

Ein UML-basiertes Metamodell des Memorandums zur vereinheitlichten Spezifikation von Fachkomponenten

Peter Fettke, Peter Loos, Kai Pastor

Johannes Gutenberg-Universität Mainz
Lehrstuhl für Wirtschaftsinformatik und BWL
ISYM – Information Systems & Management
D-55099 Mainz, Germany
{fettke,loos,pastor}@isym.bwl.uni-mainz.de
www.isym.bwl.uni-mainz.de
Tel. +49 6131 39-23051, Fax. -22185

Abstract: Metamodelle sind allgemein bekannte Instrumente zur Systemgestaltung. In diesem Beitrag wird ein UML-basiertes Metamodell für die „Vereinheitlichte Spezifikation von Fachkomponenten“ der Arbeitsgruppe „Komponentenorientierte betriebliche Anwendungssysteme“ der Gesellschaft für Informatik (Memorandum) vorgestellt. Dabei werden die Ableitungsschritte expliziert und die aus Unschärfen im Memorandum entspringenden Probleme diskutiert. Ausgehend von diesen Ergebnissen werden Gestaltungsempfehlungen für die Weiterentwicklung des Memorandums abgeleitet.

1 Ausgangspunkt

Seit einiger Zeit verfolgt die (Wirtschafts-)Informatik die Idee, Anwendungssysteme aus vorgefertigten Software-Komponenten zu montieren. Von diesem Ansatz der komponentenorientierte Software-Entwicklung werden – in Analogie zu traditionellen Industrien – Verbesserungen der Effizienz und Effektivität der Systemerstellung erwartet [Sa97 3-82; Sz02; Gr98; Tu01]. Für die Entwicklung und Beschreibung von Komponenten auf Implementierungsebene gewinnen bereits Technologien wie Enterprise Java Beans (EJB) von Sun und .NET von Microsoft an Verbreitung. Jedoch gibt es bisher keine etablierten Standards für die Spezifikation von Komponenten auf Ebene des Fachkonzepts. Die dort zu betrachtenden Fachkomponenten sind spezielle Software-Komponenten, welche Dienste aus einer betrieblichen Domäne zur Verfügung stellen [Ac02].

An dieser Stelle setzt der Arbeitskreis „Komponentenorientierte betriebliche Anwendungssysteme“ (WI-KobAs) der Gesellschaft für Informatik mit seinen Bemühungen zur vereinheitlichten Spezifikation von Fachkomponenten an. Im seit Februar 2002 vorliegenden *Memorandum* [Ac02] dieses Arbeitskreises wird aufbauend auf einem festgelegten Grundverständnis ein aus sieben Ebenen aufgebauter Vorschlag zur Spezifikation von Fachkomponenten vorgestellt, der die vollständige, widerspruchsfreie und eindeuti-

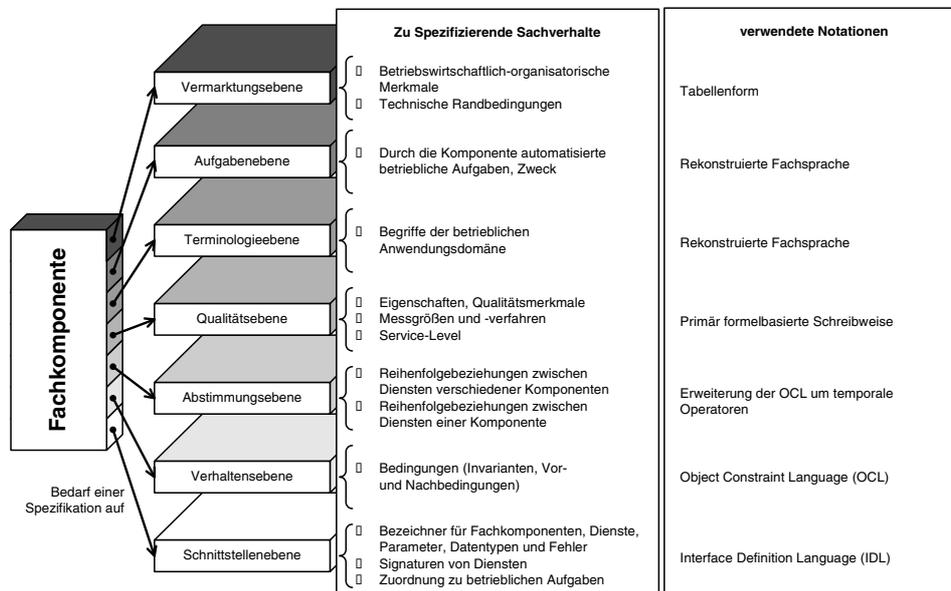


Abbildung 1: Spezifikationsebenen, Aspekte und Notationen [Ac02]

ge Beschreibung der Außensicht von Fachkomponenten anstrebt. Jede Ebene verfolgt einen bestimmten Zweck, der im Memorandum durch Aufzählung von Teilaspekten erläutert wird. Zu den Ebenen werden weiterhin Sprachen angegeben, die zur Beschreibung der jeweiligen Ebene als besonders geeignet erachtet werden. Ferner werden einzelne Zusammenhänge zwischen den Ebenen genannt. Abbildung 1 gibt einen Überblick über die verschiedenen Spezifikationsebenen.

Das Memorandum bedient sich verschiedener Beschreibungstechniken aus unterschiedlichen Quellen. So finden u. a. Tabellen, die Interface Definition Language (IDL; Teil des CORBA-Standards [OMG04]) der Object Management Group (OMG) sowie die Object Constraint Language (OCL; Teil der Unified Modeling Language (UML) der OMG [OMG03]) Verwendung. Anders als beispielsweise die UML beinhaltet das Memorandum jedoch kein formalisiertes, ebenenübergreifendes Metamodell, das die in verschiedene Ebenen zerlegte Spezifikation von Fachkomponenten wieder zusammenführt. Um Inkonsistenzen, Redundanzen bzw. Unvollständigkeiten, die bei separater Betrachtung einzelner Spezifikationsebenen nicht ersichtlich sind, zu vermeiden, schlagen Fettke/Loos vor, ein einheitliches und integriertes Metamodell zu schaffen, das alle Konstrukte der Beschreibungstechniken zusammenführt und ihre Beziehungen expliziert [FL03]. Von einem einheitlichen Metamodell können verschiedene allgemeine Nutzenpotenziale erwartet werden, bspw. die Unterstützung der Überprüfung der Beschreibungskonstrukte und ein verbessertes Verständnis seitens der Nutzer des Memorandums.

Ziel dieser Arbeit ist es, das in [FL03] vorgestellte Metamodell der im Memorandum empfohlenen Beschreibungstechniken weiterzuentwickeln, um dadurch das Memorandum sinnvoll zu ergänzen und die Anwendung und Weiterentwicklung des vorgeschlagenen Spezifikationsrahmens zu unterstützen. Das im Folgenden vorgestellte Metamodell ist eine Rekonstruktion des Memorandums, welche die erste Version des Metamodells verfeinert und präzisiert sowie UML zur Metamodellrepräsentation verwendet. Konzepte zur Komponentenspezifikation, die noch nicht im Memorandum enthalten sind, sollen bei der Rekonstruktion nicht berücksichtigt werden.

Der Beitrag ist wie folgt aufgebaut: Abschnitt 2 stellt den Begriff des Metamodells und die Nutzenpotenziale, die sich durch die Anwendung der Metamodellierung auf die Komponentenspezifikation ergeben, dar. Abschnitt 3 untersucht verwandte Forschungsarbeiten sowie den Stand der Erarbeitung eines Metamodells für das Memorandum. Abschnitt 4 beschreibt das entwickelte Metamodell. In Abschnitt 5 schließt sich eine Diskussion des vorgestellten Metamodells an. Der Beitrag endet mit einem Ausblick in Abschnitt 6.

