

# Ein Ansatz zur Konstruktion begrifflicher Schemata

Gerhard Kratz

Fachhochschule Frankfurt am Main  
Fachbereich 2 – Studiengang Informatik  
Nibelungenplatz 1  
D-60318 Frankfurt am Main  
g\_kratz@fb2.fh-frankfurt.de

**Abstract:** Der hier vorgestellte Ansatz zur Konstruktion begrifflicher Schemata (SchemaKonstruktionsAnsatz - SKA) unterscheidet sich von bisher veröffentlichten Ansätzen derselben Zielsetzung dadurch, dass die Schemaklasse mit der Menge ihrer Exemplare (ihrer Extension) definiert wird. Auch Werte können Exemplare einer Schemaklasse sein. Dadurch kann die Mengenlehre in großem Umfang für die Definition der weiteren Darstellungsmittel des SKA nutzbar gemacht werden. Es wird insbesondere versucht zu zeigen, dass man so auf einheitlicher Basis die Zeitabhängigkeit der Existenz von Objekten und der Gültigkeit von Attributwerten darstellen kann.

## 1 Einleitung

Im Entity-Relationship-Ansatz [Ch76] und vielen seiner Erweiterungen (z.B. EER nach [EN00] oder HERM nach [Th00]) gehört jeder Wert (Zahl, Zeitangabe, Zeichenfolge, Wahrheitswert) dem Wertebereich eines Wertetypen an. Ganz ähnlich in vielen Ansätzen der objekt-orientierten Analyse (s.z.B. die Unified Modeling Language – UML – nach [RJB99]): ein Wert gehört dort zum Wertebereich eines Datentypen. Weder für einen Wertetypen noch für einen Datentypen können Attribute vereinbart werden; es ist auch nicht vorgesehen, dass ein Wertetyp an einem Beziehungstypen bzw. ein Datentyp an einer Assoziation beteiligt ist. Einige Konsequenzen sollen an dem folgenden Beispiel beleuchtet werden.

Beispiel: Zur Geschäftspolitik eines Autovermieters mögen die Regeln gehören, dass jeder Privatkunde zu jeder Zeit höchstens ein Fahrzeug entliehen haben kann und dass jedes Fahrzeug zu jeder Zeit von höchstens einem Privatkunden entliehen ist.

In allen mir bekannten Ansätzen zur Darstellung der Zeit (Übersichten bieten [GJ99] und [RP92]) in begrifflichen Schemata – dies sind ausschließlich solche Ansätze, die auf einem erweiterten ER-Ansatz beruhen - werden für die Existenzzeit von Entitäten und Beziehungen sowie für die Zeitabhängigkeit von Attributwerten eigene Darstellungsmittel eingeführt. In den meisten dieser Ansätze kann die Zeit nicht zusammen mit den o.a. Kardinalitätenbeschränkungen dargestellt werden.

Gleichzeitig ist darzustellen, dass ein Kunde ein Fahrzeug für ein Zeitintervall entleiht. Es ist also anzugeben, wie einzelne Zeitpunkte so zu Intervallen zusammengefasst werden können, dass der Realweltausschnitt in geeigneter Weise abgebildet wird.

Nun ist ohne Weiteres denkbar, dass auch Wertetypen an Beziehungstypen bzw. Datentypen an Assoziationen beteiligt sein könnten. Damit könnten die Beispielsachverhalte, eine geeignete Definition der Kardinalitätenbeschränkungen vorausgesetzt, in einheitlicher Weise und direkter dargestellt werden.

Mit dem hier vorgestellten Ansatz wird versucht, weitergehend als bisher die Konstruktion begrifflicher Schemata auf einer einheitlichen Basis zu ermöglichen. d.h. es wird versucht, die Einführung von Darstellungsmitteln *ad hoc* weiter hinauszuschieben, als das bisher geschehen ist. Die Grundbegriffe des Ansatzes werden auf der Grundlage der Mengenlehre eingeführt.

## 2 Schemaklassen

Eine Schemaklasse  $\underline{K}$  ist eine Zusammenfassung von Objekten, die die Exemplare von  $\underline{K}$  sind. Es kann möglich sein, einer Schemaklasse ein Objekt hinzuzufügen bzw. aus ihr zu entfernen. Man denke beispielsweise an ein Fahrzeug, das in den Fuhrpark eines Autovermieters aufgenommen bzw. aus ihm entfernt wird. Hinzufügen bzw. Entfernen eines Exemplars erfolgen in vernachlässigbarer Zeit. Die Menge  $K$  der Exemplare von  $\underline{K}$  ist ihr Bestand.

Definition: Eine Schemaklasse  $\underline{K}$  ist ein abstrakter Datentyp

- mit den Wertebereichen:
  - $w$ : Menge von Objekten,  
wird interpretiert als Menge aller Objekte, die jemals Exemplare von  $\underline{K}$  waren, sind oder sein werden
  - $O = \wp(w)$  (Potenzmenge von  $w$ )  
so dass der Bestand  $K \in \Omega$  ist und
- mit einer Menge von statischen Integritätsbedingungen und Zustandsübergangsbedingungen, die für den Bestand  $K$  gelten. (Hier wird ausschließlich auf implizit formulierte, statische Integritätsbedingungen eingegangen.)

