, nebenläufige Strukturen
Funktionale Sicht	Datenflussdiagramm, Funktionsbaum
Szenebasierte Sicht	Interaktionsdiagramm

Je nach Sichtweise verwenden wir das Modellierungsverfahren mit der Notationsform, die dem Modell und dem damit verbundenen Zweck am ehesten gerecht wird. Zur Darstellung existiert eine Sprache und damit verbundene Notationsform, die den Anforderungen in der Informatik besonders gerecht wird: **UML (Unified Modeling Language)**. Die Unified Modeling Language ist eine Sprache und Notation zur Spezifikation, Konstruktion, Visualisierung und Dokumentation von Modellen für Softwaresysteme.

Algorithmisches Problemlösen

Bis Mitte der neunziger Jahre wurde in der Informatik der bayerischen Realschule das **algorithmische bzw. prozedurale Problemlösen** (geht zurück auf die siebziger Jahre) bevorzugt, um nicht zu sagen, es war nahezu die einzige Sicht, aus der die Lösung eines Problems in der Schule angegangen wurde. Dabei konnte das Problem durch **Strukturieren** in überschaubare Teile zerlegt und mit Hilfe einfacher **Algorithmen** gelöst werden.

Die Erstellung der **Struktur** ist hier der zentrale Punkt der Modellbildung. Dies geschieht mit Hilfe der drei Grundstrukturen **Sequenz – Auswahl – Wiederholung**. Das Struktogramm bietet sich für die Modellbildung solcher Probleme an. Das folgende allgemein gehaltene Struktogramm zeigt zunächst, wie die Grundstrukturen dargestellt werden.



Beispiel:

Es soll gezeigt werden, dass jede beliebige Ausgangszahl durch wiederholte Anwendung des folgenden Algorithmus zur Zahl 123 wird: Aus der ursprünglichen Zahl wird eine neue Zahl gebildet, bei der an erster Stelle die Anzahl der geraden Ziffern, an zweiter Stelle die Anzahl der ungeraden Ziffern und an dritter Stelle die Gesamtzahl der Ziffern dieser ursprünglichen Zahl steht.

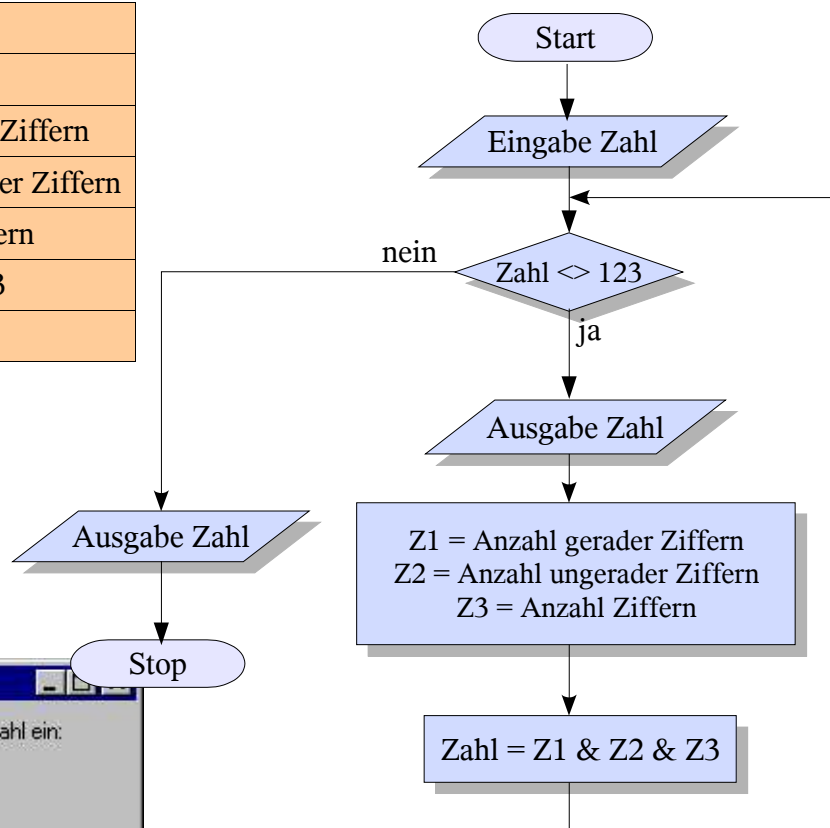
An eine solche Aufgabe geht man sinnvoller Weise zunächst mit einem sog. Schreibtischtest und probiert das Ganze mit verschiedenen Ausgangszahlen:

7432	86672	2234567891111
224	415	5813
303	123	134
123		123

Dadurch hat man sich einen genaueren Überblick über den Lösungsalgorithmus verschafft. Mit einem Struktogramm bzw. Flussdiagramm lässt sich die Aufgabe modellieren.

Struktogramm und Datenflussdiagramm zeigen zwei leicht unterschiedliche Lösungen zum beschriebenen Algorithmus. Welche?

Eingabe Zahl	
Solange Zahl \leq 123	
wiederhole	Z1 = Anzahl gerader Ziffern
	Z2 = Anzahl ungerader Ziffern
	Z3 = Anzahl der Ziffern
	Zahl = Z1 & Z2 & Z3
	Ausgabe Zahl

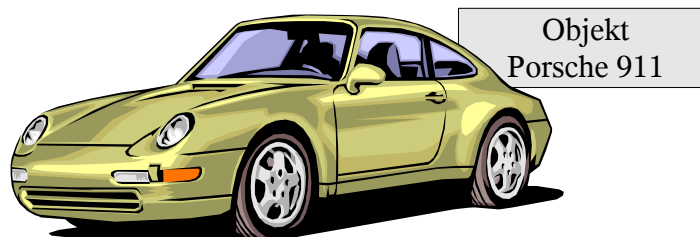


Die Simulation der Aufgabenstellung kann danach auch noch mit einer entsprechenden Programmierung in einer Programmiersprache erfolgen, wobei hierbei jedoch zusätzliche Schwierigkeiten auftauchen. Man muss zur Zerlegung der Zahl in ihre Ziffern diese in einen String (Text) umwandeln, weshalb die Besprechung von Datentypen unbedingt vorausgehen muss.

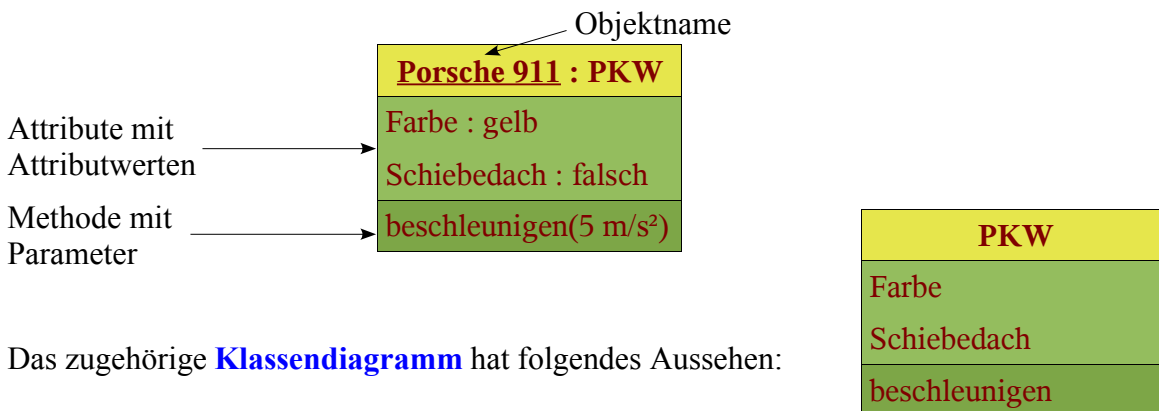
Überlegen Sie bitte an dieser Stelle, ob die unterrichtliche Umsetzung auch mit einem Tabellenkalkulationsprogramm möglich, wenn ja, ob sie sinnvoll ist.

Objektorientiertes Modell

Mit Hilfe der Beschreibungssprache UML (Unified Modelling Language) werden die Objekte mit ihren Eigenschaften und Methoden beschrieben.