2 Begriff und Nutzen der Metamodellierung

2.1 Begriff des Metamodells

Vereinfacht kann ein Metamodell als Modell eines Modells aufgefasst werden. Aus Sicht des Metamodells wird das zugrunde liegende Modell Objektmodell genannt. Es lassen sich zwei Prinzipien der Metamodellbildung unterscheiden [St96, S. 17-28]:

1. Bei der sprachbasierten Metaisierung repräsentiert das Metamodell die zur Konstruktion des Objektmodells verwendete Modellierungssprache.
2. Bei der prozessbasierten Metaisierung repräsentiert das Metamodell die zur Konstruktion des Objektmodells durchzuführenden Modellierungsschritte.

Es ist darauf hinzuweisen, dass einige Autoren dem Metamodell-Begriff ausschließlich eine sprachbasierte Betrachtungsweise unterstellen und die als Vorgehensmodell bezeichnete prozessbasierte Sicht davon strikt abgrenzen. In dieser Arbeit werden ausschließlich sprachbasierte Betrachtungen angestellt. Auf eine wiederholte Kennzeichnung des Metamodells als sprachbasiert wird im Folgenden verzichtet.

Nach dem sprachbasierten Ansatz stellt das Metamodell die Modellierungssprache durch die darin verwendeten Konstrukte und deren Beziehungen dar. Diese Elemente der Metaebene bestimmen, welche Instanzen auf der Ebene des Objektmodells erstellt werden können. Zu jedem Element des Objektmodells muss ein entsprechendes Element des Metamodells existieren. Formale Grammatikdefinitionen von textuellen Sprachen (etwa von Programmiersprachen) können als Sonderfall eines sprachbasierten Metamodells angesehen werden.

2.2 Nutzenpotenziale für die Komponentenspezifikation

Ein integriertes Metamodell expliziert die Konstrukte der Beschreibungstechniken und ihre Beziehungen. Damit eröffnet es verschiedene allgemeine Nutzenpotenziale (aufbauend auf [FL03]):

1. Überprüfung der Konstrukte: Konsistenz und Widerspruchsfreiheit des Memorandums können auf Basis des Metamodells überprüft werden. Die Ergebnisse der Überprüfung können Ausgangspunkt für die Weiterentwicklung des Memorandums sein.
2. Besseres Verständnis seitens der Nutzer: Die Nutzer können eine einheitliche, formalisierte Dokumentation aller Konstrukte und deren Beziehungen vorfinden. Personen, für die nur eine einzelne Perspektive relevant ist, können sich einen Überblick über die Konstrukte aller Spezifikationsebenen verschaffen, ohne sich mit den einzelnen Notationen auseinandersetzen zu müssen.
3. Vergleich alternativer Spezifikations- und Beschreibungstechniken: Das Metamodell kann als Ausgangspunkt für den Vergleich alternativer Techniken zur Komponentenspezifikation dienen [DW98; Be99; Fi00; Ha99]. Dabei können bspw. Aussagen über die Vollständigkeit gewonnen werden. Weiterhin können anhand des Metamodells alternative Beschreibungstechniken für einzelne Ebenen untersucht werden.
4. Implementierung von Werkzeugen: Das Metamodell kann als Grundlage zur Implementierung von Werkzeugen dienen, welche die Erzeugung, Verwaltung und Verarbeitung von Spezifikationen zum Gegenstand haben.

3 Forschungsstand

3.1 Überblick

Die vorgestellten Nutzenpotenziale von Metamodellen sind nicht völlig neu, sondern wurden bereits in anderen Forschungsbereichen in ähnlicher Form erkannt und genutzt. Anwendungsbeispiele für den Einsatz von Metamodellen finden sich bei der Evaluierung von objektorientierten Analysemethoden [St96], der Modell Driven Architecture [KWB03], der Auswahl von Workflowmanagementsystemen [Mü99] oder der Darstellung und dem Vergleich von Ontologien [Da02]. Bereits erwähnt wurde die Fundierung der UML auf einem sichtenübergreifenden Metamodell [OMG03].

Das Memorandum selbst beinhaltet kein integriertes Metamodell. Gleichwohl existieren für einzelne Sichten bereits syntaktische Regelwerke, die als Metamodelle verstanden werden können. Turowski stellt in [Tu01] ohne dedizierte Berücksichtigung von Ebenen oder Beschreibungstechniken die Begriffe Fachkomponente, Dienst, Schnittstelle, betriebliche Aufgabe, betriebliches Anwendungssystem, Implementierung und Eigenschaft und ihre Beziehungen mithilfe eines UML-Klassendiagramms dar. Dieses Modell kann als grobes Metamodell aufgefasst werden, das im Hinblick auf die hier verfolgten Zwecke der Präzisierung und Anpassung bedarf. Präzisere Metamodelle bis hin zu

formalen Grammatiken existieren für fast alle Ebenen des Memorandums. Diese Darstellungen sind unabhängig und nicht integriert, bilden aber eine wichtige Grundlage für die Entwicklung eines integrierten Metamodells. Ein erster Ansatz zur Schaffung eines integrierten Metamodells für das Memorandum ist die Arbeit von Fettke/Loos [FL03]. Abbildung 2 fasst alle Grundlagen für die Erarbeitung des weiterentwickelten Metamodells zusammen.

Spezifikationsebene	Quellen
Vermarktungsebene	[FL01]
Aufgabenebene	[Or97]
Terminologieebene	[Or97]
Qualitätsebene	-
Abstimmungsebene	[CT00]
Verhaltensebene	[OMG03]
Schnittstellenebene	[OMG03]
alle Ebenen	[Ac02] [FL03]

Abbildung 2: Grundlagen für die Erstellung eines integrierten Metamodells

3.2 Entwicklungsbedarf

Die vorliegende Arbeit setzt die in [FL03] begonnene Entwicklung eines einheitlichen Metamodells für das Memorandum zur vereinheitlichten Spezifikation von Fachkomponenten fort. Fettke/Loos wählen für die Meta-Modellierung das Entity-Relationship-Modell und begründen ihre Wahl mit der Verbreitung und der Verständlichkeit des Ansatzes. Die Arbeit liefert zwar einen ersten Beitrag, es können aber einige Schwachpunkte identifiziert werden.

1. Im Memorandum können einige Konstrukte und Beziehungen identifiziert werden, die im bisherigen Metamodell nicht berücksichtigt sind (bspw. Versionsstand auf der Vermarktungsebene). Ohne diese Elemente ist das Metamodell unvollständig.
2. Es ist sinnvoll, dass das Metamodell den Aufbau der Spezifikation einer Fachkomponente aus Sicht der einzelnen Ebenen widerspiegelt. Die von Fettke/Loos gewählte ER-Modellierungstechnik verfügt nicht über spezifische Mechanismen, um die Konstrukte des Memorandums allgemein hinsichtlich ihrer jeweiligen Spezifikationsebene zu gruppieren. Ferner können Kompositionsbeziehungen von allgemeinen Assoziationen nicht unterschieden werden.
3. Die Verwendung der klassischen ER-Modellierungstechnik für das Metamodell kann als Bruch zu den im Memorandum verwendeten, primär objektorientierten Beschreibungstechniken (insbesondere denen der OMG: IDL, OCL) verstanden werden.