Eine Variable, die für eine Schemaklasse steht, wird mit einem großen, unterstrichenen Buchstaben bezeichnet. Eine Variable, die für einen Bestand steht, wird mit einem Großbuchstaben bezeichnet, der nicht unterstrichen ist. Ist  $\underline{K}$  eine Schemaklasse, dann ist  $K$  ihr Bestand und umgekehrt.

Der Begriff des Attributs wird in der Definition der Schemaklasse nicht verwendet; Attribute werden als Rollen definiert (s. 3).

Ein Klassenschema ist eine Schemaklasse mit ihrem Namen und ohne ihren Bestand.

Eine Schemaklasse ist entweder eine empirische oder eine abgeleitete Schemaklasse.

Zu einer empirischen Schemaklasse gibt es eine Menge von Transaktionen, mit denen ihr Objekte hinzugefügt bzw. entnommen werden können. Transaktionen sind konsistent, d.h. das Ergebnis ihrer Ausführung steht nicht im Widerspruch zu den Integritätsbedingungen.

Zu einer abgeleiteten Klasse gibt es eine Ableitungsvorschrift, mit der ihr Bestand bestimmt wird. Ableitungsvorschriften und explizit formulierte Integritätsbedingungen sind nicht Gegenstand dieses Beitrags.

Eine Schemaklasse ist entweder eine Atomklasse oder eine Relationsklasse, die nachfolgend definiert werden.

## 2.1 Atomklassen

Definition: Eine Atomklasse ist eine Schemaklasse, deren Exemplare sämtlich Atome, d.h. nicht-zusammengesetzte Objekte, sind. Eine Atomklasse ist entweder eine Entitätenklasse oder eine Werteklasse.

Eine Entität ist ein Atom, das nicht durch sich selbst identifiziert wird, sondern zu dessen Identifizierung man einen Identifikator benötigt. (vgl. [Em97])

Definition: Eine Entitätenklasse  $\underline{E}$  ist eine Atomklasse, deren Exemplare sämtlich Entitäten sind. Dem Klassenschema von  $\underline{E}$  gehört eine Funktion

$$I : w \leftrightarrow NAT$$

an.  $I$  heißt der Identifikator von  $\underline{E}$  und dient dazu, die Exemplare von  $\underline{E}$  eindeutig zu kennzeichnen.

Eine Entitätenklasse  $\underline{E}$  wird im Schemadiagramm als Rechteck dargestellt, das mit dem Namen von  $\underline{E}$  beschriftet ist.

Ein Wert ist ein Atom, das durch sich selbst identifiziert wird. (vgl. [Em97]) Ein Wert ist entweder eine Zeichenfolge oder ein Wahrheitswert oder eine Größenangabe. Eine Größenangabe ist entweder eine ganze Zahl oder einer von äquidistant auf einer Zahlen- oder Zeitachse (Kalender) angeordneten Punkten (*tick*). Der Abstand zwischen zwei unmittelbar aufeinander folgenden *ticks* kann z.B. ein Hundertstel einer Währungseinheit oder ein Tag oder eine Sekunde sein. *ticks* werden zur Angabe der Ränder von Intervallen verwandt (s. 5). Zusätzlich gibt es die (Pseudo-)Größenangaben  $-\infty$  und  $\infty$ . Der Zeitpunkt, an dem das aktuelle Datum bzw. die aktuelle Uhrzeit abgelesen wird, wird mit NOW bezeichnet. (vgl. z.B. [EEK91])

Definition: Eine Werteklasse ist eine Atomklasse, deren Exemplare Werte sind. Eine Werteklasse ist entweder eine Klasse von Zeichenfolgen, eine Klasse von Wahrheitswerten oder eine Klasse von Größenangaben. Eine Klasse von Größenangaben ist entweder eine Klasse von Zahlen oder eine Klasse von Zeitangaben.

Eine Werteklasse  $\underline{W}$  wird im Schemadiagramm als abgerundetes Rechteck dargestellt, das mit dem Namen von  $\underline{W}$  beschriftet ist. Der Klassenname "ET" steht für "existence time"; sie enthält als Exemplare alle *ticks* von einer anzugebenden Anfangszeit bis zum aktuellen Zeit-*tick*.

Entsprechend den Typen der Programmiersprachen, insbesondere der Datenmanipulationssprachen, gibt es die folgenden Basiswertklassen, denen Exemplare weder hinzugefügt noch entnommen werden können:

|               | darzustellende Menge             | Basiswerteklasse |                        | darzustellende Menge | Basiswerteklasse |
|---------------|----------------------------------|------------------|------------------------|----------------------|------------------|
| Zahlenklassen | reelle Zahlen                    | REAL             | Klasse von Zeitangaben | Kalenderdatum        | DATE             |
|               | nicht-negative, reelle Zahlen    | REAL_PLUS_0      |                        | Uhrzeit              | TIME             |
|               | positive, reelle Zahlen          | REAL_PLUS        |                        | Datum-Uhrzeit        | TIMESTAMP        |
|               | ganze Zahlen                     | INTEGER          |                        | Zeichenfolge         | STRING           |
|               | nicht-negative, ganze Zahlen     | NAT_0            |                        | Wahrheitswert        | BOOLEAN          |
|               | natürliche Zahlen ( $n \geq 1$ ) | NAT              |                        |                      |                  |

Abb. 1: Basiswerteklassen im SchemaKonstruktionsAnsatz

Jede weitere Werteklasse  $\underline{W}$  ist Unterklasse einer Basiswerteklasse  $\underline{B}$ , d.h. ihr Bestand  $W$  ist Teilmenge des Bestandes  $B$  von  $\underline{B}$ . (s. 4) Eine Werteklasse, die nicht Basiswerteklasse ist, kann veränderlich sein, d.h. es können ihr Exemplare hinzugefügt bzw. entnommen werden.

