uwe KASTENS · hans KLEINE BÜNING

Modellierung

Grundlagen und formale Methoden

5., aktualisierte Auflage

Auf plus.hanser-fachbuch.de: Übungs- und Klausuraufgaben mit Lösungen · Folien für Dozierende

HANSER

Kastens / Kleine Büning

Modellierung

Uwe Kastens

Hans Kleine Büning

Modellierung

Grundlagen und formale Methoden

5., aktualisierte Auflage

HANSER

Autoren:

Prof. Dr. Uwe Kastens, UKastens@t-online.de
Prof. Dr. Hans Kleine Büning, hkb48@gmx.de

Universität Paderborn
Fakultät für Elektrotechnik, Informatik und Mathematik

MIX
Papier aus verantwortungsvollen Quellen
FSC® C083411

Bibliografische Information der Deutschen Nationalbibliothek:
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über
http://dnb.d-nb.de abrufbar.

Internet: www.hanser-fachbuch.de

Lektorat: Dipl.-Ing. Natalia Silakova-Herzberg
Herstellung: Anne Kurth
Covergestaltung: Max Kostopoulos
Coverkonzept: Marc Müller-Bremer, www.rebranding.de, München
Titelbild:© stock.adobe.com/infografx
Druck und Bindung: CPI books GmbH, Leck
Printed in Germany

Print-ISBN 978-3-446-46942-6
E-Book-ISBN 978-3-446-46956-3

Vorwort

Das Modellieren ist eine für das Fach Informatik typische Arbeitsmethode, die in allen Gebieten des Faches angewandt wird. Aufgaben, Probleme oder Strukturen werden untersucht und als Ganzes oder in Teilaspekten beschrieben, bevor sie durch den Entwurf von Software, Algorithmen, Daten oder Hardware gelöst bzw. implementiert werden. Mit der Modellierung einer Aufgabe zeigt man, ob und wie sie verstanden wurde. Das Modell ist Voraussetzung und Maßstab für die Lösungen und liefert meist auch den Schlüssel für einen systematischen Entwurf. Als Ausdrucksmittel für die Modellierung steht ein breites Spektrum von Kalkülen und Notationen zur Verfügung. Sie sind spezifisch für unterschiedliche Arten von Aufgaben und Problemen. Deshalb werden in den verschiedenen Gebieten der Informatik unterschiedliche Modellierungsmethoden eingesetzt. In den entwurfsorientierten Gebieten, wie Software-Technik und Hardware-Entwurf, ist die Bedeutung der Modellierung und die Vielfalt der Methoden besonders stark ausgeprägt.

Mit diesem Buch soll eine Übersicht über die wichtigsten Kalküle der Informatik und ein grundlegendes Verständnis für jeden der vorgestellten Kalküle vermittelt werden. Die Leser sollen an vielen praktischen Beispielen lernen, die Kalküle zur Modellierung anzuwenden, und dabei Erfahrungen im formalen Beschreiben erwerben. Sie sollen den Nutzen von klaren und präzisen Beschreibungen erkennen. Das angebotene Spektrum von Kalkülen ist als Grundausstattung zu verstehen, die noch wesentlich vertieft und verbreitert werden kann. Deshalb wird hier von jedem Kalkül nur der innere methodische Kern präsentiert und auf die Vorstellung von Erweiterungen und weniger grundlegenden Kalkülen verzichtet.

Das vorliegende Buch ist als Lehrbuch zur Vorlesung *Modellierung* entstanden, mit der die oben genannten Ziele verfolgt werden. Die Vorlesung wird an der Universität Paderborn als eine einsemestrige, vierstündige Pflichtveranstaltung für Studierende der Informatik und der Wirtschaftsinformatik angeboten. Zur Vorlesung gehören praktische Übungen in Form betreuter Kleingruppenarbeit, selbstständiger Hausarbeiten und zentraler Präsentation von Lösungen. Die Übungsaufgaben dieses Buches stammen im Wesentlichen aus diesem Übungsmaterial.

Die beiden Autoren dieses Buches haben die Vorlesung in Abstimmung mit ihren Kollegen inhaltlich konzipiert und seit 1998 im zweijährigen Wechsel gehalten. Wegen ihres unterschiedlichen fachlichen Hintergrundes setzen sie auch verschiedene Schwerpunkte bei der Vermittlung von Modellierungsaspekten: Uwe Kastens arbeitet im Gebiet Programmiersprachen und Übersetzer und ist stark in Themen der Software-Technik verwurzelt. Bei der Modellierung betont er die Anwendung der Kalküle, den Nutzen formaler Beschreibungen und die Trennung von Aufgaben und Lösungen. Hans Kleine Büning arbeitet im Gebiet Logik und Wissensbasierte Systeme. Er betont stärker die theoretischen Grundlagen der Kalküle und setzt Schwerpunkte in den Logik-Kalkülen. Die Autoren haben in enger Kooperation versucht, die unterschiedlichen Schwerpunkte und Herangehensweisen einander ergänzend in das Buch und die Vorlesung einzubringen. An diesen Diskussionen hat sich auch der Kollege Hauenschild nachhaltig beteiligt, der im Jahr 2004 die Vorlesung übernommen hat.

Nach der Einführung werden in den Kapiteln 2 bis 7 die Kalküle vorgestellt. Wir haben die Reihenfolge so gewählt, dass Vorgriffe möglichst vermieden werden können. Zu jedem Kalkül werden zunächst die Grundbegriffe eingeführt und dann typische Modellierungstechniken an möglichst anschaulichen Beispielen gezeigt. Jedes Kapitel schließt mit einer Sammlung von Übungsaufgaben. Als durchgängige Aufgabe wird die Modellierung von Aspekten eines Getränkeautomaten in Übungen eines jeden Kapitels aufgegriffen. In Kapitel 8 werden an zwei Fallstudien alle Kalküle im Zusammenhang gezeigt. Einige bibliografische Hinweise finden sich am Ende des Buches.

Im Kapitel 1 wird der Modellbegriff eingeführt, so wie er im Alltag und in der Informatik verwendet wird. Die Tätigkeit des Modellierens wird charakterisiert und ihre Notwendigkeit begründet. Am Schluss zeigen wir eine einfache formale Modellierung im Vorgriff auf später vorgestellte Kalküle.

Das Kapitel 2 führt Mengen als Modellierungskalkül ein: Einfache und zusammengesetzte Mengen definieren die Wertebereiche von Objekten und Eigenschaften des Modells. Abstrakte Konzepte wie kartesisches Produkt, Potenzmenge, Vereinigung, Folgen, Relationen und Funktionen werden zur Strukturierung der Wertemengen eingesetzt. Diese Begriffe sind Grundlagen für jede Art formaler Beschreibung.

Fast alle formalen Kalküle definieren eine Notation für Formeln, in denen Operanden mit Operationen verknüpft werden. Im Kapitel 3 werden deshalb Terme als abstrakte Grundlage von Formeln eingesetzt, die erst im Anwendungskontext spezielle Bedeutung bekommen. Notationen sowie allgemein gültige Begriffe wie Substitution, Umformung nach Regeln und Unifikation werden hier universell für Terme definiert. Im zweiten Teil des Kapitels werden diese Begriffe zu abstrakten und konkreten Algebren ausgebaut. Damit können dann Kalküle und Datenstrukturen algebraisch definiert werden. Wegen der universellen Bedeutung von Termen haben wir sie in einem separaten Kapitel definiert – und nicht als Teil der Prädikatenlogik, wo sie einen angestammten Platz haben.

