

Auf dem Weg zur Universellen Modellierungssprache

Thomas Matzner
Berater für Systemanalyse
Beltweg 22
80805 München
tamatzner@acm.org
www.tamatzner.de

In den letzten Jahrzehnten haben sich etliche Modellierungstechniken etabliert, etwa E/R-Modellierung, Strukturierte Analyse (SA), eine Vielzahl objektorientierter Techniken (OO) und Geschäftsprozeßmodellierung (GPM). Die Entscheidung für eine dieser Techniken erscheint zumeist wie eine Schicksalsfrage: mit ihr legt man sich auf eine Notation fest, auf bestimmte Aussagen, die in Modellen vorkommen können, ja sogar auf eine Art, die Welt zu deuten. Hierfür typisch ist etwa die Frage, ob die Welt (oder Ausschnitte daraus) objektorientiert oder funktionsorientiert „ist“. Brücken zwischen diesen Techniken und ihren Notationen gibt es kaum.

In diesem Beitrag wird ein Ansatz unternommen, die Gemeinsamkeiten der Modellierungssprachen herauszuarbeiten. Vorbild dabei ist die Sprache der Mathematik, die zur Modellierung der unterschiedlichsten Aspekte der Welt verwendet wird und dennoch mit einem überschaubaren Satz von Sprachkonstrukten auskommt. Der Schlüssel zu einer Universellen Modellierungssprache ist eine klare Trennung zwischen Aussagen in ihren Strukturen, ihrer konkreten Notation in Dokumenttypen sowie den Empfehlungen für ihr Einsatzgebiet. Am Ende läßt sich erkennen, daß sich die Modellierungssprachen nicht in ihrer grundsätzlichen Ausdrucksfähigkeit unterscheiden, sondern nur in den Ordnungsprinzipien, nach denen sie die immer wieder gleichen Aussagentypen anordnen.

1 Einleitung

Die Modellierungstechniken sind nicht so homogen wie ihre Bezeichnung vermuten läßt. Zu jeder der genannten Familien E/R, OO, SA und GPM gibt es unzählige Publikationen, in denen Sprachen, Notationen und Vorgehensweisen dargestellt werden.

Dieser Beitrag konzentriert sich auf Sprachen und Notationen und läßt das eigentlich Methodische, also die Frage, wie man zu einem guten Modell kommt, außer acht. Die bestehenden Sprachen und Notationen wurden von ihren Autoren fast ausnahmslos von Grund auf neu beschrieben, als ob es auf dem Gebiet der Modellierung nicht schon unzählige Sprachen gegeben hätte. Eine neue Sprache wurde im besten Fall syntaktisch und semantisch definiert; Bezüge zu ähnlichen Konstrukten bestehender Sprachen wurden nicht hergestellt. So könnte man zu der Auffassung kommen, mit jeder solcherart definierten Sprache sei ein in sich abgeschlossenes Weltbild verbunden, das zu anderen Weltbildern keinen Bezug hat.

Die Vorrangstellung der UML innerhalb der Familie objektorientierter Beschreibungstechniken scheint diese Vielfalt zu reduzieren. Tatsächlich leistet sie allein durch ihre Marktstellung eine Vereinheitlichung innerhalb der objektorientierten Modellierungssprachen: Konstrukte außerhalb der UML haben einfach weniger Durchschlagskraft. Das Problem ist damit jedoch noch nicht gelöst. Zum einen enthält die UML selbst eine Reihe von Konstrukten, deren Verwandtschaft nur unzureichend geklärt ist. Wichtiger noch ist die Frage, wie die UML als objektorientierte Beschreibungsform sich zu anderen weiterhin bedeutenden Beschreibungsformen, etwa der Geschäftsprozeßmodellierung oder der Datenflußmodellierung, verhält.

Eine unnötige Vielfalt von Sprachen und Notationen wirkt hemmend auf den Fortschritt des Software Engineering und auf jeden Anwender der Sprachen: Es besteht die Gefahr, sich zuviel mit der Sprache und zuwenig mit dem Inhalt zu befassen, der aktuell ausgedrückt werden soll. Außerdem ist für jeden Beteiligten (Methodenentwickler, Toolentwickler, Modellierer) die Anzahl miteinander unverbundener Sprachkonstrukte als eine Ressource anzusehen, die Kräfte bindet. Jedes Konstrukt muß erfunden, erlernt, unterstützt und mit anderen verbunden werden.

Deshalb wird in diesem Beitrag ein Ansatz unternommen, die Konstrukte der genannten Sprachfamilien auf eine gemeinsame syntaktische und semantische Basis zu stellen und damit vergleichbar und kombinierbar zu machen. Dies geschieht nach dem Vorbild, das uns die Sprache der Mathematik zur Verfügung stellt. Zentrale Gedanken sind die Entbündelung von abstraktem sprachlichem Konstrukt, konkreter Notation und vor allem von den Einsatzempfehlungen.