Schreibweise als **Objektdiagramm**:



Das zugehörige **Klassendiagramm** hat folgendes Aussehen:

Schreibweise als **Punktnotation**: *Porsche 911[PKW].Farbe = gelb* (Attribut)
Porsche 911[PKW].beschleunigen(5m/s²) (Methode)

Auch die eine Methode auslösenden Ereignisse werden durch Punktnotation beschrieben:

Porsche 911[PKW].Gaspedal_treten = wahr (Ereignis)

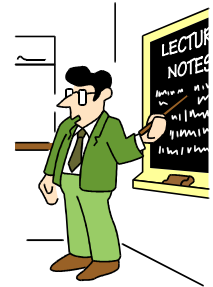
Vergleichen wir die Vorgänge beim Kauf eines Neuwagens: Wir entscheiden uns zunächst für ein Fahrzeug einer bestimmten **Klasse (Fahrzeugtyp)**. Anschließend bestimmen wir **Farbe und Ausstattung (Attribute mit den Attributwerten)**. Geliefert bekommen wir ein ganz **individuelles Fahrzeug (Objekt/Instanz)** mit den gewünschten Ausstattungsmerkmalen. Dieses Fahrzeug verfügt über gewisse **Fähigkeiten (Methoden)**, mit denen es auf bestimmte **Ereignisse** reagiert; z.B. reagiert es auf das Ereignis 'Gaspedal treten' mit der Methode 'beschleunigen'.

Datenorientiertes Modell (ERM = Entity Relationship Modell)

Um eine Datenbank aufzubauen, muss ein Abbild der relevanten Welt definiert werden, ein **Datenmodell**. Die zum Zweck der Verarbeitung gebildete Informationen heißen Daten. Der Teilausschnitt der Welt, der zu analysieren ist, wird Miniwelt genannt. Das Datenmodell soll die Miniwelt mit ihrem Bedeutungsgehalt abbilden. Dazu müssen alle relevanten Objekte und Beziehungen zwischen den Objekten in einem Datenmodell erfasst und beschrieben werden.

Betrachtet man die "Miniwelt Schule", erkennt man gewisse **Objekte** wie das Fach Englisch, einen Schüler namens Meier, einen Lehrer namens Müller, einen Schulleiter namens Direx, einen Klassenleiter namens Huber, einen Raum mit der Nummer 112. Für den Begriff Objekt wird im Zusammenhang mit der Modellbildung auch der Begriff **Entität** oder Instanz verwendet. Um das Beispiel nicht zu komplex zu gestalten, soll eine Beschränkung auf die Objekte **Schüler, Lehrer und Klassen** vorgenommen werden. Im objektorientierten Ansatz 'besitzt' ein Objekt neben Eigenschaften auch Methoden (Operationen). Eine **Klasse** definiert die Eigenschaften und Methoden gleichartiger Objekte. Zwischen diesen Objekten bestehen **Beziehungen**, die bestimmte Abläufe (Prozesse) oder Abhängigkeiten in der Miniwelt darstellen. Die Unterrichtsbelegung stellt die Beziehung zwischen einem Lehrer und den von ihm unterrichteten Klassen dar. Um zum Datenmodell zu gelangen, stellt man Objekte und Beziehungen mit gleichartigen **Attributen** zu Typen zusammen. Somit ergeben sich für das gewählte Beispiel

- die Objekttypen **Lehrer** und **Klassen**
- die Beziehung **unterrichtet**



Alle Lehrer bilden die Objektmenge **Lehrer**, alle Klassen die Objektmenge **Klasse** und alle Unterrichtsbelegungen die Beziehungsmenge **unterrichtet**. Jeder Objekt und Beziehungstyp definiert bestimmte **Eigenschaften (Merkmale, Attribute)**.