Kapitel 4 führt die beiden klassischen Gebiete der Logik ein: Aussagenlogik und Prädikatenlogik erster Stufe. Wir beschränken uns auf die für Anfänger wichtigsten Grundlagen der Kalküle: Syntax und Semantik, Umgang mit logischen Operatoren und Quantoren und einfache Normalformen sowie Transformationen. Insbesondere Kalküle und Verfahren des automatischen Beweisens überlassen wir einer späteren Vertiefung.

Der Kalkül der Graphen wird in Kapitel 5 behandelt. Er eignet sich besonders gut zum Modellieren. Er ist leicht zu verstehen, da er nur auf Relationen basiert, und er ist anschaulich, da er sehr suggestive Visualisierungen hat. Graphen sind außerordentlich vielfältig einsetzbar, da Objekte und Beziehungen zwischen Objekten in so vielen unterschiedlichen Bedeutungen beim Modellieren vorkommen. Es werden hier nur die grundlegenden Eigenschaften von Graphen vorgestellt. Auf weiterführende Begriffe der Graphentheorie wird verzichtet und stattdessen die Vielfalt der Einsatzgebiete und Modellierungstechniken ausgebreitet.

In Kapitel 6 führen wir zwei grundlegende Kalküle ein, die sich besonders zur Modellierung struktureller Eigenschaften eignen. Kontextfreie Grammatiken werden als Regelsystem vorgestellt, mit dem Baumstrukturen definiert werden können, um damit geschachtelte Strukturen zu modellieren. Die Definition von Sprachen als Menge von Symbolfolgen, die sonst im Vordergrund steht, spielt hier nur eine Nebenrolle. Im zweiten Teil wird das Entity-Relation-Ship-Modell eingeführt. Damit formuliert man Regeln, nach denen Systeme in Mengen gleichartiger Objekte mit bestimmten Eigenschaften gegliedert sind und zwischen denen Relationen bestehen. Das ER-Modell ist Grundlage sowohl der Schemata objektorientierter Datenbanken als auch der Spezifikation von Strukturen und Beziehungen in Software-Systemen, z. B. mit der Modellierungssprache UML.

In Kapitel 7 führen wir zwei grundlegende Kalküle ein, mit denen Abläufe modelliert werden können: endliche Automaten und Petri-Netze. Sie werden eingesetzt, um das dynamische Verhalten von Systemen zu beschreiben. Man gibt die Zustände an, die das System einnehmen kann, und beschreibt, unter welchen Bedingungen es aus einem Zustand in einen anderen übergehen kann. Beide Kalküle sind mit recht einfachen Regeln definiert und haben sehr anschauliche grafische Repräsentationen. Mit endlichen Automaten werden meist sequentielle, mit Petri-Netzen nebenläufige Prozesse modelliert. Beide dienen als Grundlage komplexer Kalküle (z. B. endliche Automaten für Statecharts), die hier nicht vorgestellt werden.

In Kapitel 8 präsentieren wir die Modellierungen von zwei Aufgaben vollständig im Zusammenhang: Die Auftragsabwicklung in einer Autowerkstatt und Regeln eines Gesellschaftsspieles. Verschiedene Aspekte des Gegenstandsbereiches werden mit jeweils passenden Kalkülen modelliert. Außerdem werden einzelne Aspekte zum Vergleich mit verschiedenen Kalkülen beschrieben. Dabei wird das Vorgehen ausführlich erläutert. Wir zeigen damit, dass es wichtig ist, den Gegenstandsbereich sinnvoll zu zerlegen und für Teilaspekte geeignete Kalküle zu wählen.

Zu diesem Buchprojekt haben die Autoren nachhaltige Unterstützung erfahren: Kollege Hauenschild hat im Jahr 2004 unser Material zur Vorbereitung der Vorlesung übernommen. Er hat uns viele nützliche Hinweise gegeben und zur Behebung einiger Inkonsistenzen beigetragen. Wissenschaftliche Mitarbeiter haben die Übungen zur Vorlesung betreut. Stellvertretend seien hier Dinh Khoi Le, Jochen Kreimer, Theodor Lettmann und Carsten Schmidt genannt. Sie haben sich viele Aufgaben fantasievoll ausgedacht, das Vorlesungsmaterial kritisch hinterfragt und Fehler darin korrigiert. Sigrid Gundelach hat den größten Teil des Manuskriptes geschrieben, spröde Formeln gewissenhaft übertragen und sich engagiert in das manchmal widerspenstige Textsystem eingearbeitet. Michael Thies hat uns geholfen, mit dem System den gewünschten Drucksatz zu produzieren. Sie alle haben zum Gelingen des Buches beigetragen. Wir danken ihnen sehr dafür.

Uwe Kastens, Hans Kleine Büning

Vorlesungs- und Übungsmaterial für Studierende und Dozierende

Über die Webseite plus.hanser-fachbuch.de erhalten Sie kostenlos Zugang zu umfangreichem Vorlesungs- und Übungsmaterial für Studierende und Dozierende. Den Zugangscode finden Sie auf Seite 1 Ihres Buches.

Das Zusatzmaterial umfasst:

- Lernziele für Semester
- 14 Blätter mit Übungsaufgaben
- 14 Blätter mit Lösungen zu den Übungsaufgaben
- Klausur-Beispielaufgaben
- Lösungen der Klausur-Beispielaufgaben
- Umformen mit Gesetzen der booleschen Algebra
- Beispiel für Umformungen
- Schrittweise Konstruktion und Verifikation
- Folien zur Vorlesung „Modellierung“ als begleitendes Lehr- und Lernmaterial für Studierende bzw. Dozierende.

Die komplette Aufstellung der Folien finden Sie auf Seite 301.

Inhalt

1 Einführung

Das Anfertigen von Modellen ist ein wichtiger Schritt in den frühen Phasen der Herstellung ganz unterschiedlicher Objekte: Ein Architekt erstellt ein dreidimensionales Modell für den Neubau eines besonders exponierten Gebäudes. Daran werden mit dem Auftraggeber die äußere Gestaltung und die Integration des Gebäudes in das Stadtbild diskutiert. Um die Raumaufteilung zu planen, die Sicherheit zu prüfen und die Durchführung des Baus zu steuern, werden andere Modelle erstellt, wie Bauzeichnungen und Berechnungen der Statik.

Solch ein Vorgehen ist auch typisch für Entwicklungsprozesse in der Informatik: Aufgaben und Lösungsvorschläge werden mit Modellen beschrieben, bevor mit der Herstellung einer Lösung in Software oder Hardware begonnen wird. An den Modellen können Auftraggeber und Entwickler prüfen, ob die Aufgabenstellung richtig verstanden wurde, welche Eigenschaften die vorgeschlagene Lösung haben wird und ob diese akzeptabel sind.

Jede Branche hat Techniken der Modellbildung, die sich für ihre Aufgaben besonders gut eignen, z. B. Gebäudemodelle, Bauzeichnungen und Berechnungen der Statik für die Herstellung von Gebäuden oder Schaltpläne für das Anfertigen elektrischer Anlagen. Produkte der Informatik wirken meist durch ihre Funktionen – seltener durch ihr Aussehen. Deshalb überwiegen hier abstrakte Modelle, die mit formalen Kalkülen erstellt werden. Aufgaben von unterschiedlicher Art aus einem sehr breiten Spektrum werden mit Informatikmethoden gelöst. Deshalb werden in der Informatik sehr viele verschiedene Kalküle eingesetzt, die jeweils unterschiedliche Aspekte der Aufgaben möglichst gut modellieren können. Informatiker müssen sie als ihr „Handwerkszeug" beherrschen, um für ihre Aufgaben nützliche Modelle mit geeigneten Kalkülen anzufertigen.