Ein Beispiel für eine veränderliche Werteklasse ist eine Zusammenfassung von Ländernamen. Z.B. durch Neugründung oder Vereinigung von Staaten kann es erforderlich werden, dieser Werteklasse Exemplare hinzuzufügen bzw. zu entnehmen. (s. Abb.2)

## 2.2 Relationsklassen

Definition: Gegeben seien die Schemaklassen  $\underline{K}_1, \dots, \underline{K}_n$  ( $n \geq 2$ ) mit den Beständen  $K_1, \dots, K_n$ . Die Schemaklasse  $\underline{R}$  ist eine  $n$ -stellige Relationsklasse, wenn ihr Bestand die Relation  $R \subseteq K_1 \times \dots \times K_n$  ist.

Ein Exemplar einer  $n$ -stelligen Relationsklasse ist also ein  $n$ -Tupel. Die  $\underline{K}_i$  heißen die an  $\underline{R}$  beteiligten Schemaklassen. Dabei muss es sich nicht ausschließlich um Entitätenklassen handeln, sondern an einer Relationsklassen können auch Werteklassen und andere Relationsklassen beteiligt sein. Somit ist es auch nicht notwendig, ein eigenes Mittel zu Darstellung von Relationsklassen vorzusehen, an denen Werteklassen beteiligt sind (schwache Entitätstypen im ER-Ansatz, *qualifiers* in der UML).

Das Symbol für eine Relationsklasse im Diagramm besteht aus einem Rechteck, das mit dem Namen der Relationsklasse beschriftet ist, einer Raute und einem Pfeil, der von der Raute zum Rechteck gerichtet ist. (s.z.B. die Relationsklasse "Ausleihe an Privatkunden" in Abb. 2) Wenn eine Relationsklasse nicht selbst an einer (anderen) Relationsklasse beteiligt ist, können Rechteck und Pfeil im Symbol entfallen. Ist die Relationsklasse darüberhinaus zweistellig, so können die Symbole der beiden beteiligten Klassen einfach durch eine Kante miteinander verbunden werden.

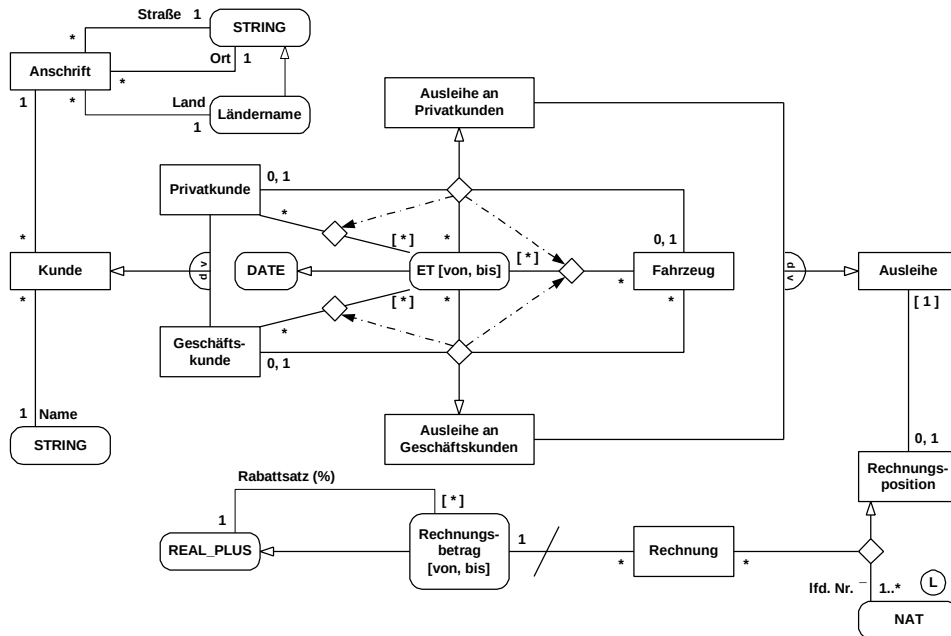


Abb. 2: einfaches, begriffliches Schema für einen Autovermieter

Zeitreihen [LE99] können ohne zusätzliche Mittel dargestellt werden. Wenn einem Exemplar einer Klasse von Zeitangaben jeweils  $n$  Objekte zugeordnet werden, so können dazu  $n$  einwertige Attribute verwandt werden.