- Die Attribute des Objekttyps Lehrer sind:
Lehrernummer, Nachname, Vorname, Amtsbezeichnung, Fächerkombination.
- Für den Objekttyp Klasse lassen sich z. B. folgende Eigenschaften feststellen: *Klassenbezeichnung, Ausbildungsrichtung, Klassenleiter L-Nr, Klassensprecher S-Nr.*
- Dem Beziehungstyp **unterrichtet** lassen sich z. B. folgende Merkmale zuordnen:
L-Nr, Klassenbez, Fach

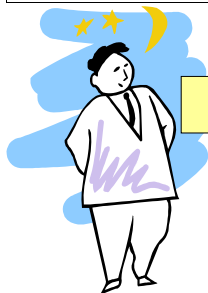
Für eine solche Datenmodellierung gibt es eine Reihe von Techniken, die je nach den Eigenheiten des Systems Vor- und Nachteile haben. Im Datenbankbereich ist die Entity-Relationship-Modellierung die mit Abstand häufigste und erfolgreichste Technik. Die Elemente des Entity-Relationship-Modells sind **Entitäten, Beziehungen zwischen Entitäten und Attribute** zur Charakterisierung von Entitäten.

Bei relationalen Datenbanken werden die Daten in Tabellen dargestellt. Alle Tabellen zusammen bilden die **Datenbasis**. Zwischen den einzelnen Tabellen einer Datenbasis können diverse Beziehungen bestehen.

Assoziation: Eine Assoziation legt fest, wie viel Datensätze einer Tabelle 2 zu einem Datensatz der Tabelle 1 gehören können.

Mögliche Assoziationstypen:

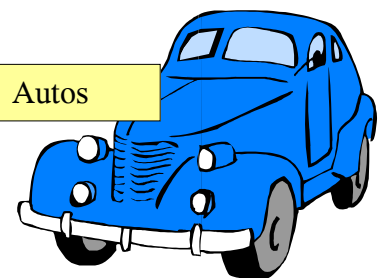
Abkürzung	Assoziationstyp	Anzahl der Datensätze in Tabelle 2
1	einfache Assoziation	genau ein Datensatz (1)
c	konditionelle Assoziation	kein oder genau ein Datensatz (0/1)
m oder n	multiple Assoziation	mindestens ein Datensatz (>0)
mc oder nc	multipl-konditionelle Assoziation	beliebig viele Datensatz (≥0)



Personen

besitzen

Autos



Am Beispiel 'Personen besitzen Autos' betrachten wir die verschiedenen Assoziationstypen:

Typ 1 Jede Person besitzt genau ein Auto.



Jede Person besitzt **genau ein** Auto bzw. jedes Auto **gehört genau** einer Person.

➤ Typ c: Eine Person kann **ein oder kein** Auto besitzen.



Jede Person besitzt **kein oder genau ein** Auto bzw. jedes Auto **gehört genau** einer Person.

➤ Typ m: Jede Person besitzt **mindestens ein** Auto.



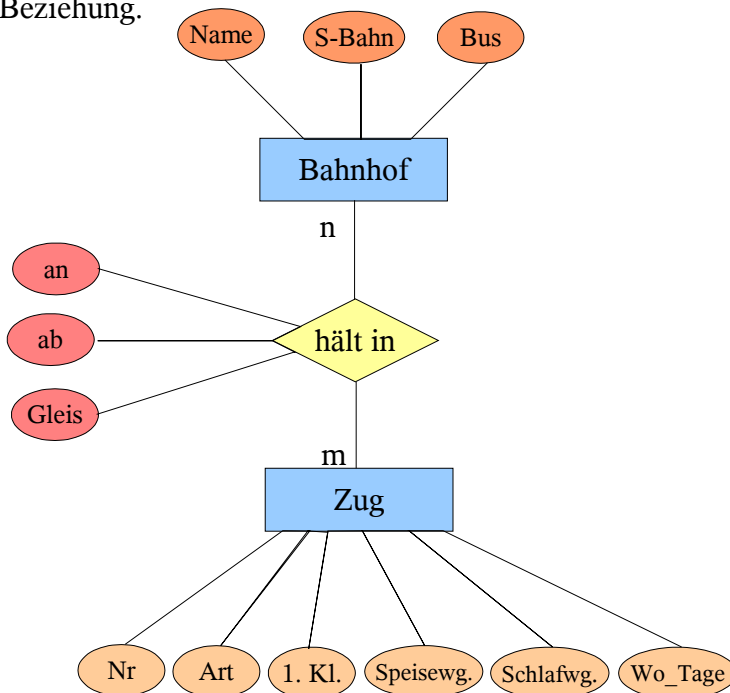
Jede Person besitzt **mindestens ein** Auto bzw. jedes Auto **gehört genau** einer Person.