1.1 Einführendes Beispiel

Wir wollen einen ersten Einblick in das Modellieren der Informatik geben. Als Beispiel betrachten wir folgende Denksportaufgabe:

Beispiel 1.1: Flussüberquerung

Ein Mann steht mit einem Wolf, einer Ziege und einem Kohlkopf am linken Ufer eines Flusses, den er überqueren will. Er hat ein Boot, das gerade groß genug ist, ihn und ein weiteres Objekt zu transportieren, sodass er immer nur eines der drei mit sich hinübernehmen kann. Falls der Mann allerdings den Wolf mit der Ziege oder die Ziege mit dem Kohlkopf unbewacht an einem Ufer zurücklässt, wird einer gefressen werden.

Ist es möglich, den Fluss zu überqueren, ohne dass die Ziege oder der Kohlkopf gefressen wird?

Solche Aufgaben sind deswegen reizvoll, weil man eine Weile daran herumknobelt und dann – wie mit einem Geistesblitz – die Lösung findet. Mit solch einem Vorgehen geht man allerdings das Risiko ein, die Lösung nicht zu finden. Bei ernsthaften Aufgaben wäre das fatal. Deshalb zeigen wir hier ein systematisches Vorgehen unter Einsatz von Informatik-Kalkülen – auch wenn dabei vielleicht der Spaß an der Knobelei verloren geht.

Zunächst klären wir, welche Aufgabe gelöst werden soll: Die Antwort auf die Frage am Ende des Textes kann „ja“ oder „nein“ lauten. In jedem Fall müssen wir eine Begründung angeben, entweder indem wir einen Plan für eine sichere Flussüberquerung entwickeln oder erklären, weshalb es einen solchen nicht gibt.

Dann beginnen wir die Modellierung, indem wir die Beschreibung der Aufgabe auf die unbedingt notwendigen Objekte, Eigenschaften und Aktionen reduzieren: Als Objekte kommen in Frage: Mann, Wolf, Ziege, Kohlkopf, Fluss, linkes Ufer und rechtes Ufer, Boot. Es ist wichtig, als Eigenschaft zu beschreiben, welche der Objekte Wolf, Ziege und Kohlkopf sich gemeinsam an jedem der beiden Ufer aufhalten; denn damit wird entschieden, ob einer von ihnen gefressen werden kann. Als Aktionen benötigen wir das Transportieren jeweils eines der drei Objekte von einem zum anderen Ufer oder Leerfahrten jeweils in beide Richtungen.

Wenn es eine Lösung der Aufgabe gibt, dann kann man sie als eine Folge solcher Fahrten über den Fluss beschreiben. Jede Fahrt verändert den Zustand des Systems. Die Zustände des Systems modellieren wir, indem wir angeben, welche der Objekte Mann, Wolf, Ziege und Kohlkopf sich an welchem Ufer befinden. Formal ist der Zustand ein Paar von Mengen (l, r), mit $l, r \subseteq \{M, W, Z, K\}$, wobei wir Abkürzungen für die vier Objekte verwenden. Da keines der Objekte verschwinden und keines hinzukommen kann, muss für jedes Paar (l, r) gelten

$$l \cap r = \emptyset \qquad \text{und} \qquad l \cup r = \{M, W, Z, K\}$$

Nun können wir auch formal beschreiben, wann ein Zustand (l, r) unzulässig ist, weil die Ziege oder der Kohlkopf gefressen werden könnte:

$$M \notin l \text{ und } (\{W, Z\} \subseteq l \text{ oder } \{Z, K\} \subseteq l) \text{ oder}$$
$$M \notin r \text{ und } (\{W, Z\} \subseteq r \text{ oder } \{Z, K\} \subseteq r)$$

Wenn z. B. im Zustand $(\{M, W, Z, K\}, \emptyset)$ die Ziege an das rechte Ufer transportiert wird, geht das System in den Zustand $(\{W, K\}, \{M, Z\})$ über. Beide Zustände sind gemäß obiger Bedingung zulässig. Wir müssen noch begründen, dass es nicht nötig ist, den Zustand während der Überfahrt zu modellieren: Ist während der Überfahrt das gerade verlassene Ufer unzulässig, so ist es auch unzulässig im Zustand, der die Ankunft modelliert. Entsprechendes gilt für den Zustand vor der Abfahrt und das zu erreichende Ufer.

Die Zustandsübergänge kann man sehr anschaulich grafisch beschreiben. Abb. 1.1 zeigt den Übergang, den wir oben als Beispiel beschrieben haben. Die Rechtecke geben die Zustände mit dem Paar von Mengen an. Der beschriftete Pfeil verbindet die Zustände vor und nach dem Transport.

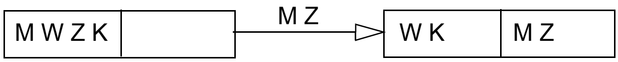


Abbildung 1.1: Ein Zustandsübergang

Wir haben nun das Prinzip des Modells entworfen. Wir brauchen es nur noch vollständig auszufüllen, indem wir alle zulässigen Zustände und Übergänge angeben. Abb. 1.2 zeigt das vollständige Modell. Es sind auch einige unzulässige Zustände angegeben, um zu zeigen, welche Entscheidungen nicht getroffen werden dürfen. Die Übergänge zwischen zulässigen Zuständen sind natürlich jeweils in beiden Richtungen möglich, obwohl es im Sinne der Aufgabe nicht sinnvoll ist, in einen früheren Zustand zurück zu rudern. In Abb. 1.2 können wir nun leicht ablesen, dass es mehrere Lösungen für die Transportaufgabe gibt:

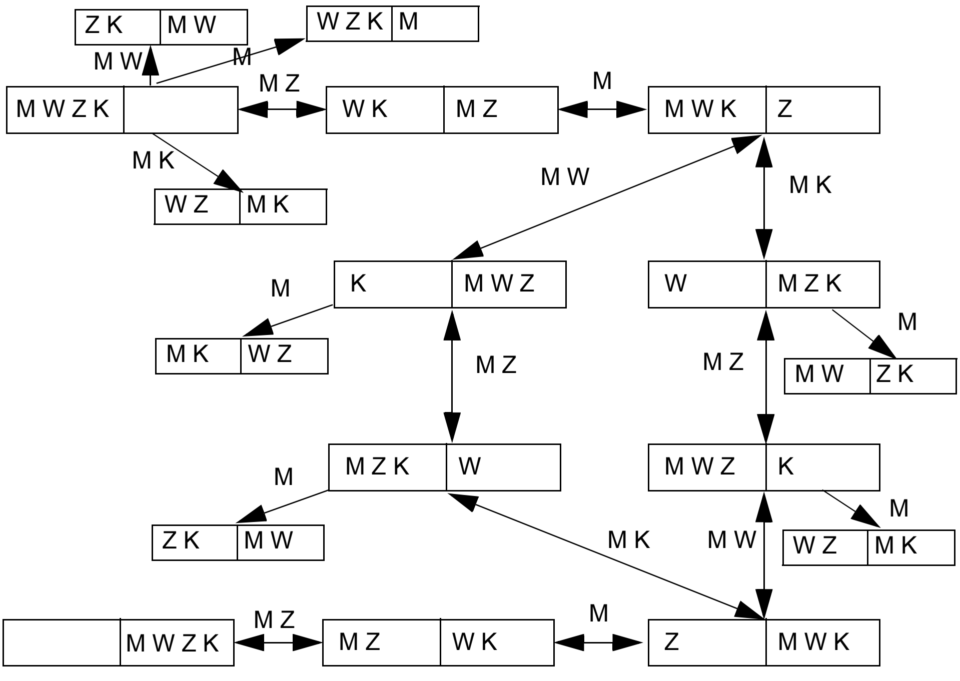


Abbildung 1.2: Modellierung der Flussüberquerung

Jeder Weg vom Startzustand zum Zielzustand beschreibt eine Lösung. Es gibt zwei verschiedene, die mit 7 Überfahrten auskommen. Alle anderen Transportfolgen enthalten überflüssige Überfahrten.