2 Das Vorbild: Die Sprache der Mathematik

Wenn es eine Sprache gibt, die ihre Eignung zur Modellierung unterschiedlichster Aspekte der Welt bereits bewiesen hat, ist dies die Sprache der Mathematik. Sie ist das nahezu ausschließliche Ausdrucksmittel der Physik und damit vieler Ingenieur-tätigkeiten. Doch auch alle anderen Naturwissenschaften, die Wirtschaftslehre und die Sozialwissenschaften bedienen sich immer dann der Sprache der Mathematik, wenn formale Modelle gebaut werden sollen.

Würde die Sprache der Mathematik ähnlich entwickelt wie die Modellierungssprachen der Informatik, so müßte vor Betreten eines neuen Anwendungsgebiets zunächst die Sprache erweitert werden. Es müßte je eine mathematische Notation für statische und dynamische Aspekte, für die belebte und unbelebte Welt geben. Und die Mathematiker müßten Kongresse abhalten, in denen darüber gestritten wird, welche dieser Sprachen sich zu welchem Zweck eignet.

Stattdessen kommt der überwältigende Teil der Literatur über Mathematik und all ihre Anwendungen mit einem überraschend kleinen Sprachumfang aus. Dieser umfaßt

- die Sprache der mathematischen Logik,
- die Sprache der Mengen, darauf aufbauend der Relationen und Funktionen
- sowie eine Reihe von Spezialsymbolen für häufig gebrauchte Mengen, Funktionen etc.

Die in der Praxis gebrauchte Sprache der Mathematik ist keineswegs minimal, etwa wird für eine Funktion, die ja ihrerseits eine Menge ist, eine andere Notation verwendet wie für Mengen allgemein. Die Sprache ist jedoch unabhängig von dem Wissensgebiet, auf das die Mathematik gerade angewendet wird. So kann $f(x) = x^2$ sowohl eine Tatsache der Physik, der Chemie oder des Kaufverhaltens einer Bevölkerung, einen statischen oder einen dynamischen Zusammenhang ausdrücken.

Es ist den meisten Benutzern der Sprache der Mathematik gar nicht bewußt, welchen ungemeinen Vorteil diese Überschaubarkeit und Anwendungsunabhängigkeit bietet. Aus $f(x) = x^2$ kann direkt abgelesen werden, daß jedem x genau ein $f(x)$ zugeordnet ist, daß $f(x) \geq 0$ sein muß usw. Man muß sich nicht informieren, ob der Autor dieser Formel Physiker oder Betriebswirt ist oder welche Sprachversion er verwendet. Und die unbestreitbare Schwierigkeit vieler mathematischer Theorien und Texte hat ihre Ursache nicht in der Sprache, deren Beherrschung von jedem Abiturienten verlangt wird, sondern in den jeweils ausgedrückten Inhalten.

Natürlich hat auch die Sprache der Mathematik sich auf einem langen Weg zu der heutigen Eleganz entwickelt. Ursprünglich war die Geometrie tatsächlich das Wissen über bestimmte ebene Figuren; Zahlen dienten zum Abzählen realer Dinge. Für wichtige Funktionen wurden ursprünglich neue Notationen geschaffen, etwa die Bruchstrich-Schreibweise für die Division. Erst eine Reduktion der Notation und eine Abstraktion von der Anwendung mathematischer Notationen in der realen Welt führte zu dem überschaubaren Satz von Sprachkonstrukten, der uns in einem mathematischen Text zumindest sprachlich sofort zurechtfinden läßt.

Auch heute noch gibt es in der Mathematik Dialekte. Die logische Implikation wird in manchen Ländern auch als \therefore geschrieben. Dennoch würde niemand auf die Idee kommen, die weltweit einheitliche semantische Fundierung dieses Konstrukts in Frage zu stellen.

3 Das Referenz-Metamodell

Um dem angestrebten Ziel eines einheitlichen syntaktischen und semantischen Fundaments für Modellierungssprachen näherzukommen, wurden bestehende Modellierungssprachen untersucht und ihre Konstrukte auf ein **Referenz-Metamodell** (RMM) abgebildet. Der wesentliche Gedanke bei dieser Abbildung war die Abstraktion von dem konkreten Anwendungsgebiet bzw. den Anwendungsempfehlungen zugunsten der Frage: *Welche Aussagentypen kommen in einem bestimmten Modelltyp vor und wie sind sie angeordnet?*