Vor dem Hintergrund dieser Defizite wird im Folgenden das Metamodell des Memorandums weiterentwickelt. Um die Kritikpunkte 2 und 3 zu adressieren, werden für die

Beschreibung des Metamodells im Rahmen dieser Arbeit die Konstrukte der statischen Struktur aus der Unified Modeling Language (UML) der OMG gewählt. Die UML ist heute weit in Industrie, Forschung und Lehre verbreitet, und die wichtigsten Konstrukte zur Modellierung der statischen Struktur sind allgemein bekannt. Sie erlaubt die Unterscheidung verschiedener Beziehungstypen sowie die Gruppierung von Modellierungselementen bspw. durch das Konstrukt des UML-Paketes. Die Wahl der UML ist im Hinblick auf das Memorandum ein konsequenter Schritt, denn

- mit der OCL empfiehlt das Memorandum einen Bestandteil der UML als primäre Notation zur Spezifikation der Verhaltensebene;
- mit der IDL empfiehlt das Memorandum einen Bestandteil eines weiteren Mitglieds der Familie der OMG-Standards als primäre Notation zur Spezifikation der Schnittstellenebene;
- für die Spezifikation der Terminologieebene empfiehlt das Memorandum ergänzend die Erstellung eines integrierten Attribute- und Dienstschemas als UML-Klassendiagramm.

Um die Anzahl der verwendeten Konstrukte gering und die Darstellung einheitlich zu halten, wird auf die Verwendung von Attributen im Klassendiagramm verzichtet. Zudem stellt die UML Attribute mit Kompositionsbeziehungen zwischen einer Klasse und den Klassen der Attribute gleich [OMG03]. Im Rahmen einer (zukünftigen) Präzisierung des Metamodells ist die Verwendung von Attributen sinnvoll. In der vorliegenden Arbeit stehen jedoch die Zusammenhänge der elementaren Konstrukte des Memorandums im Vordergrund.

Es sei angemerkt, dass die Eignung der UML zur Sprachspezifikation nicht allgemein akzeptiert ist, da sie einerseits nicht die formale Fundierung vergleichbar dem ERM aufweist, andererseits viele der UML-Konstrukte nicht benötigt werden. Dem ist zu entgegen, dass das von Chen [Ch76] ursprünglich eingeführte ERM oft als unzureichend (gelegentlich sogar schon als „veraltet“) empfunden wird, was sich in einer Vielzahl formal weniger fundierter Erweiterungen widerspiegelt. Die Konstrukte der UML sind im Vergleich zum ERM umfassender definiert, wobei die OMG die Spezifikation der UML durch ein wiederum mit dem Klassendiagramm der UML definiertes Metamodell als hinreichend formal auffasst [OMG03]. Nach Meinung der Autoren überwiegt bei einem UML-basierten Metamodell des Memorandums der Nutzen, der sich sowohl durch die Vereinheitlichung der Spezifikationstechniken als auch durch den Gewinn zusätzlicher statischer Beschreibungskonstrukte ergibt. Zwar können die Nachteile der UML im Vergleich zum ERM hinsichtlich Sprachkomplexität und Formalisierungsgrad nicht ausgeräumt werden. Allerdings kann durch die getroffene Wahl auf die Einführung einer weiteren Sprache verzichtet werden.

4 Entwicklung des Metamodells

4.1 Spezifikationsrahmen

In diesem Abschnitt wird das Metamodell des Memorandums entwickelt und mit UML-Klassendiagrammen beschrieben. (Die Diagramme selbst befinden sich zu Gunsten der Übersichtlichkeit im Anhang.) Zunächst wird auf grober Ebene der Ordnungsrahmen, den das Memorandum mit seinen Spezifikationsebenen vorgibt, auf UML-Pakete (Packages) abgebildet (Abbildung 3). In das Paket „Spezifikationsrahmen“ werden dazu gemäß den Spezifikationsebenen die Pakete „Schnittstellen“, „Verhalten“, „Abstimmung“, „Qualität“, „Terminologie“, „Aufgaben“ und „Vermarktung“ eingeordnet. Die Spezifikation einer Fachkomponente setzt sich aus je genau einer Spezifikation für jede Ebene zusammen. Für keine Ebene darf eine Spezifikation fehlen (Anspruch der Vollständigkeit). Je eine Spezifikation pro Ebene genügt zur Beschreibung einer Fachkomponente. Deshalb wird zur Vermeidung von Widersprüchen die Existenz mehrere Spezifikationen einer Komponente verboten. Es sei ausdrücklich darauf hingewiesen, dass der Spezifikationsrahmen ebenso Teil des vorgestellten Metamodells ist.

4.2 Spezifikationsebenen

Zwecks besserer Übersichtlichkeit wird in Abbildung 4 das entwickelte Metamodell in seiner Gesamtsicht dargestellt. Im Folgenden werden die einzelnen Ebenen im Detail erläutert.

Als eine primäre Notation zur Spezifikation der Schnittstellenebene schlägt das Memorandum die OMG IDL vor. Die OMG IDL ist eine textuelle Notation und formal durch eine Grammatik in Anlehnung an die erweiterte Backus-Naur-Form definiert [OMG04 3-12 - 3-18]. Das Memorandum enthält keine expliziten Aussagen, ob alle Konstrukte der IDL für die Schnittstellenspezifikation von Fachkomponenten relevant sind, und wie die Konstrukte der OMG IDL auf Begriffe des Memorandums abzubilden sind. Vom Memorandum explizit vorgegeben wird nur, dass die benötigten Dienste als IDL-Ausdruck unter `interface extern` zusammenzufassen sind. In der Erläuterung des Beispiels zur Schnittstellenspezifikation wird schließlich noch gefordert, den Namen der Fachkomponente in Verbindung mit dem IDL-Schlüsselwort `interface` festzulegen. Im Metamodell wird das IDL-`interface` durch die Klasse Schnittstelle repräsentiert. Die Bestandteile der Schnittstelle und deren Beziehungen lassen sich durch Analyse der IDL-Grammatik ermitteln. Zu den Elementen gehören Deklarationen von Typen, Konstanten, Ausnahmen, Attributen und (IDL-) Operationen. (Zwei weitere in Verbindung mit CORBA 3.0 eingeführte Bestandteile beziehen sich auf die Verwaltung von IDL-Ausdrücken in Repositories und werden hier nicht betrachtet. Vgl. dazu auch Abschnitt 5.1.) Die deklarierten Schnittstellenelemente stehen untereinander in Beziehung.

Gemäß dem Memorandum sind auf Schnittstellenebene zusätzlich Zuordnungen zwischen Fachbegriffen und (Daten-)Typen sowie zwischen Diensten und Aufgaben zu notieren. Der Begriff des Dienstes wird in der IDL-Terminologie nicht gebraucht. Aufgrund des Beispiels zur Schnittstellenspezifikation im Memorandum erscheint die Interpretation der IDL-Operation als Dienst angebracht.

Für die Spezifikation der Verhaltensebene empfiehlt das Memorandum die OCL, um Invarianten zu Diensten und Fachkomponenten sowie Vor- und Nachbedingungen zu Diensten anzugeben. Abweichend von der Formulierung im Memorandum wird im vorgeschlagenen Metamodell die Schnittstelle statt der Fachkomponente verwendet. Die Schnittstelle ist analog zum Dienst ein Element der Schnittstellenebene. Außerdem ist die Angabe von Invarianten auch für die Schnittstelle der genutzten Dienste (*interface extern*) sinnvoll. Für die OCL liegt eine formale Grammatik in erweiterter Backus-Naur-Form vor [OMG03 6-45 - 6-48].