Für die Modellierung unserer Aufgabe haben wir den Kalkül der endlichen Automaten angewandt. Er eignet sich besonders gut, wenn Abläufe in Systemen mit Übergängen zwischen verschiedenen Zuständen beschrieben werden sollen. Der Kalkül wird in Abschnitt 7.1.2 vorgestellt. Die grafische Veranschaulichung des Automaten in Abb. 1.2 benutzt den Kalkül der gerichteten Graphen, der in Abschnitt 5.1 eingeführt wird. Zu Beginn der Modellierung haben wir die Zustände des Systems durch Paare von Mengen beschrieben. Solche Abstraktionen führen wir in Kapitel 2 ein.

In den folgenden Kapiteln dieses Buches werden wir die Kalküle jeweils an typischen Beispielen vorstellen. Eine Modellierungsaufgabe werden wir dabei durchgängig immer wieder aufgreifen: Die Bedienung eines Getränkeautomaten soll modelliert werden. Das Gerät soll Getränke wie Kaffee, Tee und Kakao gegen Bezahlung mit Münzen abgeben. Man soll Varianten der Getränke wählen können, z. B. mit oder ohne Milch oder Zucker. Die Modellierung soll berücksichtigen, dass im Gerät nur begrenzte Vorräte für die Zubereitung der Getränke untergebracht werden können. Die Beschreibung der Aufgabe ist hier absichtlich unscharf gehalten. Es entspricht dem Vorgehen in der Realität, dass die Beschreibung der Aufgabe erst im Zuge der Modellierung präzisiert wird.

1.2 Modellbegriff

Der Begriff des *Modells* ist vom lateinischen Wort *modulus* für Maß, Maßstab abgeleitet. Er wird in vielen verschiedenen Zusammenhängen mit recht unterschiedlichen Bedeutungen verwendet. Er kann das Abbild eines vorhandenen Originals bezeichnen, z. B. ein Schiffsmodell, oder das Vorbild für ein herzustellendes Original, z. B. ein Gebäudemodell oder ein Vorbild für Maler oder Bildhauer. Das Modell kann konkret sein, wie das Schiffsmodell, oder abstrakt, wie ein Modell zur Rentenberechnung. Auch das Modellierte kann konkret sein, wie das Schiff, oder abstrakt, wie die zahlenmäßige Entwicklung der Bevölkerung. In Abb. 1.3 haben wir die Erklärung des Modellbegriffes aus einem allgemeinen Lexikon angegeben. Für unsere Zwecke ist darin die Variante zum „Sprachgebrauch verschiedener Wissenschaften" zutreffend. Die Modelle in der Informatik sind im Allgemeinen abstrakte Abbilder oder Vorbilder zu konkreten oder abstrakten Originalen. So ist das Modell der Flussüberquerung im vorigen Abschnitt ein abstrakter, endlicher Automat. Auch das Modellierte ist in diesem Beispiel abstrakt: der schrittweise Ablauf der Flussüberquerung mit den dabei durchlaufenen Zuständen.

Man beachte, dass in der Logik, einem Teilgebiet der Mathematik und Informatik, der Begriff Modell mit einer sehr speziellen, anderen Bedeutung verwendet wird: *Eine Struktur S ist ein Modell der logischen Formeln F, wenn alle Formeln aus F für S gelten* (siehe auch Kapitel 4).

Modell [italien., zu lat. modulus „Maß, Maßstab"], allg. Muster, Vorbild, Entwurf.
– Mensch (auch Tier), der (das) als Vorbild für künstler. Studien oder Kunstwerke dient („sitzt").
– in der Bildhauerei meist in verkleinerter Form ausgeführter Entwurf einer Plastik oder Tonarbeit, die in Bronze gegossen werden soll. - †Architekturmodell.
– in der Modebranche Bez. für 1. ein nur einmal oder in eng begrenzter Anzahl hergestelltes Kleidungsstück. (*M.kleid*); 2. die Vorlage für eine Vervielfältigung; 3. svw. Mannequin.
– im Sprachgebrauch verschiedener Wiss. (Philosophie, Naturwiss., Soziologie, Psychologie, Wirtschaftswiss., Politikwiss., Kybernetik u.a.) ein Objekt materieller oder ideeller (Gedanken-M.) Natur, das von einem Subjekt auf der Grundlage einer Struktur-, Funktions- oder Verhaltensanalogie für ein anderes Objekt (*Original*) eingesetzt und genutzt wird, um Aufgaben zu lösen, deren Durchführung unmittelbar am Original selbst nicht möglich bzw. zu aufwendig ist (z. B. Flugzeug-M. im Windkanal). Die **Modellmethode** vollzieht sich in vier Schritten: 1. Auswahl (Herstellung eines dem [geplanten] Original entsprechenden M.; 2. Bearbeitung des M., um neue Informationen über das M. zu gewinnen (**Modellversuch**; †Ähnlichkeitsgesetze); 3. Schluss auf Informationen über das Original (meist Analogieschluß); ggf. 4. Durchführung der Aufgabe am Original. Infolge der Relationen zw. Subjekt, Original und M. (**Modellsystem**) ist ein M. einsetzbar u. a. zur Gewinnung neuer Informationen über das Original (z. B. Atom-M.), zur Demonstration und Erklärung (z. B. Planetarium), zur Optimierung des Originals (z. B. Netzplan), zur Überprüfung einer Hypothese oder einer techn. Konstruktion (z. B. Laborversuch). - Abweichend von diesem M.begriff versteht die *mathemat. Logik* unter M. eine Interpretation eines Axiomensystems, bei der alle Axiome dieses Systems wahre Aussagen darstellen. Diese **Modelltheorie** liefert grundlegende Verfahren zur Behandlung von Fragen der Vollständigkeit, Widerspruchsfreiheit und Definierbarkeit.

Abbildung 1.3: Modellbegriff aus *Meyers Neues Lexikon* [20]

Modelle sind absichtlich nicht originalgetreu; sie heben bestimmte Eigenschaften hervor und lassen andere weg. Der intendierte Verwendungszweck des Modells bestimmt, welche Eigenschaften modelliert werden und welches Kalkül zu deren Beschreibung besonders geeignet ist. So werden beim Hausbau für verschiedene Zwecke ganz unterschiedliche Arten von Modellen verwendet:

- ein Gebäudemodell zur Vermittlung des optischen Eindruckes,
- ein Grundriss zur Einteilung der Räume und des Grundstückes,
- ein Kostenplan zur Finanzierung und
- ein Gewerkeplan zur Durchführung des Baus.