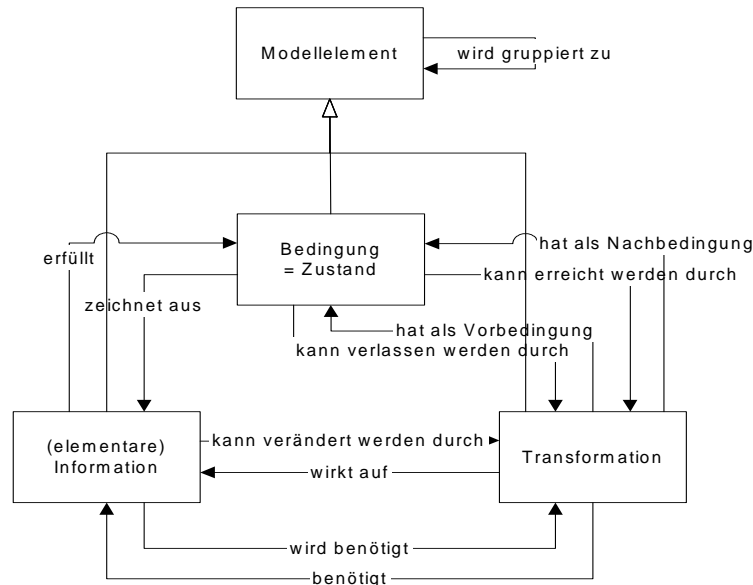
Das RMM entstand somit bottom-up aus den bei der Sichtung der Modellsprachen gefundenen Elementen. Es entstand jedoch gleichzeitig top-down durch eine Beurteilung, welche abstrakten Begriffe zur Klassifikation der unterschiedlichen Aussagentypen dienen können und wie diese abstrakten Begriffe zusammenhängen müssen.

In diesem Aufsatz werden aus Platzgründen nur Submengen der Modellierungssprachen detailliert abgebildet. Die hier beschriebene Technik, Modellierungssprachen durch ein RMM miteinander verträglich zu machen, steht noch an ihrem Anfang. Die bereits geleisteten Abbildungen sollen das Potential einer solchen Technik illustrieren und die Kollegen in Forschung und Praxis zu weiteren Tätigkeiten anregen.

Das RMM wird in der folgenden Abbildung dargestellt. Es enthält Begriffe, deren mathematische Fundierung auf Mengen, Funktionen, Tupel und Prädikate leicht erkennbar ist. Es enthält jedoch nicht einfach nur diese aus der Mathematik ohnehin bekannten Begriffe. Es soll ein Metamodell speziell für die Modellierung im Umfeld von Informationssystemen sein. Dies bedeutet eine spezielle Ausrichtung der enthaltenen Begriffe.

- Auch wenn es meist in der Modellierungs-Literatur nicht erwähnt wird, nehmen die Modelle bereits Rücksicht auf die Begrenzungen durch die Algorithmik. Es fehlen kontinuierliche Vorgänge und dichte Mengen. Alle Strukturen sind diskret, alle Abläufe verlaufen in Schritten.
- Alle genannten Modellierungsfamilien (mit Ausnahme der Datenmodellierung) basieren auf dem Gedanken von Zustandsveränderungen. Es herrscht kein funktionaler Spezifikationsstil, bei dem Argument x und Funktionsergebnis $f(x)$ voneinander separate Elemente sind. Vielmehr werden Transformationen beschrieben, bei denen dasselbe Element durch eine Zustandsveränderung im Lauf der Zeit unterschiedliche Werte erhält.

Das RMM ist in einer ad-hoc-Notation aufgezeichnet und verwendet neutrale Begriffe, um den Verdacht auszusräumen, es gehe darum, eine bestehende Sprache zur Referenzsprache zu ernennen. Natürlich kann es bei Bedarf leicht in E/R- oder UML-Notation überführt werden. Das RMM baut also auf den Begriffen der **Information**, der **Trans-**



Das Referenz-Metamodell

formation sowie der **Bedingung**, gleichbedeutend des **Zustandes** auf.

Mit (elementarer) Information ist solche Information gemeint, die innerhalb der betrachteten Modellierungssprache nicht mehr weiter zergliedert wird. Der Charakter des Metamodells als reines Sprach-Untersuchungsmittel äußert sich darin, daß in seinem Rahmen keine Empfehlungen gegeben werden, welche Informationen in diesem Sinn elementar sein sollen. „Elementare Information“ ist ein Begriff, auf den Modellkonstrukte abgebildet werden. Das können die Attribute in E/R sein, ebenso die Attribute, aber auch die Parameter in OO.

Mit Transformation ist die Veränderung bestimmter Informationen gemeint. Dies können Transformationen aus einem Datenflußdiagramm von SA sein, ebenso Operationen aus OO, Zustandsübergänge, Aktivitäten oder Prozeßschritte.