➤ Typ mc: Jede Person kann **beliebig viele** Autos besitzen (auch keines!).



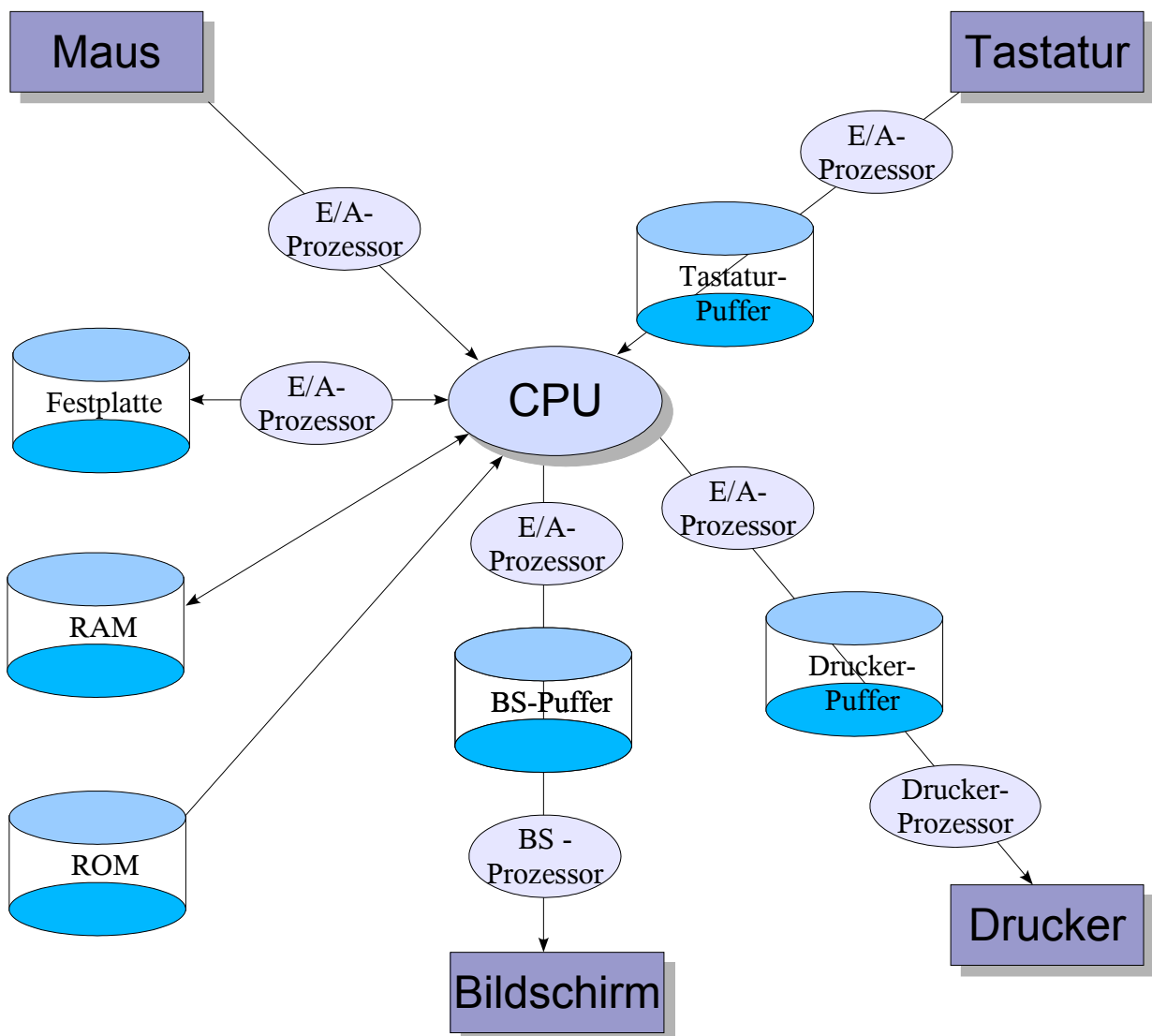
Jede Person kann **beliebig viele** Autos besitzen (0, 1 oder mehr) bzw. jedes Auto **gehört genau** einer Person.