Eine um temporale Operatoren erweiterte OCL soll für die Spezifikation der Abstimmungsebene verwendet werden, um Bedingungen für Dienste anzugeben. Dabei wird auf weitere Dienste Bezug genommen. Bedingungen, die sich nur auf die Dienste der spezifizierten Komponente selbst beziehen, heißen Intraabstimmungsbedingungen. Werden externe Dienste referenziert, spricht man von Interabstimmungsbedingungen.

Die Spezifikation der Qualitätsebene problematisiert das Memorandum relativ ausführlich, verzichtet letztlich aber auf die Angabe einer empfohlenen Notation und eines Beispiels. Daher wurden zur Rekonstruktion die im Memorandum angegebene Primärquelle [ScSc00] hinzugezogen. Im Metamodell wird die Spezifikation dahingehend interpretiert und präzisiert, dass die Spezifikation der Qualität sich zusammensetzt aus Angaben von Eigenschaften, die als Tupel aus Qualitätskriterium, Messverfahren, Referenzumgebung und Maßeinheit aufgefasst werden. Qualitätskriterium, Messverfahren und Maßeinheit sind als Begriff auf Terminologie-Ebene zu klären.

Für die Spezifikation der Terminologieebene wählt das Memorandum Fachnormsprachen als primäre Notation. Ergänzend wird die Angabe eines integrierten Attribute- und Diensteschemas als UML-Klassendiagramm empfohlen. Im hier vorgeschlagenen Metamodell wird die Klärung der Begriffe durch generische Beziehungen abgebildet, d. h. die Begriffe stehen durch Beziehungen, deren Semantik (Typ) vom Anwender des Memorandums bestimmt wird, in Verbindung. Die präzise Metamodellierung des integrierten Attribute- und Diensteschemas wird hier nicht durchgeführt, stattdessen sei auf die entsprechenden Abschnitte des UML-Standards [OMG03 2-12 - 2-73] verwiesen.

Fachnormsprachen kommen auch auf Aufgabenebene zum Einsatz. Hier kann analog zur Terminologieebene eine Metamodellierung durch generische Beziehungen vorgenommen werden. Dabei müssen neben reinen Aufgabenbeziehungen auch Bezüge auf Fachbegriffe berücksichtigt werden.

Bei der Spezifikation der Vermarktungsebene kommt eine Tabelle zum Einsatz. Die zu notierenden Attribute und deren Multiplizitäten werden vom Memorandum fest vorgegeben. Im Metamodell werden diese Angaben aus Gründen der Einheitlichkeit des Metamodells nicht direkt als Attribute der Spezifikation der Vermarktungsebene, sondern durch Klassen, die über Kompositionsbeziehungen mit der Spezifikation verbunden sind, abgebildet.

Die Abhängigkeiten der einzelnen Spezifikationsebenen untereinander sind in Abbildung 12 mit Hilfe eines Package-Dependency-Diagramms angeführt. Dabei zeigt sich, dass die Vermarktungsebene keine Abhängigkeiten zu anderen Ebenen aufweist. Keine weitere Spezifikationsebene ist unabhängig von anderen Ebenen. Klare Abhängigkeiten bestehen auf der Qualitäts-, Abstimmungs- und Verhaltensebene: Jede dieser drei Ebenen besitzt eine Abhängigkeit zur Schnittstellenebene bzw. die Qualitätsebene noch zusätzlich zur Terminologieebene. Die Schnittstellenebene wiederum stützt sich ab auf die Aufgaben- und die Terminologieebene, die Aufgabenebene nur auf die Terminologieebene. Indes stützt sich die Terminologieebene ebenso auf die Schnittstellenebene ab. Mit anderen Worten: Zwischen der Schnittstellen- und der Terminologieebene besteht eine zyklische Abhängigkeit.

5 Diskussion

5.1 Zur Überprüfung des Metamodells

Eine formale Verifikation des Metamodells gegenüber dem Memorandum ist nicht möglich, weil das Memorandum selbst nur zum Teil formale Spezifikation enthält und in weitem Maße Gebrauch natürlichsprachiger Formulierungen aufweist. Vielmehr ist das formalisierte Metamodell ein Beitrag, der die Verifikation von Umsetzungen des Memorandums, bspw. Werkzeuge, unterstützt.

Eine durchführbare Überprüfung des Metamodells stellt die Untersuchung auf Widerspruchsfreiheit dar. Im Metamodell werden Aussagen hinsichtlich der Ober- und Untergrenzen der Multiplizität von Objektbeziehungen gemacht. Dabei sind bestimmte Widersprüche vorstellbar. Es lässt sich jedoch nicht ohne weitere Analyse feststellen, ob diese Fehler der Metamodell-Erstellung oder bereits dem zugrunde liegenden Memorandum anzurechnen sind.

Das hier entwickelte formalisierte Metamodell stellt einerseits eine sorgfältige Interpretation und Rekonstruktion des nicht-formalen Textes des Memorandums dar. Die Überprüfung, ob das Metamodell die hinter dem Memorandum stehende Aussageabsicht adäquat wiedergibt, ist eine Frage einer möglichen Interpretation. Einerseits erscheinen hier Autoren des Memorandums für diese Aufgabe besonders geeignet, da sie evtl. Unklarheiten oder Mehrdeutigkeiten geeignet interpretieren können. Andererseits können gerade die Autoren des Memorandums verschiedene implizite Prämissen und Vorannahmen unterstellen, die eine adäquate Interpretation erschweren.

Bei der Erstellung des vorliegenden Metamodells waren drei Personen beteiligt. Zwei davon sind selbst an der Erstellung des Memorandums beteiligt gewesen. Die unabhängige Diskussion der Metamodellbildung unter den Autoren des Metamodells erlaubt damit ein gewisses Überprüfungs-niveau im folgenden Sinne: Gegenstand der Validierung des Metamodells ist die Überprüfung, ob das Memorandum hinreichend genau wiedergegeben wird. Im Rahmen der Konstruktion des vorliegenden Metamodells wurde eine Validierung durch Inspektion und Plausibilitätsprüfung in engem Personenkreis durchgeführt. Die Autoren räumen ein, dass dieses Vorgehen nur eine minimale Überprüfungsadäquanz sicherstellt. Eine gründlichere Validierung durch einen weiteren Personenkreis aus Erstellern und Anwendern des Memorandums ist wünschenswert.

Ebenso erscheint eine empirische Überprüfung des Metamodells angebracht. Hierbei gilt es zu unterscheiden, ob der empirische Gehalt des Metamodells oder empirische Aussagen hinsichtlich der Nützlichkeit, Relevanz, Klarheit des Metamodells in der Praxis untersucht werden. Eine genauere Untersuchung des empirischen Gehaltes des Metamodells wird vermutlich ergeben, dass dieser sehr gering ist, wodurch bspw. eine Falsifikation des Metamodells erschwert, wenn nicht sogar ausgeschlossen ist. Hypothesen zur Nützlichkeit von Metamodellen im Allgemeinen werden in der Wirtschaftsinformatik meist akzeptiert, obgleich eine nähere Untersuchung notwendig erscheint. Diese ist aber nicht Gegenstand der vorliegenden Untersuchung.

5.2 Probleme bei der Metamodellerstellung

Um ein einheitliches Metamodell für das Memorandum zur vereinheitlichten Spezifikation von Fachkomponenten zu entwickeln, müssen die essenziellen Konstrukte und Zusammenhänge des Memorandums identifiziert werden. Dabei werden im Memorandum verborgene Inkonsistenzen und Unschärfen deutlich. Mit der Erstellung des Metamodells beginnt so bereits die Realisierung des ersten Nutzenpotenzials, nämlich der Überprüfung der Konstrukte des Memorandums.