### 3 Möglichkeiten, eine Relationsklasse auszuwerten

#### 3.1 Auswertungen

Wenn  $(x_1, \dots, x_i, \dots, x_n) \in R$  und  $x_i \in K_i$  ist, dann ist die  $i$ -te, einzelwertige Projektion des Bestands  $R$  der Relationsklasse  $\underline{R}$ :

$$p_i(R; x_1, \dots, x_i, \dots, x_n) = x_i$$

$\underline{R}$  ist Urbildklasse,  $K_i$  ist Bildklasse von  $p_i$ . Eine einzelwertige Projektion der Relationsklasse  $\underline{R}$  wird im Diagramm durch eine Kante zwischen der Raute des Symbols von  $\underline{R}$  und ihrer Bildklasse dargestellt. Wird der Name einer einzelwertigen Projektion angegeben, so wird ihm ein kleiner Pfeil in Richtung seiner Bildklasse angefügt. (vgl. Projektion "lfd. Nr." in Abb. 2).

Eine  $m$ -stellige, mengenwertige Projektion des Bestands  $R$  der Relationsklasse  $\underline{R}$  auf die Bestände der an ihr beteiligten Schemaklassen  $\underline{K}_{i_1}, \dots, \underline{K}_{i_m}$  mit  $\{i_1, \dots, i_m\} \subseteq \{1, \dots, n\}$  und  $x_j \subseteq K_j$  ist

$$p_{i_1, \dots, i_m}(R) = \{(x_{i_1}', \dots, x_{i_m}') \mid \exists (x_1, \dots, x_n) \in R : x_{i_k}' = x_{i_k}\}$$

Eine Selektion mit Gleichheit auf dem Bestand einer Relationsklasse  $\underline{R}$  an Hand von Exemplaren der an ihr beteiligten Schemaklassen  $\underline{K}_{i_1}, \dots, \underline{K}_{i_m}$  mit  $\{i_1, \dots, i_m\} \subseteq \{1, \dots, n\}$  und  $x_j \subseteq K_j$  ist

$$s_{i_1, \dots, i_m}(R; x_{i_1}', \dots, x_{i_m}') = \{(x_1, \dots, x_n) \in R \mid x_{i_k}' = x_{i_k}\}$$

Gegeben seien die disjunkten, nicht-leeren Mengen  $U = \{p_{u,1}, \dots, p_{u,l}\}$  und

$B = \{p_{b,1}, \dots, p_{b,m}\}$  von einzelwertigen Projektionen der Relationsklasse  $\underline{R}$ . Ferner seien  $x_{u,j} \in K_{u,j}$  und  $x_{b,k} \in K_{b,k}$ . Dann ist die Auswertung mit Gleichheit  $v$  der Relationsklasse  $\underline{R}$  mit den Urbildprojektionen  $U$  und den Bildprojektionen  $B$  folgendermaßen gegeben:

$$\begin{aligned} v_{U,B}(R; x_{u,1}, \dots, x_{u,l}) = \\ \{(x_{b,1}, \dots, x_{b,m}) \mid \\ \exists r \in R : p_{u,1}(r) = x_{u,1} \wedge \dots \wedge p_{u,l}(r) = x_{u,l} \wedge p_{b,1}(r) = x_{b,1} \wedge \dots \wedge p_{b,m}(r) = x_{b,m}\} \end{aligned}$$

Die Auswertung mit Gleichheit  $v$  lässt sich folgendermassen als Verknüpfung einer Selektion mit Gleichheit und einer mengenwertigen Projektion erklären:

$$v_{U,B}(R; x_{u,1}, \dots, x_{u,l}) = p_{b,1; \dots; b, m}(s_{u,1; \dots; u, l}(R; x_{u,1}, \dots, x_{u,l}))$$

### 3.2 Kardinalitätenbeschränkungen

Die Menge  $C$  der zugelassenen Mächtigkeiten einer Auswertung mit Gleichheit  $v$  heißt die Kardinalitätenbeschränkung von  $v$ :

$$C(v_{U,B}(R; x_{u,1}, \dots, x_{u,l})) \subseteq NAT$$

(vgl. [Th00], 170) Dabei ist zu beachten, dass dieser Begriff der Kardinalitätenbeschränkung nicht an eine 2-stellige Relationsklasse (vgl. UML [RJB99]) gebunden ist, noch ist er beschränkt auf genau eine Urbildprojektion (vgl. structural constraints bei [EN00]) oder auf genau eine Bildprojektion.

Die Kardinalitätenbeschränkungen werden in derselben Weise notiert wie in der UML. Allerdings gibt es im SKA auch Kardinalitätenbeschränkungen für 3- oder mehrstellige Relationsklassen. Z.B. kann lt. Abb. 2 ein Privatkunde zu jedem Zeitpunkt höchstens ein Fahrzeug entliehen haben, während ein Geschäftskunde zu einem Zeitpunkt auch mehrere Fahrzeuge entliehen haben kann.

An der Relationsklasse  $\underline{R}$  sei die Klasse  $NAT$  beteiligt, und zwar sei

$$R \subseteq K_1 \times \dots \times K_{n-1} \times NAT \text{ mit } p_n : R \rightarrow NAT$$

Wenn an der Kante, mit der  $p_n$  im Diagramm dargestellt wird, ein "L" in einem Kreis annotiert ist, dann ergibt  $v_{\{p_1, \dots, p_{n-1}\}, \{p_n\}}(R; x_1, \dots, x_{n-1})$  die ersten  $m$  natürlichen Zahlen. (s. Projektion "lfd. Nr." in Abb. 2) Der Bestand  $R$  von  $\underline{R}$  lässt sich also auf Listen aufteilen.