Eine besondere Erklärung benötigt die Gleichsetzung von Bedingung und Zustand. Im alltäglichen Sprachgebrauch wie in der Modellierung bedeuten diese Begriffe zunächst etwas Verschiedenes. Eine Bedingung *muß* gelten, ein Zustand *kann* angenommen werden. Angewandt auf Information, die transformiert wird, ist es jedoch nützlich, die beiden Begriffe zusammenzulegen und bei Bedarf durch ein Metaattribut zu unterscheiden. Beides sind Prädikate über Informationen. Eine Bedingung ist ein Prädikat, das stets gelten muß; als Zustand bezeichnet, ist es ein Zustand, der nicht verlassen werden kann. Ein Zustand im allgemeinen ist ein Prädikat, das manchmal gilt.

Wenn man bei einer Weiterentwicklung dieses Ansatzes Bedingungen und Zustände durch ein Metaattribut unterscheidet, kann man auch überlegen, das Ereignis, d.h. eine nur kurzzeitig geltende Bedingung, als dritten Wert dieses Metaattributs einzuführen. Damit schafft man die Möglichkeit, zwischen dem Zustand "es regnet" und dem Ereignis "es beginnt zu regnen" zu unterscheiden.

4 Fundierung von Modellierungssprachen im Referenz-Metamodell

Die Abbildung von Modellierungssprachen vollzieht sich auf mehreren Ebenen.

- Das RMM selbst enthält die Grundbegriffe, auf die abgebildet wird. Es ist voll generisch und wird für die Abbildung einer Modellierungssprache nicht verändert.

- Der **Modelltyp** ist eine Abbildung von Begriffen aus einer Modellierungssprache auf das RMM. Bei dieser Abbildung werden für die jeweilige Modellierungssprache spezifische Gruppierungen gebildet und die Begriffswelt der Modellierungssprache verwendet. Für die Abbildung können je nach Modellierungssprache einschränkende Bedingungen gelten, etwa das Datenabstraktionsprinzip bei den OO-Sprachen. Der Modelltyp berücksichtigt noch keine technischen oder kognitiven Begrenzungen etwa der Elemente, die gleichzeitig dargestellt werden können.
- Der **Dokumenttyp** ist die Abbildung eines Modelltyps auf Dokumente. Auf dieser Ebene wird der Informationsumfang von Dokumenten beschrieben, jedoch noch nicht die konkrete Syntax. Hier spielen Begrenzungen von Anzahl und Umfang gleichzeitig darstellbarer Elemente eine Rolle. Dort, wo miteinander zusammenhängende Elemente nicht gleichzeitig dargestellt werden können, kann ihr Bezug nicht mehr in einem Dokument hergestellt werden. Es muß ein **Fernbezug** hergestellt werden, der je nach technischem Träger des Dokuments unterschiedlich komfortabel ausfällt. Fernbezüge auf Papier sind Verweise, auf elektronischen Medien können sie Hyperlinks o.ä. sein. Im Gegensatz zum Fernbezug heißt die Darstellung einer Metabeziehung innerhalb desselben Dokuments **Nahbezug**. Weiterhin wird beim Dokumenttyp für die in ihm enthaltenen Elemente eine Beschreibungstiefe festgelegt.
- Der **Notationstyp** gibt die konkrete Syntax für einen Dokumenttyp an. Seine Erarbeitung ist am wenigsten aufschlußreich für das Verständnis einer Modellierungssprache und wird in vorliegendem Aufsatz deshalb stets übersprungen.

Von grundlegender Bedeutung für die Untersuchung ist eine klare Unterscheidung zwischen Modell- und Dokumenttyp. Der Modelltyp bestimmt die grundsätzliche Ausdrucksfähigkeit einer Modellierungstechnik und enthält methodische Forderungen zur Anordnung von Aussagen, etwa das Datenabstraktionsprinzip. Der Dokumenttyp ist deshalb nicht weniger wichtig; denn er bestimmt durch die Zuordnung bestimmter Modellteile zu Dokumenten, wie handhabbar Modelle sein können. Bildschirme und Papierseiten können nur begrenzte Sprachäußerungen darstellen, Menschen können nur begrenzt viel Text und Grafik aufnehmen. Im Zeitalter praktisch unbegrenzter CPU- und Speicherkapazität gilt weiterhin: *Die Beschränkung des Raums für Nahbezüge ist die einzige limitierte Ressource bei der Modellierung.*

Die **Beschreibungstiefe** ist ein weiterer Tribut an die Begrenztheit der Ressourcen und die Aufnahmefähigkeit der Menschen. Sie kann die Werte „formal spezifiziert“, „halbformal spezifiziert“ (etwa Pseudocode), „informell beschrieben“, „benannt“ und „unterstellt“ annehmen. Unterstellt bedeutet dabei, daß ein nach dem RMM notwendiges Beschreibungselement noch nicht einmal in Form einer Bezeichnung vorhanden ist.