Vergleicht man die Terminologie, die zur Beschreibung des Zwecks einer Ebene im Memorandum verwendet wird, mit der Terminologie der empfohlenen Notationen, so lassen sich speziell auf Schnittstellenebene deutliche Unterschiede feststellen, die durch das Memorandum nicht überbrückt werden. Eine klare Zuordnung zwischen Begriffen der Zweckbeschreibung und der Notation fehlt. Solche Unterschiede treten in mehreren Formen auf:

- Analoge Konzepte werden unterschiedlich bezeichnet.
Beispiel: Statt Dienst verwendet die OMG IDL die Bezeichnung Operation.
- Elemente der Terminologie der Zweckbeschreibung haben keinen Repräsentant in der Terminologie des Notationsvorschlags.
Beispiel: Es gibt keinen IDL-Ausdruck zur Angabe von Fehlermeldungen.

- Elemente der Notation haben keinen Bezug zu Begriffen der Zweckbeschreibung im Memorandum.

Beispiel: Es gibt keine Zuordnung des gültigen IDL-Konstrukts „Repository Identifier Prefix Declaration“ (Schlüsselwort `typeprefix`) zu einem Konstrukt der Zweckbeschreibung des Schnittstellenebene. Dieses Element darf laut OMG IDL 3.0 in Schnittstellenspezifikationen verwendet werden.

- Begriffsstrukturen sind unscharf oder stehen in Konflikt.

Beispiel: Auf Schnittstellenebene wird u. a. die Benennung der Dienste und das Festlegen der Signaturen bezweckt, ohne anzugeben, welche Elemente der OMG IDL diese Angaben tatsächlich bewerkstelligen. Ein Anwender könnte sich fragen, ob ein Dienst mehrere Signaturen besitzen darf und wie dies zu spezifizieren ist. Dies wird vom Memorandum nicht explizit beantwortet.

Eine besondere Problematik stellt dar, dass abseits des Memorandums im Umfeld der OMG IDL und der OCL mittlerweile ebenfalls von Komponenten gesprochen wird. So erlaubt die OMG IDL seit der wenige Monate nach dem Memorandum erschienenen Version 3.0 die Deklaration von Komponenten. Solche Komponenten können mehrere Schnittstellen zur Verfügung stellen. Ähnliches gilt für die Komponenten der UML, deren Bestandteil die im Memorandum verwendete OCL ist. Der Anwender des Memorandums könnte sich fragen, in welchem Verhältnis diese Komponentenbegriffe und -spezifikationsmöglichkeiten zum Memorandum stehen.

5.3 Verbesserungspotenziale

Eine präzise Festlegung der Terminologie insbesondere auf Schnittstellenebene ist erforderlich, um die bisher bestehende Unschärfen und Inkonsistenzen auszuräumen. Es liegt nahe, dafür das Mittel einzusetzen, das auch für die Spezifikation der Terminologie von Fachkomponenten vorgesehen ist, also eine Fachnormsprache. Ergänzend zur fachnormsprachlichen Definition der Begriffe empfiehlt das Memorandum die Angabe eines integrierten Attribute- und Dienstschemas. Übertragen auf das Memorandum selbst ist die Angabe eines einheitlichen Metamodells als Bestandteil der Beschreibung des Spezifikationsrahmens hilfreich. Die Definition der Terminologie des Memorandums wäre somit gleichzeitig beispielgebend für die Definition der Terminologie von Fachkomponenten.

Die für einzelne Ebenen empfohlenen Notationen OMG IDL und OCL sind in einem anderen Kontext definiert, der nicht ohne weiteres auf das Memorandum übertragen werden kann. Deshalb muss das Memorandum klären, wie in der Notation fehlende Elemente des Memorandums zu behandeln sind, und welche Notationselemente nicht benötigt werden. Ausgehend von den festgelegten Begriffen ist eine präzisere Festlegung der Zuordnung der Konzepte des Memorandums zu den Elementen der Notationen erforderlich. Wo eine unmittelbare Zuordnung gegeben ist, kann diese – in Analogie zur Zuordnung von Datentypen und Fachbegriffen im Memorandum – in Tabellenform notiert werden.

Bei der Gestaltung von Software-Architekturen gilt die Vermeidung von zyklischen Abhängigkeiten zwischen einzelnen Architekturkomponenten als wesentlicher Erfolgsfaktor. Dieses Gestaltungsprinzip wird im hier vorgestellten Metamodell weitgehend beachtet, wobei eine Verletzung festgestellt werden kann: Zwischen der Terminologie- und Schnittstellenebenen bestehen gegenseitige Abhängigkeiten. Einerseits greift die Schnittstellenebene auf die Terminologieebene zurück. Andererseits erfordert die Definition eines integrierten Attribute- und Dienstschemas auf der Terminologieebene die Kenntnisse der Dienste auf der Schnittstellenebene. Zwar wird dieses Schema auf der Terminologieebene als optional angegeben, aber ausdrücklich im Memorandum empfohlen. Hier ist zu fragen, ob Vorteile des Dienstschemas die sich ergebenden Nachteile der zyklischen Abhängigkeit zwischen beiden Spezifikationsebenen aufwiegen oder ob das Dienstschema in der Schnittstellenebene zu spezifizieren ist.

6 Zusammenfassung und Ausblick

In dieser Arbeit wurde das in [FL03] vorgeschlagene Metamodell des Memorandums zur vereinheitlichten Spezifikation von Fachkomponenten neu formuliert und weiterentwickelt. Das präsentierte Metamodell ist mit den Mitteln der UML notiert. Die bei der Entwicklung des Metamodells getätigten Konstruktionsschritte wurden expliziert. Mit der Erstellung des Metamodells ging eine erste Überprüfung der Konstrukte des Memorandums einher. Bei der Überprüfung wurden Probleme bei der Verbindung der Terminologien, der verbalen Zweckbeschreibung und der Notationen auf Schnittstellenebene festgestellt. Zur Verbesserung der Präzision und Konsistenz des Memorandums wird die Festlegung der Terminologie und der Zuordnung zu Notationselementen in Analogie zu Beschreibungsmitteln des Memorandums vorgeschlagen.

Das vorliegende Metamodell hat im Rahmen einer weiteren Überprüfung das Potenzial, Gestaltungsempfehlungen für die Evolution des Memorandums zu liefern. Ein interessanter Ansatz wäre beispielsweise, die Untersuchung von Strukturanalogien [Be95] zwischen Verhaltens- und Abstimmungsebene sowie zwischen Terminologie- und Aufgabenebene vorzunehmen. Ferner kann das Metamodell verfeinert und fester Bestandteil einer künftigen Memorandums-Version werden.

Literaturverzeichnis

- [Ac02] Ackermann, J.; Brinkop, F.; Conrad, S.; Fettke, P.; Frick, A.; Glistau, E.; Jaekel, H.; Kotlar, O.; Loos, P.; Mrech, H.; Raape, U.; Ortner, E.; Overhage, S.; Sahm, S.; Schmientendorf, A.; Teschke, T.; Turowski, K.: Vereinheitlichte Spezifikation von Fachkomponenten - Memorandum des Arbeitskreises 5.10.3 Komponentenorientierte betriebliche Anwendungssysteme. <http://wi2.wiwi.uni-augsburg.de/gi-memorandum.php.htm>, Abruf 2003-03-01.