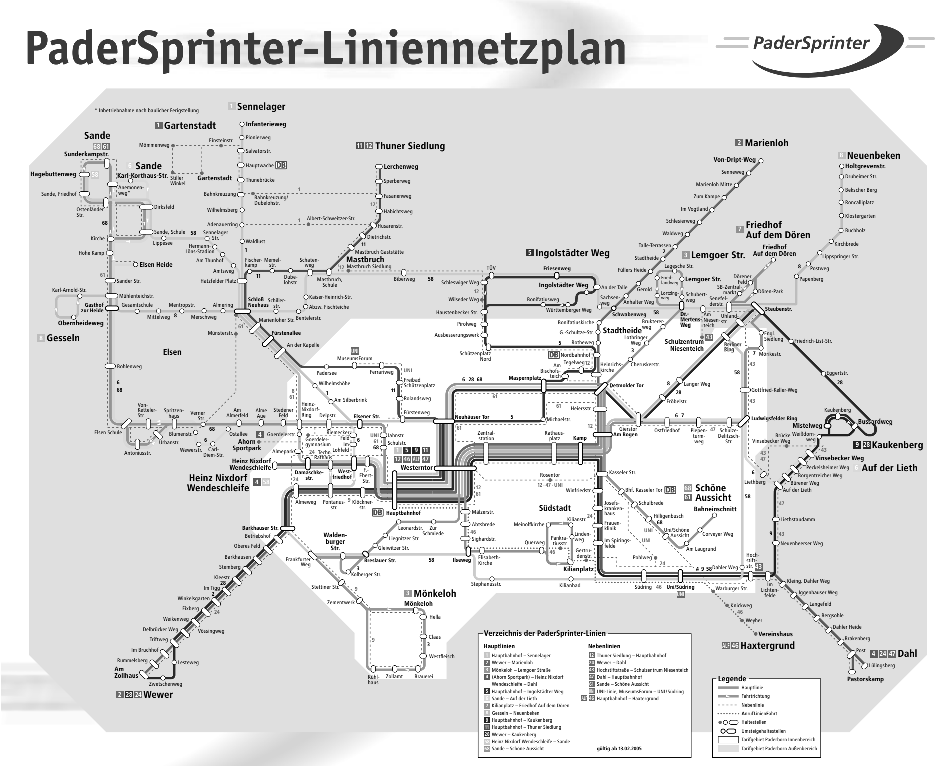
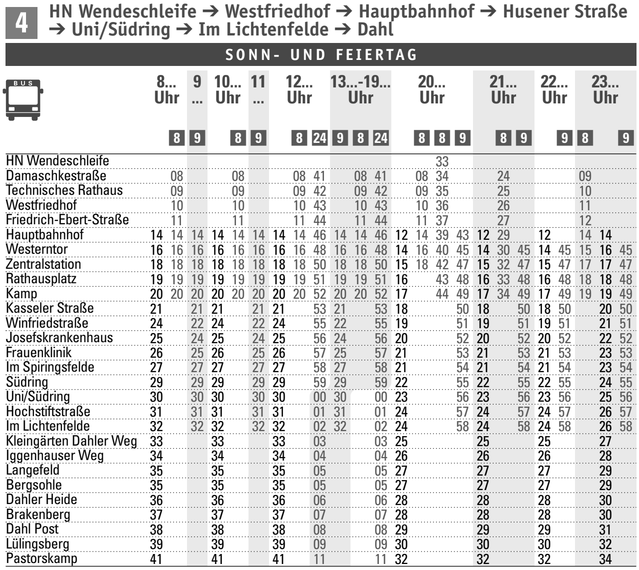


Abbildung 1.4: Streckenplan der Paderborner Buslinien [21]

Abb. 1.4 zeigt einen Streckenplan der Paderborner Buslinien. Er soll beschreiben, welche Stadtteile und Haltestellen von welchen Buslinien angefahren werden und welche Umsteigemöglichkeiten es gibt. Deshalb sind der Zusammenhang einzelner Linien, die Umsteigemöglichkeiten, die Haltestellen und die Stadtteile hervorgehoben. Genauere topografische Informationen und Straßenverläufe sind weggelassen. Sie sind für den intendierten Zweck nicht relevant. Der Fahrplan in Abb. 1.5 modelliert eine einzelne Buslinie für einen ganz anderen Zweck: Man soll zu jeder Haltestelle alle Abfahrtszeiten der Busse entnehmen und feststellen können, wann der Bus auf seiner Strecke andere Haltestellen erreicht.

Mit dem fertiggestellten Modell werden meist weitere Arbeiten durchgeführt, die der Zweckbestimmung entsprechen, z. B.

- Operationen, die man am Original nicht durchführen kann, etwa die Messung des Auftriebs neuer Formen von Flugzeugflügeln im Windkanal oder im Simulator;
- bestimmte Aspekte eines komplexen Gebildes untersuchen und verstehen, z. B. die Geschäftsabläufe in einer Firma. Schon das Herstellen eines Modells dafür vertieft das Verständnis deutlich;



4 HN Wendeschleife ➔ Westfriedhof ➔ Hauptbahnhof ➔ Husener Straße ➔ Uni/Südring ➔ Im Lichtenfelde ➔ Dahl

SONN- UND FEIERTAG

BUS	8... Uhr		9 ...	10... Uhr		11 ...	12... Uhr			13...-19... Uhr			20... Uhr				21... Uhr			22... Uhr		23... Uhr		
		8	9		8	9		8	24	9	8	24		8	8	9		8	9		9	8		9
HN Wendeschleife															33									
Damaschkestraße		08			08			08	41		08	41		08	34			24				09		
Technisches Rathaus		09			09			09	42		09	42		09	35			25				10		
Westfriedhof		10			10			10	43		10	43		10	36			26				11		
Friedrich-Ebert-Straße		11			11			11	44		11	44		11	37			27				12		
Hauptbahnhof	14	14	14	14	14	14	14	14	46	14	14	46	12	14	39	43	12	29		12		14	14	
Westerntor	16	16	16	16	16	16	16	16	48	16	16	48	14	16	40	45	14	30	45	14	45	15	16	45
Zentralstation	18	18	18	18	18	18	18	18	50	18	18	50	15	18	42	47	15	32	47	15	47	17	17	47
Rathausplatz	19	19	19	19	19	19	19	19	51	19	19	51	16		43	48	16	33	48	16	48	18	18	48
Kamp	20	20	20	20	20	20	20	20	52	20	20	52	17		44	49	17	34	49	17	49	19	19	49
Kasseler Straße	21		21	21		21	21		53	21		53	18			50	18		50	18	50		20	50
Winfriedstraße	24		22	24		22	24		55	22		55	19			51	19		51	19	51		21	51
Josefskrankenhaus	25		24	25		24	25		56	24		56	20			52	20		52	20	52		22	52
Frauenklinik	26		25	26		25	26		57	25		57	21			53	21		53	21	53		23	53
Im Spiringsfelde	27		27	27		27	27		58	27		58	21			54	21		54	21	54		23	54
Südring	29		29	29		29	29		59	29		59	22			55	22		55	22	55		24	55
Uni/Südring	30		30	30		30	30		00	30		00	23			56	23		56	23	56		25	56
Hochstiftstraße	31		31	31		31	31		01	31		01	24			57	24		57	24	57		26	57
Im Lichtenfelde	32		32	32		32	32		02	32		02	24			58	24		58	24	58		26	58
Kleingärten Dahler Weg	33			33			33		03			03	25				25			25			27	
Iggenhauser Weg	34			34			34		04			04	26				26			26			28	
Langefeld	35			35			35		05			05	27				27			27			29	
Bergsohle	35			35			35		05			05	27				27			27			29	
Dahler Heide	36			36			36		06			06	28				28			28			30	
Brakenberg	37			37			37		07			07	28				28			28			30	
Dahl Post	38			38			38		08			08	29				29			29			31	
Lülingsberg	39			39			39		09			09	30				30			30			32	
Pastorskamp	41			41			41		11			11	32				32			32			34	

Abbildung 1.5: Busfahrplan von Paderborner Buslinien [21]

- das Modell kann der Kommunikation dienen zwischen Auftraggeber und Hersteller des Originals, wie beim Hausbau, oder zwischen dem Anbieter und dem Nutzer eines Dienstes, wie die Buspläne in Abb. 1.4 und 1.5;
- bei solchen Auseinandersetzungen wird häufig das Modell so lange angepasst, bis Einigkeit über die beschriebenen Eigenschaften hergestellt ist. Dann fixiert das Modell dieses Ergebnis, z. B. als Anforderungen und Spezifikation in der Software-Erstellung.