5 Abbildung objektorientierter Modellierungssprachen auf das Referenz-Metamodell

Die meisten Modellierungstechniken sind in der Literatur anhand ihrer Dokumenttypen beschrieben. In der praktischen Anwendung baut man deshalb zunächst die Dokumenttypen auf. Wir beginnen mit dem Dokumenttyp Klassendiagramm, noch ohne Generalisierung/Spezialisierung (Abbildung „Der Dokumenttyp Klassendiagramm“).

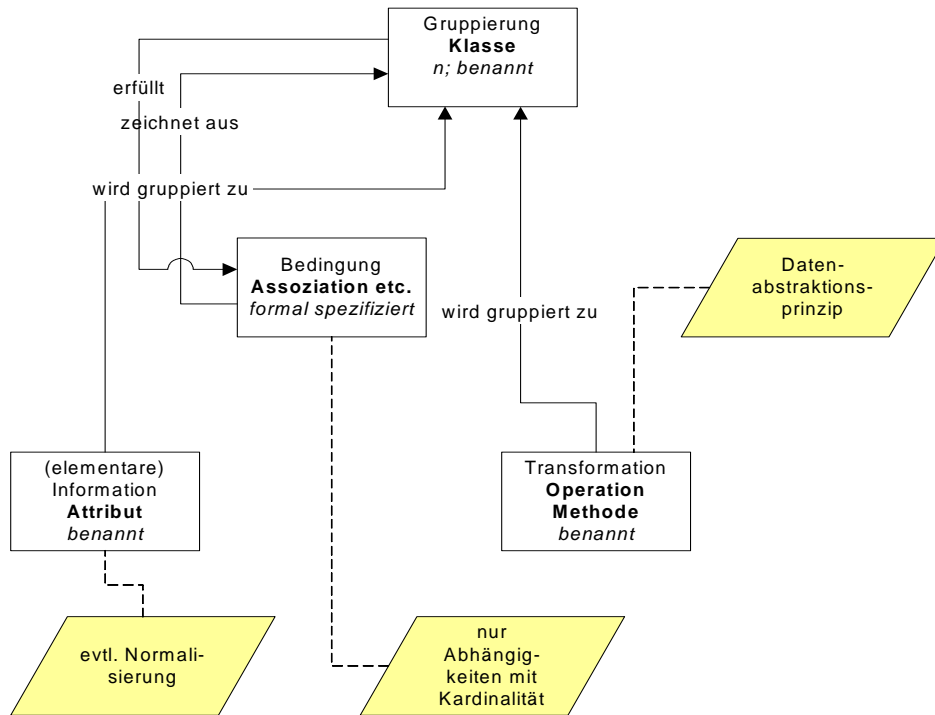
„Attribut“ wurde auf „elementare Information“ abgebildet, „Operation“ auf „Transformation“, „Assoziation“ auf „Bedingung“. Wir müssen jedoch wahrnehmen, daß nicht alle möglichen Bedingungen in einem Klassendiagramm untergebracht werden können, sondern nur diejenigen, die Abhängigkeiten zwischen Klassen beschreiben. Die Operationen müssen dem Datenabstraktionsprinzip genügen, was allerdings auf der Ebene des Klassendiagramms nicht erkennbar ist, da dort gar nicht dargestellt wird, welche Attribute von welchen Operationen benötigt werden.

Bei den abgebildeten Elementen vermerkt ist jeweils die Beschreibungstiefe, hier meist "benannt". Das Klassendiagramm beschränkt sich also auf das übersichtsartige Benennen und erlaubt keine Details zu spezifizieren, außer bei den Assoziationen. Sie werden sogar formal spezifiziert.

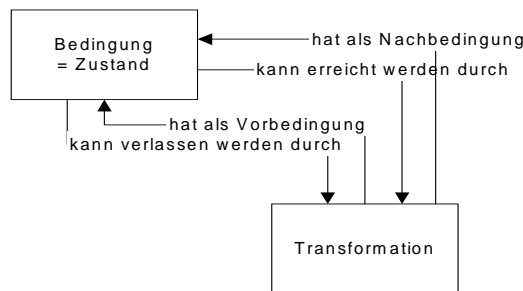
Die Kardinalität n bei der Klasse sagt aus, daß in einem Dokument mehrere Klassen enthalten sein können. Wenn sich diese Kardinalität auf andere Modellelemente einfach überträgt (alle Attribute, Operationen etc. zu den dargestellten Klassen), wird sie bei diesen nicht wiederholt.