### 3.3 Rollen

Eine Auswertung mit Gleichheit  $r = v_{\{p_{u,j}\}, \{p_{b,k}\}}(R; x_{u,j})$  der Relationsklasse  $\underline{R}$  ist eine Rolle. Die Schemaklasse  $\underline{K_{u,j}}$  ist die Urbildklasse und die Schemaklasse  $\underline{K_{b,k}}$  ist die Bildklasse von  $r$ .  $p_{u,j}$  ist die Urbildprojektion und  $p_{b,k}$  ist die Bildprojektion von  $r$ . Alle Rollen sind mengenwertig.

Eine Rolle, deren Bildklasse eine Wertklasse ist, ist ein Attribut. Ein Identifikator ist also in diesem Ansatz kein Attribut. Einem Attribut kann eine abgeleitete Relationsklasse zugrunde liegen; dann handelt es sich um ein abgeleitetes Attribut, das durch einen Schrägstrich als solches im Diagramm kenntlich gemacht wird (s.z.B. "Rechnungsbetrag" in Abb.2).

Die für eine Relationsklasse  $\underline{R}$  gegebenen Projektionen, Selektionen, Rollen und Kardinalitätenbeschränkungen gehören zum Klassenschema von  $\underline{R}$ .

Der Name einer Rolle einer 2-stelligen Relationsklasse kann an die Kante geschrieben werden, die für ihre Bildprojektion steht.

## 4 Spezialisierung / Generalisierung

Gegeben sei eine Schemaklasse  $\underline{Q}$  und eine nicht-leere Menge von Schemaklassen  $\underline{U_i}$  für die gilt:  $\underline{U_i} \subseteq \underline{Q}$ .  $S = (\underline{Q}, \{\underline{U_1}, \dots, \underline{U_i}, \dots, \underline{U_n}\})$  heißt Spezialisierung (oder Generalisierung);  $\underline{Q}$  ist die Oberklasse, die  $\underline{U_i}$  sind die Unterklassen von  $S$ . Hat  $S$  mindestens zwei Unterklassen und sind deren Bestände paarweise disjunkt, so heißt  $S$  eine Spezialisierung mit disjunkten Unterklassen; ansonsten hat  $S$  möglicherweise nicht-disjunkte Unterklassen. Hat  $S$  mindestens zwei Unterklassen und ist jedes Exemplar der Oberklasse Exemplar mindestens einer Unterklasse, so heißt  $S$  vollständig, im anderen Fall ist sie möglicherweise unvollständig.

Ein begriffliches Schema (engl.: *conceptual schema*) ist eine Menge von Klassenschemata und eine Menge von Spezialisierungen.

Eine Spezialisierung mit genau einer Unterklasse wird durch einen Pfeil dargestellt, der vom Symbol der Unterklasse zum Symbol der Oberklasse gerichtet ist. Eine Spezialisierung mit mehr als einer Unterklasse wird durch einen Halbkreis dargestellt, der mit jedem Symbol einer Unterklasse mit einer (nicht-gerichteten) Kante verbunden ist. Vom Halbkreis zum Symbol der Oberklasse der Spezialisierung ist ein Pfeil gerichtet. Der Halbkreis enthält Buchstaben, die darüber Auskunft geben, ob die Unterklassen paarweise disjunkt ("d") oder möglicherweise nicht-disjunkt ("o" – overlapping) sind und ob

die Spezialisierung vollständig ("v") oder unvollständig ("u") ist. (vgl. die Spezialisierung mit der Oberklasse "Kunde" in Abb. 2 bzw. die Spezialisierung mit der Oberklasse "Mitarbeiter" in Abb. 3)

In Abb. 3 ist "Betriebszugehörigkeit" Relationsklasse und Oberklasse einer Spezialisierung, die Folgendes bedeutet: Zu irgendeinem Zeitpunkt der Betriebszugehörigkeit eines Mitarbeiters gehört er dem Betrieb entweder als leitender Angestellter oder als technischer Angestellter oder als Verwaltungsangestellter an.

## 5 Intervalle

Ein Intervall  $I_k$  enthält entweder genau ein *tick* als Element oder  $I_k$  ist eine Menge von unmittelbar aufeinander folgenden *ticks*:

$$I_k = \{t_{k_1}, \dots, t_{k_i}, t_{k_{i+1}}, \dots, t_{k_m}\} = [t_{k_1}, \dots, t_{k_n}]$$

Wie üblich, heißt  $t_{k_1}$  der linke Rand und  $t_{k_m}$  der rechte Rand von  $I_k$ . Die Pseudo-Größenangaben  $-\infty$  und  $\infty$  sind ebenfalls zugelassen:  $-\infty$  ist nur als linker Rand,  $\infty$  ist nur als rechter Rand erlaubt. Jede der beiden Größenangaben kann in höchstens einem Intervall eines Bestandes vorkommen.