- [Be95] Becker, J.: Strukturanalogien in Informationsmodellen - Ihre Definition, ihr Nutzen und ihr Einfluß auf die Bildung der Grundsätze ordnungsmäßiger Modellierung (GoM). In: W. König (Hrsg.): Wirtschaftsinformatik '95 - Wettbewerbsfähigkeit, Innovation, Wirtschaftlichkeit. Heidelberg 1995, S. 133-150.
- [Be99] Beugnard, A.; Jézéquel, J.-M.; Plouzeau, N.; Watkins, D.: Making Components Contract Aware. In: IEEE Computer 32 (1999) 7, S. 38-45.
- [Ch76] Chen, P. P.-S.: The Entity-Relationship Model - Toward a Unified View of Data. In: ACM Transactions on Database Systems 1 (1976) 1, S. 9-36.
- [CT00] Conrad, S.; Turowski, K.: Vereinheitlichung der Spezifikation von Fachkomponenten auf der Basis eines Notationsstandards. In: J. Ebert; U. Frank (Hrsg.): Modelle und Modellierungssprachen in Informatik und Wirtschaftsinformatik - Beiträge des Workshops "Modellierung 2000", St. Goar, 5.-7. April 2000. Koblenz 2000, S. 179-194.
- [Da02] Davies, I.; Green, P.; Rosemann, M.: Facilitating an Ontological Foundation of Information Systems with Meta Models. In: A. Wenn; M. McGrath; F. Burstein (Hrsg.): Proceedings of the 13th Australasian Conference on Information Systems (ACIS 2002), 3-6 December. Melbourne 2002, S. 937-947.
- [DW98] D'Souza, D. F.; Wills, A. C.: Objects, Components, and Frameworks with UML - The Catalysis Approach. Reading, MA, et al. 1998.
- [Fi00] Fischer, B.: Specification-Based Browsing of Software Component Libraries. In: Journal of Automated Software Engineering 7 (2000) 2, S. 179-200.
- [FL01] Fettke, P.; Loos, P.: Ein Vorschlag zur Spezifikation von Fachkomponenten auf der Administrations-Ebene. In: K. Turowski (Hrsg.): Modellierung und Spezifikation von Fachkomponenten: 2. Workshop im Rahmen der vertIS (verteilte Informationssysteme auf der Grundlage von Objekten, Komponenten und Agenten) 2001, Bamberg, Deutschland, 05. Oktober 2001. Bamberg 2001, S. 95-104.
- [FL03] Fettke, P.; Loos, P.: Entwicklung eines Metamodells für die "Vereinheitlichte Spezifikation von Fachkomponenten". In: K. Turowski (Hrsg.): Modellierung und Spezifikation von Fachkomponenten: 4. Workshop im Rahmen der Modellierung betrieblicher Informationssysteme (MobIS), Bamberg, Deutschland, 10. Oktober 2003, Tagungsband. 2003, S. 13-21.
- [Gr98] Griffel, F.: Componentware - Konzepte und Techniken eines Softwareparadigmas. Heidelberg 1998.
- [Ha99] Han, J.: An Approach to Software Component Specification. Proceedings of 1999 International Workshop on Component Based Software Engineering. Los Angeles, USA 1999
- [KWB03] Kleppe, A.; Warmer, J.; Bast, W.: MDA Explained: The Model Driven Architecture: Practice and Promise. Boston et al. 2003.

- [Mü99] Mühlen zur, M.: Evaluation of Workflow Management Systems Using Meta Models. Proceedings of the 32th Hawaii International Conference on Systems Science (HICSS '99). Hawaii 1999
- [OMG03]OMG: OMG Unified Modeling Language Specification, Version 1.5. 2003.
- [OMG04]OMG: Common Object Request Broker Architecture: Core Specification, Version 3.0.3. 2004.
- [Or97] Ortner, E.: Methodenneutraler Fachentwurf - Zu den Grundlagen einer anwendungsorientierten Informatik. Stuttgart, Leipzig 1997.
- [Sa97] Sametinger, J.: Software Engineering with Reusable Components. Berlin et al. 1997.
- [ScSc00] Schmietendorf, A.; Scholz, A.: Spezifikation der Performance-Eigenschaften von Softwarekomponenten. In: K. Turowski (Hrsg.): Modellierung und Spezifikation von Fachkomponenten: Workshop im Rahmen der MobIS 2000 Modellierung betrieblicher Informationssysteme, Siegen, Deutschland, 12. Oktober 2000, Tagungsband. Siegen 2000, S. 41-50.
- [St96] Strahinger, S.: Metamodellierung als Instrument des Methodenvergleichs - Eine Evaluierung am Beispiel objektorientierter Analysemethoden. Aachen 1996.
- [Sz02] Szyperski, C.: Component Software - Beyond Object-Oriented Programming. 2. Aufl., London et al. 2002.
- [Tu01] Turowski, K.: Fachkomponenten - Komponentenbasierte betriebliche Anwendungssysteme. Habil.-Schr., Magdeburg 2001.