Man erkennt durch Vergleich mit dem RMM, daß lange noch nicht alle Aussagentypen, die in Modellen überhaupt vorkommen, im Klassenmodell enthalten sind. Also erweitern wir es, im ersten Schritt um Zustandsübergangsdiagramme. „Zustand“ wird auf „Bedingung“, „Zustandsübergang“ auf „Transformation“ abgebildet (Abbildung „Der Dokumenttyp Zustandsdiagramm“).

Diese beiden Abbildungen verdeutlichen ein wesentliches Prinzip der vorgestellten Technik zur Integration von Modellierungssprachen. Die Fundierung im RMM erlaubt es uns, Modellteile syntaktisch und semantisch zusammenzufügen, und zwar unter objektiven Kriterien. Überall dort, wo Modelle dieselben Referenz-Konstrukte enthalten, können sie zusammengefügt werden. Das gilt im vorliegenden Fall für das Modellkonstrukt Transformation. Verwendet man also Klassen- und Zustandsdiagramme zusammen, erhält man insgesamt eine Dokumentenstruktur, bei der die



Der Dokumenttyp Klassendiagramm



Der Dokumenttyp Zustandsdiagramm

Dokumente zwar physisch getrennt, aber durch Fernbezüge auf Transformationen klar miteinander verbunden sind (Abbildung „Vereinigung von Klassen- und Zustandsdiagramm“).

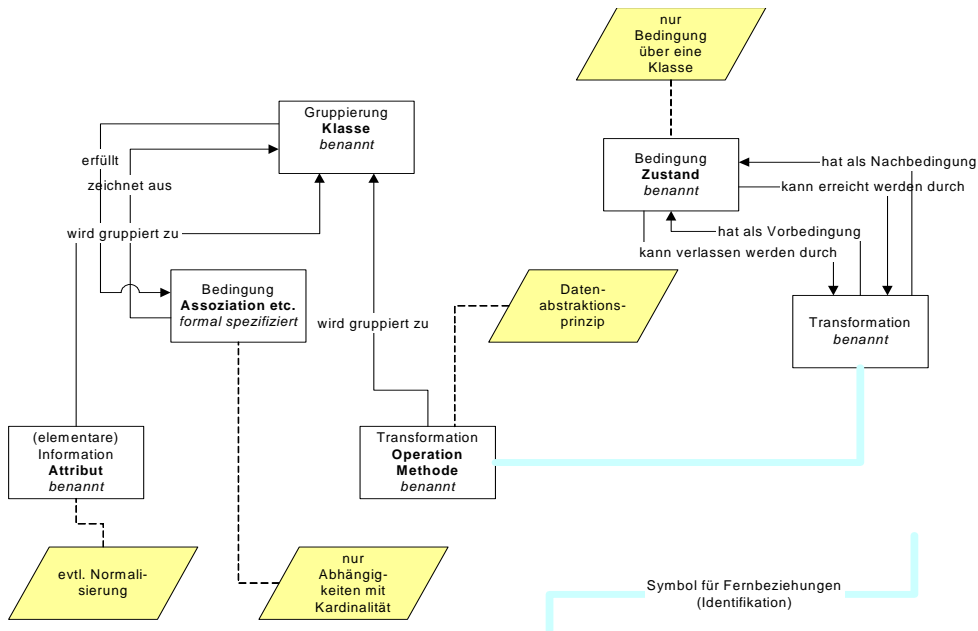
Nun können wir den Schritt zum Modelltyp vollziehen. Dieser umfaßt die Vereinigung aller in allen Dokumententypen enthaltenen Aussagen. Man kommt von den Dokumententypen zum Modelltyp, indem man

- alle durch Fernbezüge miteinander verbundenen Elemente miteinander identifiziert und
- als Beschreibungstiefe eines Elements stets die maximale in einem der Dokumententypen vorhandene einsetzt.