Auch nachfolgendes Beispiel, die Modellierung eines Fahrplanausschnittes (Hubwieser 'Didaktik der Informatik') zeigt eine entsprechende Datenrepräsentationen. Da in einem Bahnhof mehrere Züge halten und die meisten Züge in mehr als einem Bahnhof, handelt es sich hier um eine m:n-Beziehung.



Funktionales Modell

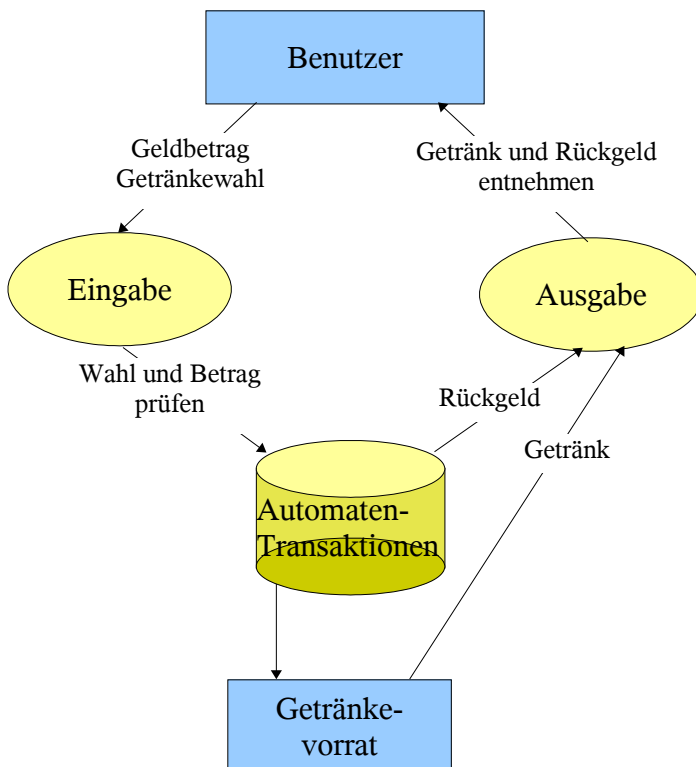
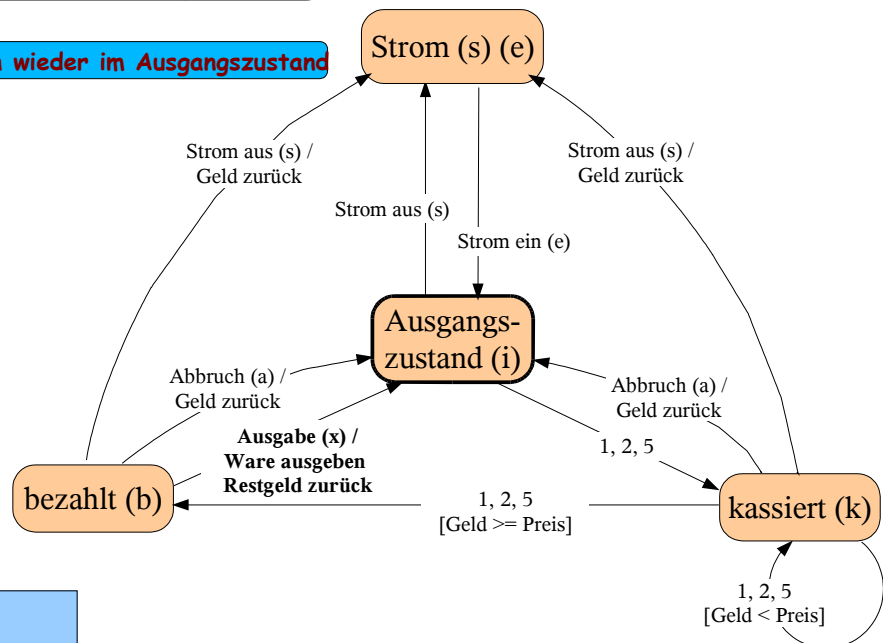
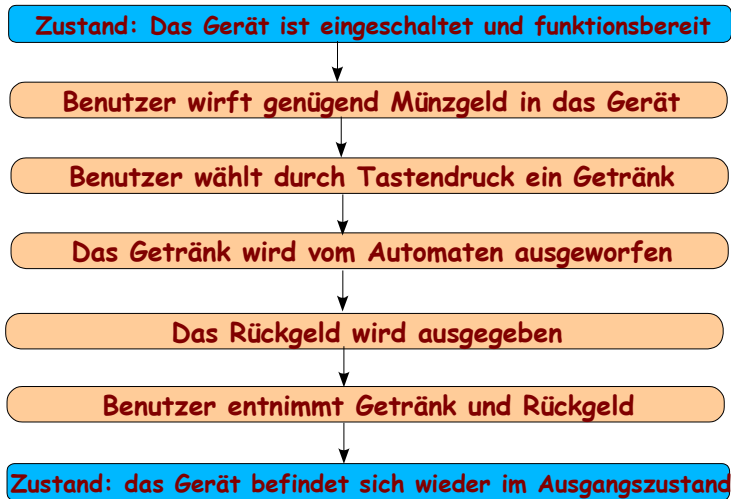
Es zeigt die Datenflüsse zwischen den Komponenten eines Systems, dabei werden die dynamischen Zusammenhänge dargestellt. Beispielsweise lässt sich das Zusammenwirken der Komponenten und damit die Funktion eines Einzelplatzrechners wie in nachfolgendem Funktionsdiagramm darstellen (Hubwieser, Didaktik der Informatik). Dabei sind informationsverarbeitende Prozesse durch Ellipsen, dauerhafte Datenobjekte durch Zylinder und die Ein- bzw. Ausgabegeräte durch Rechtecke dargestellt. Die Pfeile in den Datenleitungen zeigen die Kommunikationswege.