Anhang

A Spezifikationsrahmen

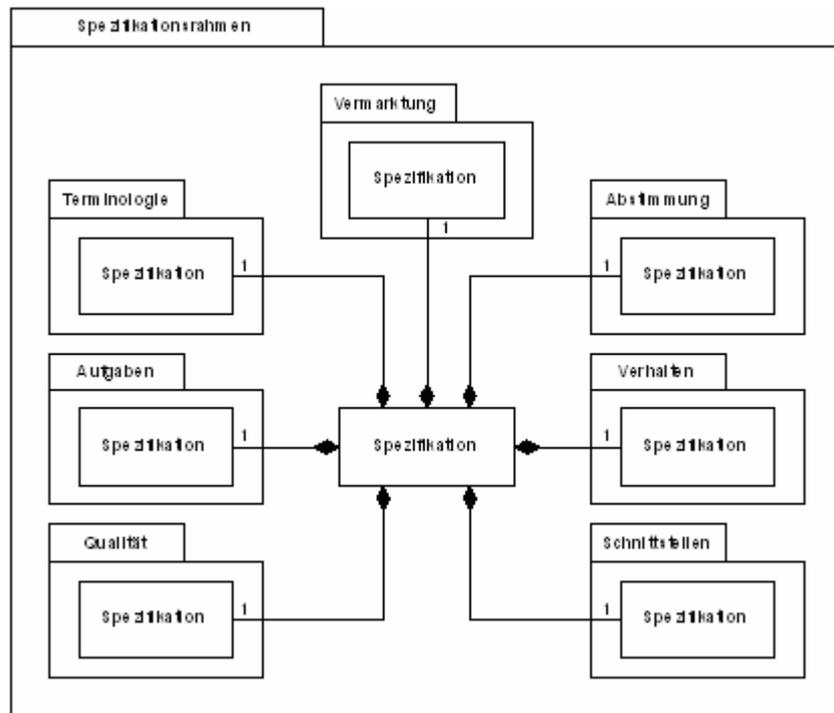


Abbildung 3: Strukturierung des Spezifikationsrahmens und der Spezifikation

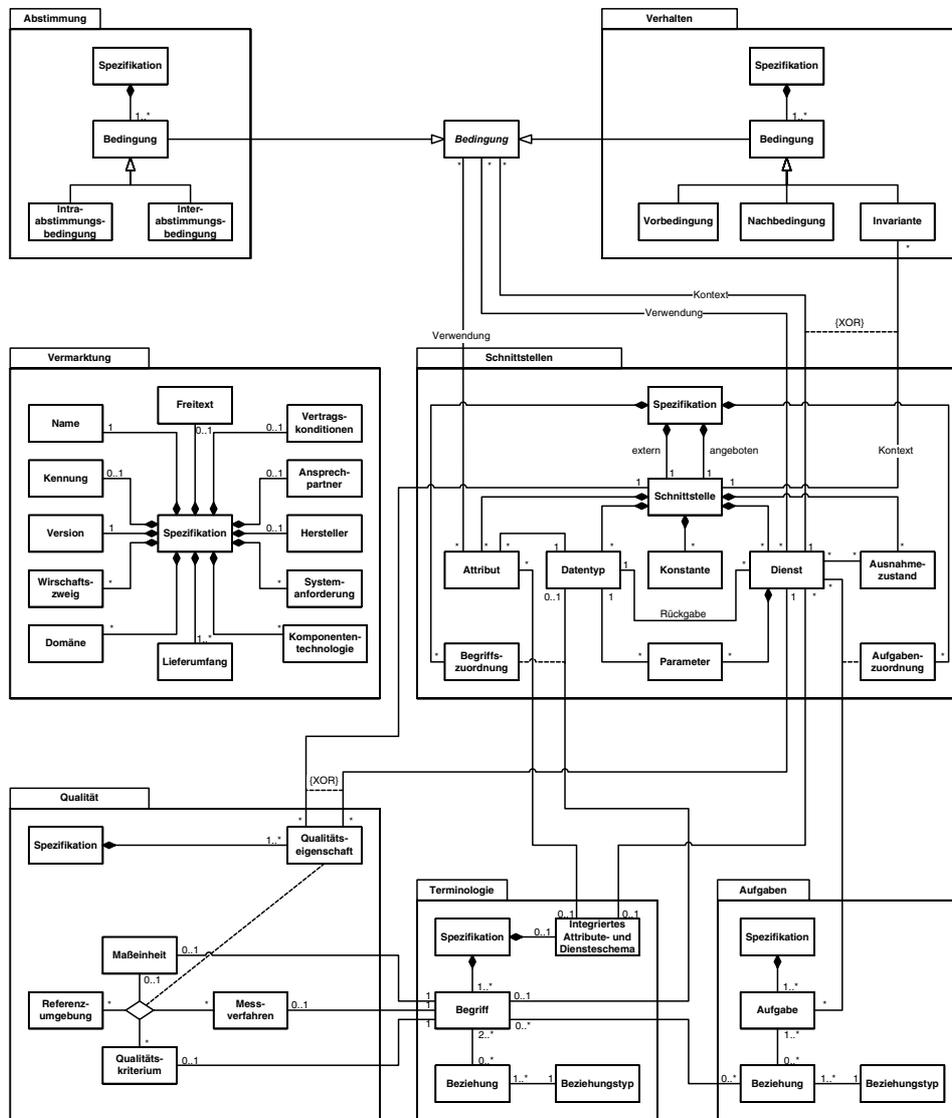


Abbildung 4: Gesamtansicht

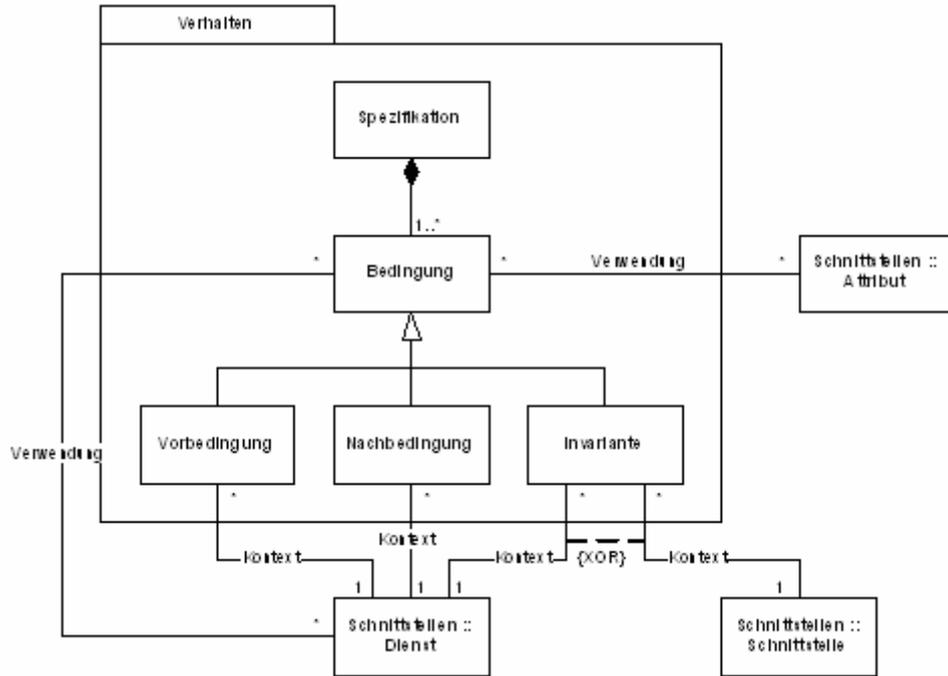


Abbildung 6: Verhaltensebene

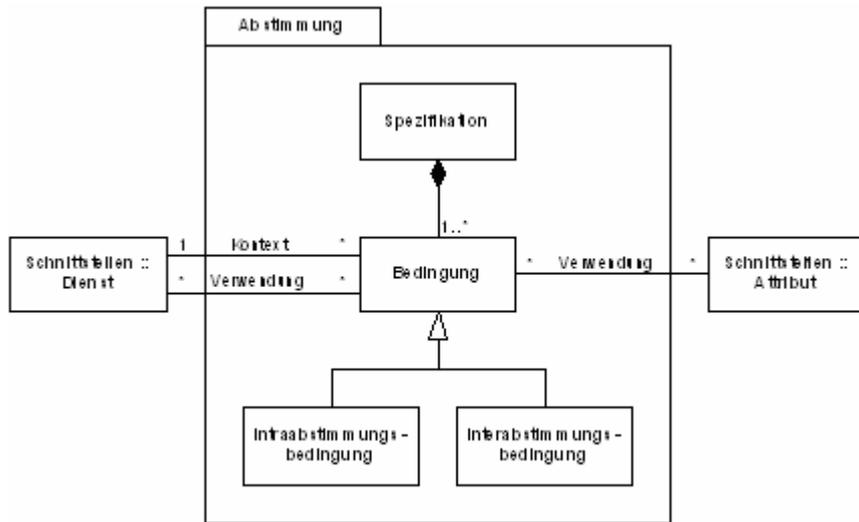


Abbildung 7: Abstimmungsebene