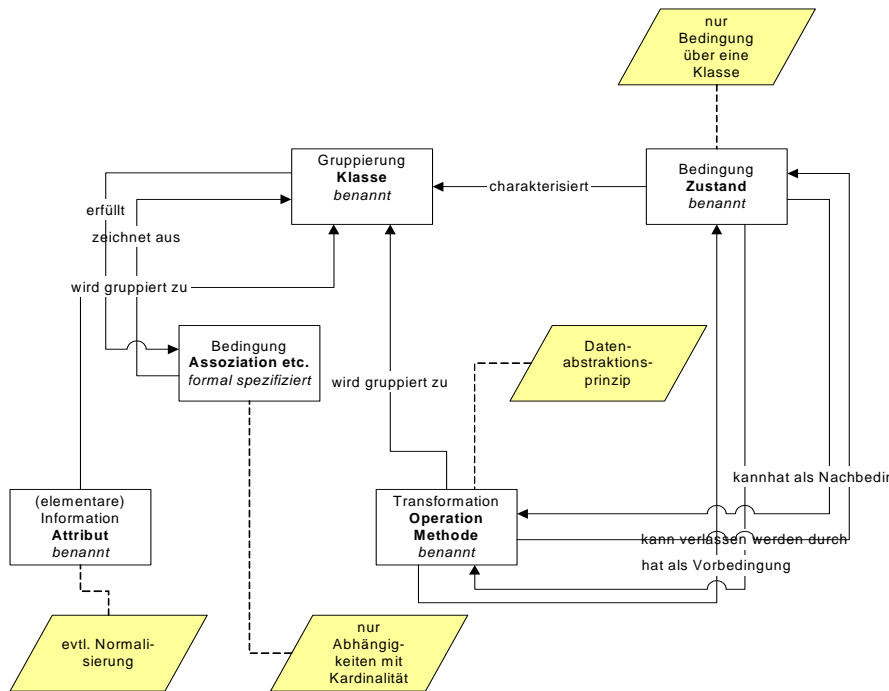
Der Modelltyp, der sich aus Kombination von Klassen- und Zustandsdiagramm ergibt, ist im Anschluß dargestellt (Abbildung „Modelltyp Klassen- und Zustandsübersicht“). Auch er enthält noch nicht alle überhaupt möglichen Modellelemente. Eine weitere Integration von Modellteilen, die hier im Detail nicht dargestellt werden kann, führt schließlich zu einer im Hinblick auf die enthaltenen Aussagen kompletten Modellierung nach objektorientierten Prinzipien. Der zentrale Dokumententyp, der zu dieser Komplettierung führt, ist eine Spezifikation der Operationen, d.h. ihrer Wirkung auf die Attribute sowie ihrer Delegationsbeziehungen. Ob eine solche Spezifikation etwa algebraisch, im Pre-/Postcondition-Stil oder operativ aufgeschrieben wird, ist für die grundsätzliche Ausdrucksfähigkeit zweitrangig; wichtig ist, daß die Beziehungen von Operationen auf die Attribute und auf andere Operationen hergestellt sind.

6 Vergleich der Ausdrucksfähigkeit unterschiedlicher Modellierungssprachen

Ein Einsatzgebiet der beschriebenen Abbildung von Sprachen auf das RMM ist der Vergleich zwischen Sprachen bzw. Sprachfamilien.



Vereinigung von Klassen- und Zustandsdiagramm



Modelltyp Klassen- und Zustandsübersicht

Auf ähnliche Weise wie im vorigen Abschnitt beschrieben, kann man z.B. auch für die in den SA-Techniken verwendeten Sprachmittel einen Dokumenttypen aufstellen und einen umfassenden Modelltyp ermitteln. Nimmt man E/R-Modell, Data Dictionary, Datenflußdiagramm und Minispec für die Funktionen zusammen, so erhält man den Modelltyp SA-Modell (Abbildung „Der Modelltyp SA-Modell“).

Man erkennt, daß dieser Modelltyp, ebenso wie der Modelltyp Klassenmodell, alle im RMM vorhandenen Aussagentypen zuläßt. Allerdings fordert er eine andere Anordnung der Aussagen. Bei den Klassenmodellen war das Datenabstraktionsprinzip strukturbildend. Es forderte, daß der Teil der Wirkung einer Operation, der nicht Attribute der eigenen Klasse betrifft, durch einen Bezug auf eine andere Operation dargestellt wird. Diese Forderung wird in der SA-Sprachfamilie nicht aufgestellt. Für die Dokumenttypen bedeutet dies, daß bei OO-Dokumenten Nahbezüge zwi-

7 Umgang mit Modellerweiterungen

Die beschriebenen Diagrammumfänge waren gegenüber den im realen Gebrauch befindlichen reduziert. In diesem Abschnitt wird skizziert, wie mit den hier noch fehlenden Darstellungsmitteln umgegangen werden kann.

Das augenfälligste fehlende Beschreibungsmittel war das der **Generalisierung und Spezialisierung**. Die Abstraktion auf Aussagen und ihre Struktur gibt einen einfachen Weg zum Umgang mit diesem Sprachmittel vor. Wird eine Oberklasse O in Unterklassen U1, U2 spezialisiert, bedeutet das für die Aussagen dasselbe, als würden zwei Klassen gebildet, von denen die eine die Aussagenmenge von O und U1, die andere von O und U2 enthält. Der Vorzug dieser Interpretation ist, daß sie die Generalisierungstechnik viel weiter ausdehnt, als das in den OO-Verfahren üblich ist. Man kann sie auf beliebige Aussagen-Zusammenfassungen anwenden, nicht nur auf Attribut- und Operationsbeschreibungen einer Klasse. Die aus den Vor- und Nachbedingungen von Operationen entstandenen Zustandsdiagramme beispielsweise lassen sich auf diese Weise ebenfalls generalisieren.