Der letzte Aspekt ist von besonderer Bedeutung: An Modellen kann man prüfen und nachweisen, dass alle für den Zweck relevanten Eigenschaften korrekt und vollständig erfasst sind. Man bezeichnet den Vorgang auch als Validierung des Modells oder engl. *model checking*. So kann man prüfen, ob der Finanzplan alle Kosten erfasst, sie korrekt aufsummiert und die Kostengrenze eingehalten wird. Formale Modellierungskalküle erlauben es, solche Prüfungen auch systematisch oder automatisch auszuführen. So kann man z. B. an dem endlichen Automaten für die Flussüberquerung in Abb. 1.2 nachweisen, dass es keine weiteren zulässigen Zustände außer den angegebenen gibt.

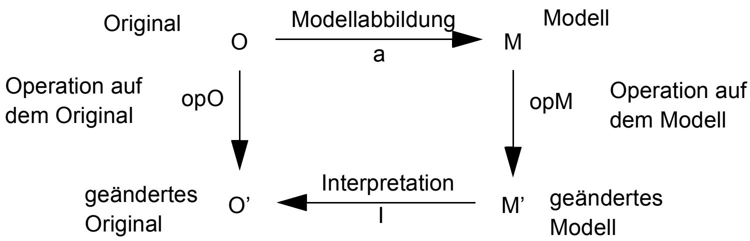


Abbildung 1.6: Bezug zwischen Original und Modell

Zwischen Modell und Original muss ein enger Bezug bestehen, damit das Modell die oben beschriebenen Zwecke erfüllen kann. Das Diagramm in Abb. 1.6 beschreibt diese Relationen: Die Modellabbildung führt vom Original zum Modell. Dabei wird meist vergröbert, abstrahiert, formalisiert und Irrelevantes weggelassen. Die Interpretation des Modells führt wieder zum Original. Wir verlangen, dass Änderungen am Original denen am Modell bezüglich dieser Abbildungen entsprechen. Für alle relevanten Operationen *opO* am Original muss das Diagramm kommutieren, d.h.

opO (O) = I (opM (a (O)))

Soll z. B. im Neubau eines Hauses die Position einer Tür verändert werden, so muss dies durch eine entsprechende Modifikation des Grundrisses eindeutig verändert werden können. Soll jedoch das mit KIND beschriftete Zimmer nicht als Kinderzimmer, sondern als Arbeitszimmer genutzt werden, ist diese Änderung irrelevant für den Grundriss und wird nicht vom Modell erfasst.

Ein Modell beschreibt immer nur bestimmte Aspekte des Originals und seiner Teile. Wir geben im Folgenden an, welche Kalküle sich besonders zur Beschreibung der Aspekte eignen. Die Beispiele entnehmen wir dem Modell der Flussüberquerung aus Abschnitt 1.1.

1. **Struktur:** Zusammensetzung des Originals aus Bestandteilen, z. B. ein Zustand ist ein Paar von Teilmengen von {M, W, Z, K}.
   Kalküle: **Wertebereiche, Entity-Relationship, Kontextfreie Grammatiken.**
2. **Eigenschaften** von Teilen des Originals, z. B. ein Zustand ist zulässig oder unzulässig.
   Kalküle: **Wertebereiche, Entity-Relationship, Logik.**
3. **Beziehungen** zwischen Teilen des Originals, z. B. der Wolf frisst die Ziege, die Ziege frisst den Kohlkopf.
   Kalküle: **Logik**, **Entity-Relationship**.
4. **Verhalten** des Originals bei Operationen, z. B. die Folgen von Zustandsübergängen.
   Kalküle: **endliche Automaten**, **Petri-Netze**, **Algebren**, **Graphen**.

Zum Schluss möchten wir dafür plädieren, dass Modelle möglichst *deklarativ* oder *deskriptiv* formuliert werden. Das bedeutet, dass das Modell Aussagen über Struktur, Eigenschaften, Beziehungen und Verhalten des Originals macht, die immer gelten.

Im Gegensatz dazu stehen *operationale* Beschreibungen, die angeben, wie sich das Original unter bestimmten Operationen verhält. Diese werden häufig durch Beispiele von Abläufen gegeben und sind eher unpräzise. Deshalb erfüllen sie die Anforderungen an die Modellierung weniger gut als präzise deklarative Aussagen. Wir demonstrieren die beiden Arten der Beschreibung an einer Balkenwaage, wie sie in Abb. 1.7 abstrahiert ist.

Deklarativ: Die Waage ist im Gleichgewicht, wenn sich die Gewichte *x* und *y umgekehrt proportional zu den Längen der Balken a und b verhalten:* $x * a = y * b$.

Operational: Wenn ich auf den *Balken der Länge a ein Gewicht x auflege*, muss ich auf den *Balken der Länge b ein Gewicht* $y = x * a/b$ auflegen, damit die Waage wieder im Gleichgewicht ist.

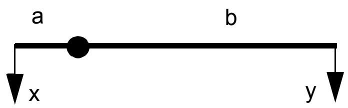


Abbildung 1.7: Modell einer Balkenwaage

Übungen

1.1 Getränkeautomat

In der Firma Kastens AG steht ein Getränkeautomat. Damit man gut zu ihm hin findet, ist er rot lackiert. Man kann Cola, Fanta, Sprite und Wasser kaufen. Jedes Getränk kostet 1 Euro. Der Automat hält genügend viele 1-Euro-Stücke zum Wechseln bereit. Ein Mitarbeiter möchte nun eine Cola kaufen. Er hat ein 2-Euro-Stück und ein 0,50-Euro-Stück in der Geldbörse.

a) Geben Sie die relevanten Objekte an, die zum Modellieren des Bezahlens benötigt werden.

b) Beschreiben Sie die Zustände des Systems. Welche Informationen liefern diese Zustände?

c) Modellieren Sie das Bezahlen analog zum Beispiel in Abschnitt 1.1.

1.2 Schokolade

Gegeben seien 3 Stapel aus Schokolade. Der erste besteht aus 4 Tafeln, der zweite aus 6 Tafeln und der dritte aus 14 Tafeln. Die Stapel sollen nun ausgeglichen werden, so dass auf jedem Stapel genau 8 Tafeln liegen. Allerdings darf in jedem Schritt nur zwischen genau zwei Stapeln umgeschichtet werden. Zudem müssen auf einen Stapel immer so viele Tafeln gelegt werden, wie bereits darauf liegen. Modellieren Sie das Problem analog zum Beispiel in Abschnitt 1.1.

Hinweis: Zweimal Umschichten genügt!

1.3 Modelle aus dem Alltag

Geben Sie 3 Modelle aus dem Alltag an, die nicht im Kapitel 1 vorkommen. Schreiben Sie zu jedem Modell auf:

a) Was wird modelliert?

b) Welchem Zweck dient das Modell?

c) Was sind die wichtigen Objekte, Eigenschaften, Zusammenhänge und Operationen?