Die Exemplare einer Klasse von Größenordnungen können zu paarweise disjunkten Intervallen  $I_k$  zusammengefasst werden.

Der Bestand einer Klasse von Größenangaben  $\underline{G}$  ist die Vereinigung von paarweise disjunkten Intervallen  $I_k$ , die die Exemplare von  $\underline{G}$  enthalten:

$$G = I_1 \cup \dots \cup I_n$$

Das bedeutet, dass die Angabe der Ränder der  $I_k$  zur Darstellung des Bestandes einer Klasse von Größenangaben ausreicht. Wenn  $\underline{G}$  Urbildklasse einer Rolle  $r$  ist, so wird ein Wert  $w$  von  $r$  für ein Intervall  $I_k$  angegeben.  $w$  ist dann der Wert von  $r$  für alle Exemplare von  $\underline{G}$ , die zum Intervall  $I_k$  gehören. Ein Intervall ist in diesem Ansatz kein Exemplar einer Werteklasse; Intervalle werden allerdings häufig zur leichteren Darstellung des Bestands einer Klasse von Größenangaben verwandt werden.

Die Exemplare einer Klasse von Größenordnungen  $\underline{G}$  werden zu Intervallen zusammengefasst, wenn im Symbol von  $\underline{G}$  in eckigen Klammern zwei Zeichenfolgen angegeben werden, die die Namen der Intervallgrenzen sind. (s.z.B. das Symbol der Klasse "ET" in Abb. 2)

## 6 Intervallteilbestände

An der Relationsklasse  $\underline{R}$  sei eine Klasse von Größenangaben  $\underline{G}$  beteiligt, und zwar sei

$$R \subseteq K_1 \times \dots \times K_{n-1} \times G \text{ mit } p_n : R \rightarrow G$$

Diejenigen Exemplare  $g$  von  $\underline{G}$ , für die es ein Tupel  $(x_1, \dots, x_{n-1}, g) \in R$  mit  $x_i \in K_i$  gibt, lassen sich ebenfalls zu paarweise disjunkten Intervallen  $I_k$  zusammenfassen. Die Menge  $R_k(x_1, \dots, x_{n-1}) = \{(x_1, \dots, x_{n-1}, t_{k,1}), \dots, (x_1, \dots, x_{n-1}, t_{k,m})\}$  mit  $I_k = [t_{k,1}, \dots, t_{k,m}]$  heißt der  $k$ -te, zu  $(x_1, \dots, x_{n-1})$  gehörende Intervallteilbestand von  $\underline{R}$ . Analog oben reicht es zur Angabe von  $R_k(x_1, \dots, x_{n-1})$  aus, die Intervallgrenzen von  $I_k$  anzugeben. So kann z.B. dargestellt werden, dass auf alle Rechnungsbeträge aus einem Intervall ein bestimmter Rabattsatz angewandt wird (Abb. 2).

Die Namen der Intervallgrenzen können in eckigen Klammern an der Kante notiert werden (s.z.B. die Projektionen "Zugehörigkeit" in Abb. 3). Ist dies nicht der Fall, so wird der Bestand von  $\underline{R}$  genau dann in Intervallteilbestände aufgeteilt, wenn im Symbol von  $\underline{G}$  die Namen von Intervallrändern angegeben werden.

Es wird in eckigen Klammern an der Kante für ein  $p_n$  notiert, wieviele Intervallteilbestände zu einem Tupel  $(x_1, \dots, x_{n-1})$  erlaubt sind. So wird z.B. in Abb. 2 dargestellt, dass zu einer Rechnungsposition genau ein Intervallteilbestand von "Ausleihe" gehört. Ist die Bildklasse von  $p_n$  eine Klasse von Größenangaben, so wird durch die Kardinalitätenbeschränkung in eckigen Klammern ausgedrückt, wieviele Intervalle zu einem Tupel  $(x_1, \dots, x_{n-1})$  gehören dürfen.

Der Bestand von  $\underline{R}$  kann auf unterschiedliche Weise zu paarweise disjunkten Intervallteilbeständen zusammengefasst werden, was an folgendem Beispiel verdeutlicht werden soll:

Beispiel (Abb. 3): Zu jedem Zeitpunkt der Zugehörigkeit eines Angestellten zu einem Betrieb gehört er genau einer Abteilung an. Die Zeitintervalle der Betriebszugehörigkeit und der Abteilungszugehörigkeit des Angestellten brauchen nicht übereinzustimmen. Hier erscheint es notwendig, zwei möglicherweise verschiedene Aufteilungen des Bestandes der Schemaklasse "Betriebszugehörigkeit" zuzulassen. Für die Aufteilung "Abteilungszugehörigkeit" ist die Rolle "Abteilung" gegeben. Das bedeutet, dass alle Elemente eines Intervallteilbestands von "Abteilungszugehörigkeit" denselben Wert der Rolle "Abteilung" haben.