Weitere nicht behandelte Darstellungsmittel sind **Rucksäcke**, d.h. Modellsprachkonstrukte, die keine tatsächliche Erweiterung darstellen, sondern lediglich Notationen für bereits enthaltene Konstrukte bieten. Ein Beispiel hierfür sind die unterschiedlichen Darstellungsmittel für Operationen und Bedingungen in Statecharts. Für den Umgang mit solchen Konstrukten wird die **transformative Semantik** empfohlen, die es erlaubt, eine Sprache um Konstrukte zu erweitern, ohne eine von Grund auf neue Semantik für sie angeben zu müssen. Das neue Konstrukt wird transformativ als Ausdruck über bereits bestehenden Konstrukten definiert und damit die Abbildung auf das RMM geleistet.

In der UML spielen **exemplarische Dokumenttypen**, etwa die Sequenzdiagramme, eine wichtige Rolle. Auch sie führen keine neuen Aussagentypen ein, sondern machen Aussagen über Informationen, Transformationen und Bedingungen, allerdings ohne den Anspruch, daß die in einem Dokument gemachten Aussagen vollständig sind. Dies läßt sich durch ein Meta-Attribut mit der Bedeutung "vollständig" abbilden.

8 Potential der neuen Technik

Die Abbildung von Modellierungssprachen auf ein RMM öffnet der Modellierungs-Gemeinde eine Reihe von Perspektiven. Die wichtigste ist eine Objektivierung des Vergleichs von Sprachen und Sprachfamilien. Am wichtigsten deshalb, weil jeder Modellierer damit in die Lage versetzt wird, pragmatisch diejenigen Darstellungsmittel auszuwählen, die ihm zur Verfügung stehen und für den konkreten Zweck nützlich erscheinen, unabhängig davon, ob es eine Autor gibt, der genau diese Kombination von Sprachmitteln in einem Methodenbuch erlaubt.

Insgesamt führt diese Technik zu einer Ent-Personalisierung von Darstellungsmitteln. Ihre Bedeutung ist durch ihre mathematische Fundierung bestimmt und nicht durch eine Aussage eines bestimmten Autors. Die Empfehlung bestimmter Sprachmittel für einen Zweck bleibt natürlich legitim, aber sie ist streng zu trennen von der Definition des Sprachmittels.

Die Hersteller von Tools zur Unterstützung der Modellierung haben womöglich den höchsten rechenbaren Nutzen. Das RMM ist das einzige Modell, das ein Tool in seinem Repository wirklich fest verdrahtet abbilden muß. Alle Darstellungsformen können ohne Erweiterung des Repository darauf aufgesetzt werden. Als zusätzliche, heute nur sehr schwach angebotene Funktionalität von Tools kann das automatische Umsetzen von Darstellungsformen, z.B. von OO-Spezifikationen zu Datenflußdiagrammen, angeboten werden.

Ein zweifelhafter Nutzen entsteht für die Schöpfer von Modellierungssprachen. Auf der einen Seite wird ihre Aufgabe vereinfacht, da sie bei der Erfindung neuer Konstrukte eine klare Basis haben, auf der sie ihre Syntax und Semantik definieren können. Allerdings könnte sich bald zeigen, daß das Neuerfinden von Sprachkonstrukten keine vorrangige Aufgabe mehr ist. Die Rolle des Sprachentwerfers verliert dadurch an Bedeutung.

9 Ausblick

Der Anspruch, eine einheitliche Basis für viele Modellierungssprachen geschaffen zu haben, kann in einem Aufsatz naturgemäß nur andeutungsweise belegt werden. Auch steht die beschriebene Technik erst an ihrem Anfang. Wer diese Ideen für fruchtbar hält, findet viele Gebiete, auf denen sie weiter entwickelt werden können, etwa:

- Die vollständige Abbildung existierender Modellierungssprachen auf das RMM.
- Die Erweiterung des RMMs z.B. um Metaattribute für den Charakter von Informationen und Bedingungen.
- Die Überlegung, ob andersartige RMMs Vorteile bringen.
- Aufstellen von RMMen für die Detailspezifikation von Transformationen. Hierdurch würde ein Vergleich unterschiedlicher Spezifikationsstile für Transformationen möglich.
- Formale Regeln für das Zusammenführen von Dokument- und Modelltypen.
- Formale Regeln für das Überführen von Modellen aus einem Dokument- oder Modelltyp in einen anderen.