2

Modellierung mit Wertebereichen

In Modellen von konkreten oder abstrakten Systemen oder Aufgaben kommen Objekte unterschiedlicher Art und Zusammensetzung vor. Damit die Beschreibung nicht nur ein Beispiel, sondern hinreichend allgemein ist, gibt man für einige Teile des Modells nur an, von welcher Art sie sind, lässt aber offen, welches individuelle Objekt dafür eingesetzt wird. Dafür fassen wir alle infrage kommenden Objekte in einem Wertebereich zusammen, der ihre gemeinsamen, für das Modell relevanten Eigenschaften charakterisiert. Welcher Wert aus dem Wertebereich gewählt wird, kann dann im Modell offen bleiben.

Betrachten wir als kleines Beispiel folgende informell formulierte Aufgabe:

Aufgabe 2.1

Ziehe drei Karten aus einem Kartenspiel. Bestimme die höchste der drei Karten.

Um diese Aufgabe zu präzisieren, müssen wir den Wertebereich für Karten eines Kartenspiels charakterisieren: Jede Karte wird durch zwei Angaben beschrieben: die Kartenart: Kreuz, Pik, Herz oder Karo, und das Kartensymbol: 7, 8, 9, 10, Bube, Dame, König, Ass. Jedes Einzelne ist ein Wertebereich. Alle Kombinationen aus einer Kartenart und einem Kartensymbol bilden den Wertebereich eines Kartenspiels, aus dem einzelne Karten als Elemente stammen. Den zweiten Teil der Aufgabe können wir präzisieren, indem wir eine Relation angeben, die für je zwei Karten entscheidet, welche höher ist. Solch eine Relation besteht aus einer Menge von Kartenpaaren. Je nach Spielregel kann es unterschiedliche solcher Vergleichsrelationen geben. Jede von ihnen ist aber Element aus demselben Wertebereich und erfüllt Anforderungen, die an Ordnungsrelationen gestellt werden.

In diesem Kapitel führen wir abstrakte Konzepte ein, mit denen man Wertebereiche für einfache und zusammengesetzte Werte präzise angeben kann. Wertebereiche sind Mengen, die beim Formulieren eines Modells eine bestimmte Rolle spielen. In Abschnitt 2.1 stellen wir deshalb die wichtigsten Begriffe der Mengenlehre vor.

Definition 2.1: Wertebereich

Ein ***Wertebereich ist eine Menge von Werten****, die im Sinne eines Modells als gleichartig angesehen werden: Wo ein Wert eines Wertebereiches W gefordert wird, kann prinzipiell jedes Element aus W diese Rolle übernehmen.* ■

Die Gleichartigkeit im Sinne des Modells illustriert der Wertebereich der Kartensymbole im obigen Beispiel: Er enthält so unterschiedliche Werte wie „10“ und „Bube“, da sie bei der Identifikation von Spielkarten die gleiche Rolle haben.

Aus Wertebereichen für einfache Werte, wie den Kartensymbolen oder Kartenarten, kann man Wertebereiche für zusammengesetzte Werte bilden, z.B. Paare zur Identifikation von Spielkarten. Dafür stellen wir in den Abschnitten 2.2 bis 2.7 die grundlegenden Begriffe Potenzmengen, kartesische Produkte, Vereinigungen, Folgen, Relationen und Funktionen vor. Dabei geben wir jeweils typische Anwendungsbeispiele an und weisen auf spezielle Modellierungstechniken hin.

Durch die Abschnitte hindurch werden wir jeweils Beiträge zu einer zusammenhängenden Modellierungsaufgabe entwickeln. Sie wird hier informell beschrieben und später schrittweise formalisiert:

Beispiel 2.1: Arbeitskreise der EU

Die EU-Kommission hat beschlossen, die Entscheidungsprozesse der EU mit formalen Methoden zu modellieren. Damit sollen drei Arbeitskreise befasst werden. An der Aktion beteiligen sich zunächst die Nationen Deutschland, Frankreich, Österreich und Spanien. Jede entsendet drei Delegierte. Die Arbeitskreise sollen so gebildet werden, dass in jedem Arbeitskreis jede Nation vertreten ist und dass unter Berücksichtigung der Fremdsprachenkenntnisse der Delegierten es in jedem Arbeitskreis eine gemeinsame Sprache gibt, die alle beherrschen. Es soll nur die Situation modelliert werden; ein Lösungsverfahren wird nicht gesucht.

In jeder präzisen, formalen Beschreibung wird angegeben, aus welchen Wertebereichen Objekte und die Werte von Variablen stammen. Das gilt für mathematische oder theoretische Definitionen ebenso wie für Spezifikationen von Aufgaben, sowie von Software- und Hardware-Systemen. Dabei werden die in diesem Kapitel eingeführten Begriffe verwendet, meist ohne sie einem speziellen Kalkül zuzuordnen. Auch andere formale Kalküle definiert man unter Verwendung dieser Begriffe.

Die Abstraktionen, die zur Konstruktion von Wertebereichen benutzt werden, sind auch Grundlage für Typsysteme in Programmiersprachen. So sind z.B. struct-Typen in C und record-Typen in Ada konkrete Ausprägungen der Abstraktion kartesisches Produkt, und abstrakte Folgen entsprechen Listen-Strukturen, wie sie in manchen Sprachen gebildet werden können oder als Typen vordefiniert sind. Deshalb ist auch das Modellieren mit abstrakten Wertebereichen, so wie es hier eingeführt wird, ein wichtiger Entwurfsschritt vor der Implementierung durch Datentypen und -strukturen im Programm. Auf der abstrakten Entwurfsebene kann man sich bemühen, das Modell möglichst gut auf die Aufgabe hin zu entwickeln, ohne dabei technische Restriktionen der Programmierung beachten zu müssen.

2.1 Mengen

Der Mengenbegriff ist grundlegend für das Modellieren mit Wertebereichen. Die dafür nötigen Aspekte der elementaren Mengenlehre fassen wir in diesem Abschnitt zusammen.

Definition 2.2: Menge

*Eine **Menge M** ist eine Zusammenfassung von verschiedenen Objekten, den **Elementen der Menge**. a ist ein Element der Menge M. Es wird notiert* $a \in M$.■

Diese informelle Definition reicht hier aus. Die axiomatische Mengenlehre definiert den Mengenbegriff strenger.

Mengen können auf zwei prinzipiell unterschiedliche Weisen angegeben werden:

- *extensional*, d.h. durch Aufzählen ihrer Elemente, z.B. $\{1, 4, 9, 16, 25\}$ oder
- *intensional*, d.h. durch Angabe einer Bedingung; alle Werte, die sie erfüllen, und nur diese, sind Elemente der Menge, z.B. $\{ a \mid a \in \mathbb{N}, a \text{ ist Quadratzahl und } a < 30\}$.

Die Bedingung entscheidet für jeden Wert unabhängig, ob er Element der Menge ist; Zusammenhänge zwischen mehreren Elementen kann sie nicht ausdrücken. Um unerwünschte Anomalien solcher Definitionen zu vermeiden (siehe unten), geben wir meist als Teil der Bedingung eine größere, schon definierte Menge an, aus der die infrage kommenden Werte stammen, wie $a \in \mathbb{N}$ in obigem Beispiel.

Eine Menge kann beliebig viele Elemente enthalten. Die *leere Menge* schreibt man auch $\emptyset$. Nicht-endliche Mengen kann man intensional definieren. Die beiden oben angegebenen Beispiele beschreiben dieselbe Menge auf unterschiedliche Weise. Genau genommen ist eine Menge ein Abstraktum, das wir nicht direkt angeben, sondern nur in einer ausgewählten Notation beschreiben.