Eine Aufteilung in Intervallteilbestände (und auch in Intervalle), deren Name nicht der Name der zu Grunde liegenden Schemaklasse  $\underline{K}$  ist, wird als Rechteck dargestellt, das mit dem Namen der Aufteilung beschriftet ist und von dem ein gestrichelter Pfeil zum Symbol von  $\underline{K}$  gerichtet ist. (s.z.B. "Abteilungszugehörigkeit" in Abb. 2)

Ein Fahrzeug kann nur zu einem Zeitpunkt entliehen sein, in dem es im Besitz des Autovermieters ist. Ein Kunde kann nur zu einem Zeitpunkt ein Fahrzeug entleihen, in dem er als Kunde geführt wird. (Existenzzeitbeschränkung, lifespan constraint nach [EEK91])

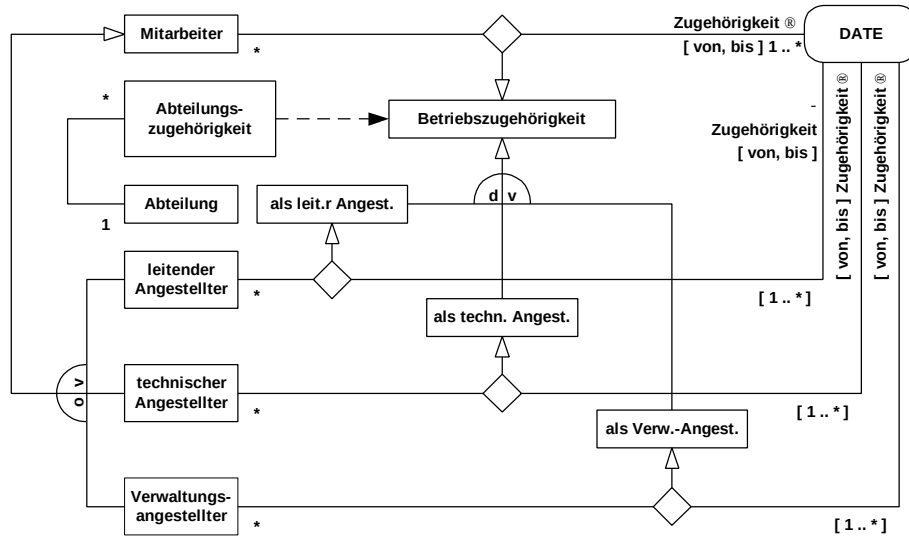


Abb. 3: begriffliches Schema für ein einfaches Personalinformationssystem

Definition: Gegeben seien zwei Relationsklassen  $\underline{R}$  und  $\underline{R}'$ . Für jede einzelwertige Projektion  $p_i$  von  $\underline{R}$  ( $1 \leq i \leq n$ ) gibt es eine ebenfalls einzelwertige Projektion  $p_i'$  von  $\underline{R}'$ , wobei die Namen und die Bildklassen von  $p_i$  und  $p_i'$  übereinstimmen. (Zu  $\underline{R}'$  können weitere einzelwertige Projektionen gehören.).  $\underline{R}'$  heißt in  $\underline{R}$  projektiv enthalten, wenn  $p_{1,\dots,n}'(R') \subseteq p_{1,\dots,n}(R)$  ist. Wenn  $\underline{R}'$  in  $\underline{R}$  projektiv enthalten ist und die gemeinsame Bildklasse eines Projektionenpaars  $p_i$  und  $p_i'$  eine Klasse von Zeitangaben ist, dann ist die Existenzzeit der Exemplare von  $\underline{R}'$  durch die Exemplare von  $\underline{R}$  beschränkt. Wie im Beispiel oben kann die Existenzzeit der Exemplare einer Relationsklasse von den Exemplaren mehrerer Relationsklassen beschränkt sein.

Dies wird durch einen strichpunktierten Pfeil ausgedrückt, der vom Symbol von  $\underline{R}'$  zum Symbol von  $\underline{R}$  gerichtet ist. In Abb. 2 ist z.B. die Existenzzeit der Ausleihe eines Fahrzeugs an einen Privatkunden sowohl durch die Existenzzeit des Fahrzeugs als auch durch die Existenzzeit des Kunden beschränkt

Insbesondere zur Darstellung von raum-zeitlichen Quadern (vgl. [Ta97]) wird nun zugelassen, dass an einer Relationsklasse mehr als eine Klasse von Größenangaben beteiligt sei.

An der Relationsklasse  $\underline{R}$  seien  $m$  Klassen von Größenangaben  $\underline{G}_1, \dots, \underline{G}_m$  beteiligt, und zwar sei

$$R \subseteq K_1 \times \dots \times K_{n-m} \times G_1 \times \dots \times G_m$$

$$\text{mit } 1 \leq i \leq (n-m) \Rightarrow p_i : R \rightarrow K_i \text{ und } (n-m) < i \leq n \Rightarrow p_i : R \rightarrow G_{i-(n-m)} .$$

Diejenigen Tupel  $(g_1, \dots, g_m) \in G_1 \times \dots \times G_m$ , für die es ein Tupel