Zustands- bzw. ablaufsorientiertes Modell

Der zeitliche Ablauf eines Systems wird beschrieben.

Dies kann durch ein Aktivitätsdiagramm oder ein Zustands-Übergangs-Diagramm erfolgen. Als Beispiel soll hierfür die Modellierung eines Getränkeautomaten dienen. Das Aktivitätsdiagramm ist bewusst einfach gehalten, so kann man die Modellierung auch in den unteren Klassenstufen realisieren. Dagegen ist das Zustands-Übergangs-Diagramm sehr abstrakt und eignet sich eigentlich besser für die gymnasiale Oberstufe.



Da das Beispiel 'Getränkeautomat' explizit im Lehrplan genannt wird, wollen wir uns hierfür auch das funktionale Modell ansehen.

Zur Simulation der Funktion eines solchen Automaten kann man auch hierfür ein selbstgeschriebenes Programm einsetzen.

Eine von der Programmierung etwas einfachere Version (Bezahlungsbetrag wird in ein Textfeld eingegeben) findet man im Bay. Realschulnetz unter Materialien für das Fach Informatik

(<http://www.realschule.bayern.de/>
Lehrer – Materialien - Informatik).



Getränkeautomat

Soft - Drinks

Status: EIN

Ein Power Aus

Geldeingabe: 10 Ct 50 Ct 1 €

Geldbetrag: 0,00 €

Getränkeauswahl: Cola 0,80 € Limo 0,80 €

Getränkewahl: keine

Ausgabe Abbruch

Getränkeausgabe: Cola

Geldrückgabe: 0,20 €

Eine Simulation mittels Tabellenkalkulationsprogramm ist ebenfalls möglich. In der komplexeren Version sind hierfür auch einige Programmierkenntnisse nötig, da die Makrosprache bei den Office-Programmen von Microsoft VBA (Visual Basic for Applications) ist.

Getränkeautomat
einfache Version

Folgende Getränke stehen zur Auswahl:

Cola	1,20 €
Limo	1,00 €
Wasser	0,80 €

Bitte wählen Sie:

Zahlbetrag

Rückgeld

Ausgabe:

Eingaben nur in grüne Felder!

Getränk und Rückgeld entnehmen