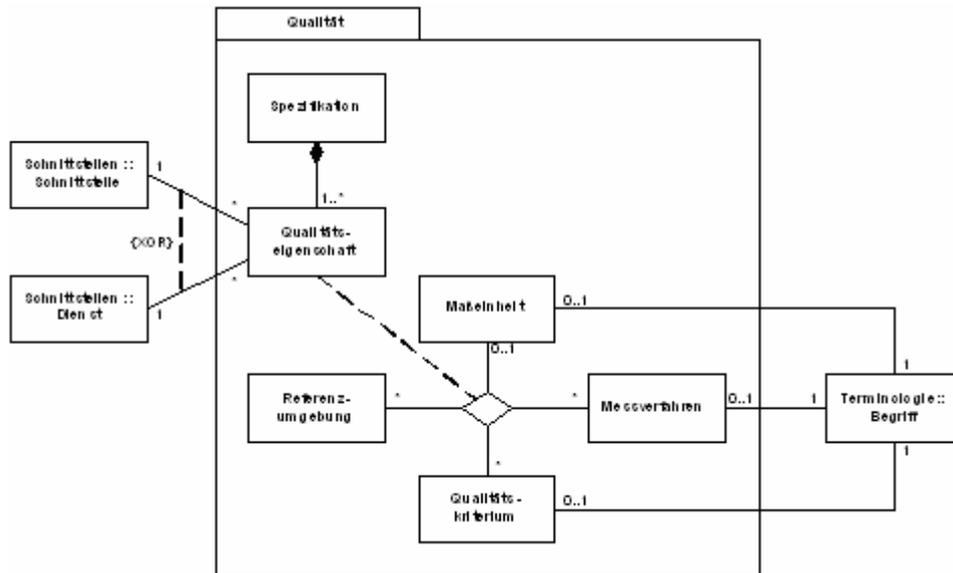


Abbildung 8: Qualitätsebene

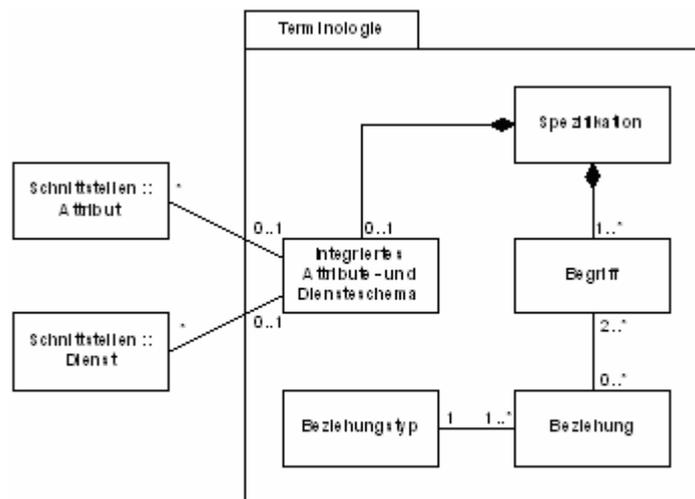


Abbildung 9: Terminologieebene

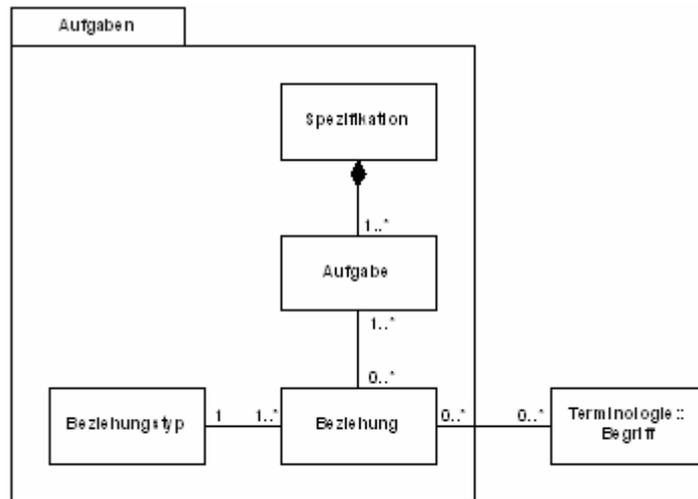


Abbildung 10: Aufgabenebene

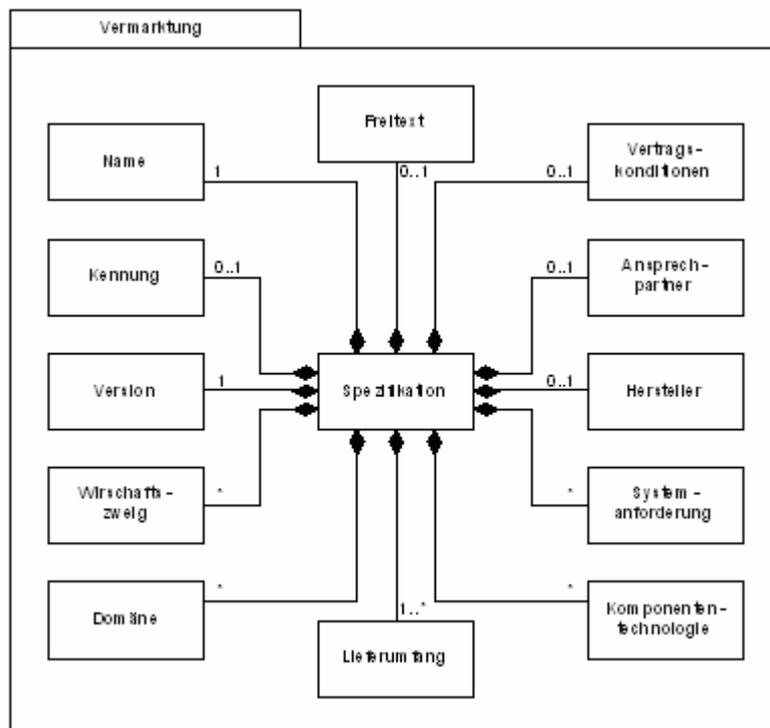


Abbildung 11: Vermarktungsebene

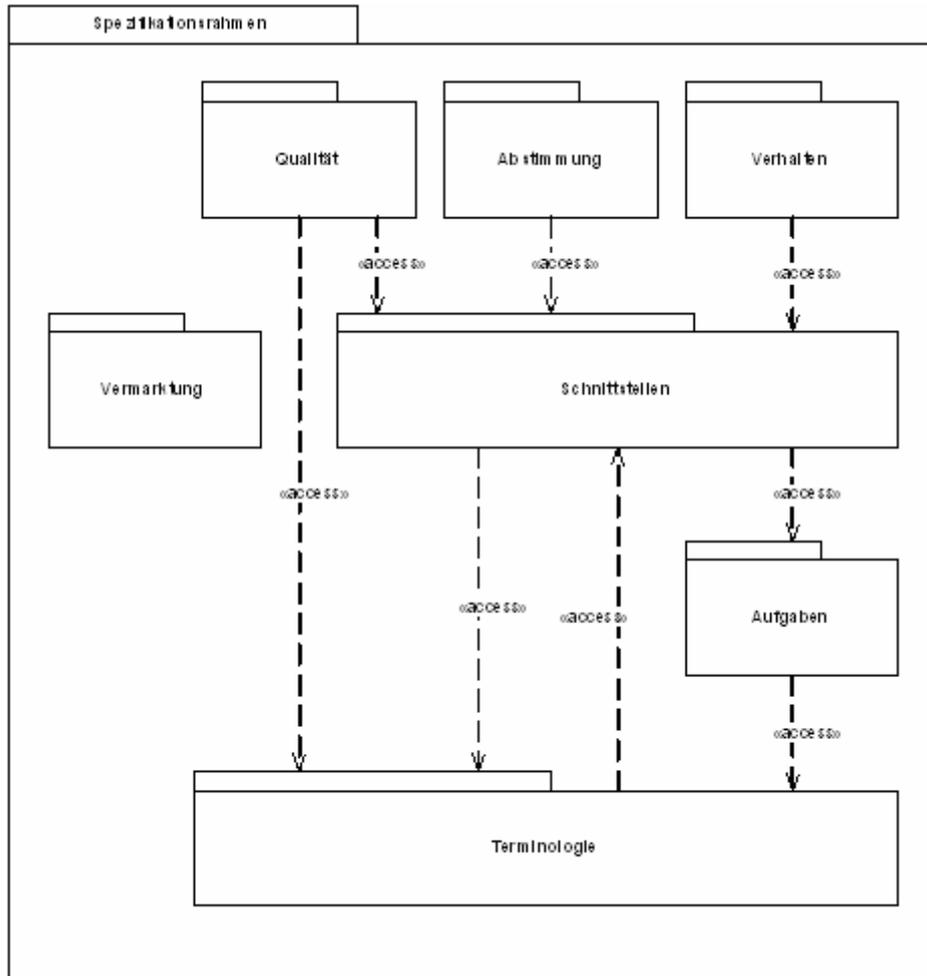


Abbildung 12: Abhängigkeiten der Spezifikationsebenen