$(x_1, \dots, x_{n-m}, g_1, \dots, g_m) \in R$  mit  $x_i \in K_i$  gibt, lassen sich in Verallgemeinerung des ein-dimensionalen Falls zu paarweise disjunkten,  $m$ -dimensionalen Quadern  $Q_k$  mit den orthogonalen Kanten  $[t_{1,k,1}, \dots, t_{1,k,m_1}], [t_{2,k,1}, \dots, t_{2,k,m_2}], \dots, [t_{m,k,1}, \dots, t_{m,k,m_m}]$  zusammenfassen. Die Menge  $R_k(x_1, \dots, x_{n-m}) = \{(x_1, \dots, x_{n-m}, t_{1,k,1}, \dots, t_{m,k,1}), \dots, (x_1, \dots, x_{n-m}, t_{1,k,m_1}, \dots, t_{m,k,m_m})\}$  heißt wie oben der  $k$ -te, zu  $(x_1, \dots, x_{n-m})$  gehörende Intervallteilbestand von  $\underline{R}$ . Analog oben reicht es zur Angabe von  $R_k(x_1, \dots, x_{n-m})$  aus, die Intervallgrenzen von  $Q_k$  anzugeben.

## 7 Zusammenfassung

Im hier vorgestellten SchemaKonstruktionsAnsatz sind Werte Exemplare von Schemaklassen. Damit sind solche Meta-Objekte entbehrlich, die eigens zur Darstellung einer Relationsklassen  $\underline{R}_W$  eingeführt werden müssten, an der Werteklassen beteiligt sind (*qualifiers*, schwache Entitätstypen, Zeitreihen, *snapshot* vs. *lifetime cardinality constraints*). Wenn  $\underline{R}_W$  Oberklasse einer Spezialisierung  $S$  ist, dann gelten für die Exemplare der Unterklassen von  $S$  die zu  $S$  gehörenden, implizit angegebenen Integritätsbedingungen (ggf. paarweise disjunkte Unterklassen, ggf. vollständige Spezialisierung), ohne dass eine neue Art von Integritätsbedingungen eingeführt zu werden bräuchte.

Der Bestand einer Werteklasse lässt sich als Vereinigung von paarweise disjunkten Intervallen darstellen. Damit sind solche in der Praxis gebräuchlichen Tabellen im SKA darstellbar, in denen allen Werten eines Intervalls ein Objekt eindeutig zugeordnet wird.

Der Bestand einer Relationsklasse  $\underline{R}_W$  kann auf unterschiedliche Weise zu ein- oder mehrdimensionalen Intervallteilbeständen zusammengefasst werden. Auf der Basis einer solchen Aufteilung wird allen Elementen eines Intervallteilbestands  $Q_k$  dasselbe Objekt zugeordnet.

Die oben definierte, mengenwertige Projektion wird dazu benutzt, zu definieren, wann eine Relationsklasse  $\underline{R}'$  in einer anderen Relationsklasse  $\underline{R}$  projektiv enthalten ist. Diese Relation kann ihrerseits dazu verwandt werden, die Beschränkung der Existenzzeit der Exemplare von  $\underline{R}'$  von den Exemplaren von  $\underline{R}$  zu erklären.

Fundamentales Darstellungsmittel des SKA ist die Schemaklasse mit ihrem Bestand; alle weiteren Meta-Objekte werden auf dem Begriff der Schemaklasse aufbauend und unter Verwendung der Mengenlehre eingeführt. Eine Einführung von Darstellungsmitteln unabhängig vom Begriff der Schemaklasse (*ad hoc*) ist entbehrlich.

## Literatur

- [Ch76] Chen, P. P.-S.: The Entity-Relationship Model – Toward a Unified View of Data. *ACM Trans Database Systems* 1 (1976) 9-30
- [EEK91] Elmasri, R.; El-Assal, I.; Kouramajian, V.: Semantics of temporal data in an extended ER model. *in: The Core of Conceptual Modeling* (H. Kangassalo; Hrsg.) Elsevier, Amsterdam, 1991; S. 239-254
- [EN00] Elmasri, R.; Navathe, S.B.: *Fundamentals of Database Systems*. Addison-Wesley, Reading, Mass., 2000
- [Em97] Embley, D.W.: *Object Database Development*. Addison-Wesley, Reading, Mass., 1997
- [GJ99] Gregersen, H.; Jensen, C.S.: Temporal Entity-Relationship Models – A Survey. *IEEE Trans. Software Engineering* 11 (1999) 464-497
- [LE99] Lee, J.Y.; Elmasri, R.A.: An EER-Based Conceptual Model and Query Language for Time-Series Data. *in: Conceptual Modeling – ER '98* (T.W. Ling, S. Ram, M.L. Lee; Hrsg.) Springer, Berlin, 1999, S. 21-34
- [RP92] Roddick, J.F.; Patrick, J.D.: Temporal Semantics in Information Systems – A Survey. *Informations Systems* 17 (1992) 249-267
- [RJB99] Rumbaugh, J.; Jacobson, I.; Booch, G.: *The Unified Modeling Language Reference Manual*. Addison-Wesley, Reading, Mass., 1999
- [Ta97] Tagashira, T.; Amagasa, T.; Aritsugi, M.; Kanamori, Y.: Interval-Based Representation of Spatio-Temporal Concepts. *in: Advanced Information Systems Engineering* (A. Olivé, J.A. Pastor; Hrsg.) Springer, Berlin, 1997; S. 231-244
- [Th00] Thalheim, B.: *Entity-Relationship Modeling*. Springer, Berlin, 2000