Gemäß Definition 2.1 sind alle Elemente einer Menge verschieden. Die Mengenangabe $\{1, 2, 2, 3\}$ ist zwar korrekt, aber redundant. Sie gibt die gleiche Menge an wie $\{1, 2, 3\}$. Auch durch Vereinigung von Mengen (siehe unten) mit gleichen Elementen kommt kein Wert mehrfach in einer Menge vor. Wenn wir mehrfaches Vorkommen gleicher Werte modellieren wollen, müssen wir andere Konzepte anwenden (Funktion in Abschnitt 2.7 oder Folgen in Abschnitt 2.5).

In einer Menge sind ihre Elemente nicht geordnet. Die Reihenfolge, in der sie extensional aufgezählt werden, ist nicht relevant: $\{1, 2, 3\}$ und $\{1, 3, 2\}$ geben dieselbe Menge an.

Mengen können aus atomaren oder zusammengesetzten Elementen gebildet werden. Zusammengesetzte Elemente können z.B. Paare sein, die Spielkarten modellieren, wie in $\{(\text{Pik}, 10), (\text{Herz}, \text{Dame})\}$. Auch Mengen können Elemente einer Menge sein, z.B. die Menge $\{\{1\}, \{1, 4\}, \emptyset\}$ mit drei mengenwertigen Elementen. Für solche Mengen führen wir in den nächsten Abschnitten spezielle Begriffe ein.

Eine Menge kann auch verschiedenartige oder unterschiedlich strukturierte Elemente enthalten, z.B. in $\{1, (\text{Pik}, 10), \{1, 3\}, 9\}$ zwei ganze Zahlen, ein Paar und eine Menge. Aber bei der Modellierung mit Wertebereichen erlauben wir verschiedenartige Elemente nur in speziellen Fällen (siehe Abschnitt 2.4 Vereinigung).

Wenn wir Mengen als Wertebereiche definieren, geben wir ihnen meist einen Namen. Solche Definitionen notieren wir wie im Beispiel

Nationen := { Deutschland, Frankreich, Österreich, Spanien }

In der Modellierung sollte man möglichst **aussagekräftige Namen** wählen. Für unser Beispiel 2.1 der EU-Arbeitskreise benötigen wir noch folgende Mengen mit einfachen Elementen:

Sprachen := { Deutsch, Französisch, Spanisch }
DelegiertenIndex := { 1, 2, 3 }
ArbeitskreisIndex := { 1, 2, 3 }

Die beiden letzten Mengen haben wir eingeführt, um die Delegierten jeder Nation und die drei Arbeitskreise zu identifizieren. In der Modellierung nennt man solche Mengen, die zur Unterscheidung gleichartiger Objekte eingeführt werden, *Indexmengen*. Zu diesem Zweck werden wir sie bei der Weiterentwicklung des Beispiels in den nächsten Abschnitten einsetzen. Meist werden kleine ganze Zahlen als ihre Elemente gewählt, aber auch beliebige andere Werte erfüllen den Zweck der Indizierung. So hätte man den drei Arbeitskreisen auch symbolische Namen geben können, wie z.B. ParlamentsEntscheidungen, die einen Hinweis auf ihre Aufgabe geben.

Auch wenn man die beiden Indexmengen wie oben gleich definiert, sollte man nicht der Versuchung erliegen, eine davon einzusparen, denn sie spielen ganz verschiedene Rollen und können auch unterschiedliche Werte haben, z.B. wenn eine Nation mehr Delegierte stellt, als es Arbeitskreise gibt.

Bei der Definition und Anwendung von Mengen als Wertebereichen werden auch Verknüpfungen mit Operationen über Mengen formuliert. Wir definieren sie hier informell:

Definition 2.3: Mengenoperationen

Bezeichnung	*Notation*	*Bedeutung*
ist Teilmenge	$M \subseteq N$	*aus* $a \in M$ *folgt* $a \in N$
ist echte Teilmenge	$M \subset N$	$M \subseteq N$ *und* $M \neq N$
Vereinigung	$M \cup N$	$\{x \mid x \in M \text{ oder } x \in N\}$
Durchschnitt	$M \cap N$	$\{x \mid x \in M \text{ und } x \in N\}$
Differenz	$M \setminus N$	$\{x \mid x \in M \text{ und } x \notin N\}$

Außerdem bedeutet die Formulierung zwei Mengen M und N sind ***disjunkt****, dass gilt* $M \cap N = \emptyset$.

Die Anzahl der Elemente einer Menge M heißt ihre ***Kardinalität*** *und wird notiert als* $|M|$ *oder* $Card(M)$. ■

Zum Schluss weisen wir noch darauf hin, dass man mit intensionalen Beschreibungen auch Mengen definieren kann, für die man die Frage „gilt a $\in$ M?" prinzipiell nicht entscheiden kann. Dies wird mit Russels Paradoxon demonstriert:

> P sei die Menge aller Mengen, die sich nicht selbst als Element enthalten, also $P := \{ x \mid x \notin x \}$. Dann führt die Frage „Ist P ein Element von P?" zum Widerspruch.

Die Ursache des Problems liegt darin, dass P selbst zum Wertebereich gehört, aus dem die Elemente von P stammen. Um solche Anomalien zu vermeiden, geben wir in intensionalen Mengendefinitionen an, aus welchem größeren, schon definierten Wertebereich die Elemente stammen, z.B.:

$$M := \{ a \mid a \in \mathbb{N}, a \text{ ist eine Quadratzahl und } a < 30 \}$$

Damit tatsächlich entschieden werden kann, welche Elemente M enthält, muss man noch verlangen, dass die Bedingung über a entscheidbar ist. Diese Einschränkungen schließen natürlich nicht aus, Mengen rekursiv zu definieren, was häufig sinnvoll ist, z.B.

$$\begin{aligned} \text{Sonnensystem} := \; & \{ \text{Sonne} \} \cup \\ & \{ x \mid x \in \text{Himmelskörper}, x \text{ umkreist } y \text{ und } y \in \text{Sonnensystem} \} \end{aligned}$$

2.2 Potenzmengen

Häufig treten in der Modellierung Gegenstände auf, die durch Mengen beschrieben werden, z.B. die Menge der Sprachen, die ein Delegierter spricht. Der Wertebereich, aus dem sie stammen, ist dann eine Menge, welche Mengen als Elemente enthält. Eine spezielle Form davon sind die Potenzmengen:

Definition 2.4: Potenzmenge

Die ***Potenzmenge*** *(engl.* ***powerset****) einer Grundmenge U ist die Menge aller Teilmengen von U, geschrieben* ***Pow*** *(****U****) oder* $\wp$ *(****U****). Als Formel:*

$$Pow(U) := \{ M \mid M \subseteq U \}$$ ■

Zum Beispiel ist zur Grundmenge $U := \{ d, f, s \}$ die Potenzmenge
$Pow(U) := \{ \emptyset, \{ d \}, \{ f \}, \{ s \}, \{ d, f \}, \{ d, s \}, \{ f, s \}, \{ d, f, s \} \}$

Jedes Element der Potenzmenge ist eine Menge, deren Elemente eine Kombination von Elementen der Grundmenge sind. Die Anzahl aller Kombinationen von Elementen aus U bestimmt auch die *Kardinalität von Potenzmengen*: Es gilt $|Pow(U)| = 2^n$, falls die Grundmenge U endlich ist, $|U| = n$. Falls U leer ist, enthält $Pow(U)$ nur die leere Menge. Nehmen wir z.B. als Grundmenge

$$\text{Sprachen} := \{ \text{Deutsch, Französisch, Spanisch} \}$$

dann wird die Potenzmenge SprachMengen := Pow (Sprachen) wie im Beispiel oben gebildet. Wenn wir Werte modellieren, die Teilmengen einer Menge U sind, dann stammen sie aus dem Wertebereich Potenzmenge von U, z.B.