

Die Modellierung von Produkten für Handelsplattformen im Internet - ein Ansatz auf der Basis von Metakzepten

Ulrich Frank

Institut für Wirtschaftsinformatik, Universität Koblenz-Landau

Rheinau 1, 56075 Koblenz

Zusammenfassung

Der elektronische Handel erfordert neben einer angemessenen Reorganisation von Geschäftsprozessen die Einführung leistungsfähiger Informationssysteme. Ähnlich wie es für die zugrundeliegenden betriebswirtschaftlichen Bezugsrahmen gilt, kann dabei teilweise auf bewährte Konzepte zurückgegriffen werden. Es gibt jedoch auch eine Reihe spezifischer Anforderungen, die beim Entwurf dedizierter Systeme zu beachten sind. Der folgende Beitrag ist auf die objektorientierte Modellierung von Produkten gerichtet, die auf Handelsplattformen im Internet angeboten werden. Dabei ist nicht zuletzt zu berücksichtigen, dass die in solchen Systemen zu repräsentierenden Produktarten zum Zeitpunkt der Systemerstellung nicht vollständig bekannt sind und gleichzeitig während des Systembetriebs u.U. fortlaufend neue Produktarten zu erfassen sind. Außerdem sollte die Abbildung von Produktvarianten und die Erstellung individueller Produktkonfigurationen möglich sein. Vor dem Hintergrund einer Reihe zusätzlicher Anforderungen, die an solche Systeme zu richten sind, werden zunächst zwei naheliegende Modellierungsansätze beschrieben. Anschließend wird ein leistungsfähigerer Ansatz dargestellt, der in Teilen auf der Verwendung von Metakzepten beruht. Dieser Ansatz ist im Rahmen des Entwurfs eines Referenzmodells für Handelsplattformen im Internet entstanden.

1. Motivation

Die Anbahnung und Abwicklung von Geschäften über das Internet oder kurz "E-Commerce" erfordert neben geeigneten Marketing-Konzepten die Reorganisation von Geschäftsprozessen und ggfs. branchenübergreifende strategische Allianzen. Daneben kommt leistungsfähigen Informations- und Kommunikationssystemen eine herausragende Bedeutung zu. Zur Bewältigung der damit verbundenen Aufgaben kann auf eine große Bandbreite bewährter Konzepte und Methoden aus der Betriebswirtschaftslehre oder der (Wirtschafts-) Informatik zurückgegriffen werden. Gleichzeitig weist E-Commerce allerdings eine Reihe von Besonderheiten auf, die reizvolle Forschungsaufgaben mit sich bringen. Im Hinblick auf die Wirtschaftsinformatik, aus deren Blickwinkel der folgende Beitrag entstanden ist, sind die folgenden Aspekte von besonderer Bedeutung:

Zunehmende Automatisierung von Geschäftsprozessen: Auch wenn E-Commerce vordergründig auf die Anbahnung von Geschäften über das Internet zu fokussieren scheint (und manche gegenwärtige Angebote sich tatsächlich auch darauf beschränken), liegt es aus wirtschaftlichen Gründen nahe, auch vor- und nachgelagerte Prozesse wie Beschaffung, Logistik und Zahlungsverkehr zu integrieren und weitgehend zu automatisieren. Die technische Unterstützung solcher Prozesse erfordert Systeme, die eine hohe Zuverlässigkeit und Integrität gewährleisten. Dazu sind u.a. Konzepte zu entwerfen, die eine enge Integration der Systeme verschiedener Akteure erlauben.

Neuartige Formen der Preisbildung: Im traditionellen Handel werden Preise i.d.R. durch den Anbieter festgelegt und allenfalls durch Verhandlungen mit entsprechend autorisierten Mitarbeitern individuell angepaßt. Der Handel im Internet bietet eine Reihe von Preisbildungsmechanismen (diverse Auktionsformen, Zeitrabatte, Nachfragebündelung etc.), die zwar seit langem bekannt sind, aber dadurch eine neue Qualität erhalten, dass sie - aus der Sicht des Anbieters - eine automatisierte Preisbildung erlauben. Die Herausforderung für die Wirtschaftsinformatik besteht hier u.a. darin, Abstraktionen zu finden, die Handelsplattformen im Internet hinsichtlich der Wahl des Preisbildungsmechanismus ein hohes Maß an Flexibilität und Integrität bieten.

Kontingenz der angebotenen Produkte: Die Vielfalt der in Handelsplattformen im Internet angebotenen

Produkte ist mitunter erheblich, in einige Fällen - wie z.B. bei Auktionshäusern - nahezu unbegrenzt. Im Unterschied zu Kaufhäusern oder Versandhändlern, die häufig auch eine breite Produktpalette anbieten, muß das Sortiment aber nicht mittelfristig geplant und im Zeitverlauf relativ stabil sein. Statt dessen sind täglich vielfältige, kaum vorhersehbare Änderungen in der Produktpalette möglich: heute Ledersessel, morgen Hubschrauber. Da all diese Produkte in Informationssystemen verwaltet werden und zahlreiche Operationen auf ihnen auszuführen sind, müssen sie im Hinblick auf entsprechende Anwendungsfälle *sinnvoll* beschrieben werden. Es geht also um das scheinbare Paradoxon, Gegenstände gehaltvoll zu beschreiben, die man noch gar nicht kennt. Der letzte Aspekt markiert eine Herausforderung für die konzeptionelle Modellierung. Er ist Gegenstand dieses Beitrags.

2. Anforderungen an die Modellierung von Produkten

Im Fachhandel stehen dem Kunden qualifizierte Verkäufer zur Verfügung, um die in Frage kommenden Produkte zu finden und ggfs. zu erläutern. Die Funktion des Verkäufers ist im Internet-Handel durch geeignete Produktklassifikationen und -beschreibungen zu ersetzen. In der einschlägigen Literatur werden denn auch seit geraumer Zeit entsprechende Kataloge, mitunter "Internet Electronic Product Catalogs" (IEPC) genannt, als wesentliche Komponente von Handelsplattformen diskutiert. Die mit solchen Katalogen verbundenen Anforderungen fokussieren aus verständlichen Gründen vor allem auf die an den Kunden gerichtete Präsentation der Produkte. So sind nach Timm und Rosewitz [TiRo98] IEPCs darauf gerichtet, den Kunden "multimedia product representations" zur Verfügung zu stellen. Ein ähnliches Verständnis von Produktkatalogen findet sich bei Keller und Genesereth [KeGe97].

Im Hinblick auf die Realisierung leistungsfähiger Informationssysteme ist die Beschränkung auf die Präsentation von Produktbeschreibungen nicht hinreichend. Wir werden deshalb im folgenden nicht den Begriff Produktkatalog verwenden. Statt dessen sprechen wir von *konzeptionellen Produktmodellen*. Bei der Betrachtung der Anforderungen an Systeme ist es i.d.R. ratsam zwischen verschiedenen Nutzergruppen, also im vorliegenden Fall zwischen den Kunden und den Betreibern einer Handelsplattform, zu unterscheiden. Wir sehen allerdings im folgenden von einer solchen Differenzierung ab, weil sich die Anforderungen beider Gruppen teilweise überschneiden und zudem die Rollentrennung nicht immer klar durchzuführen ist: Ergänzend zur traditionellen Kundenrolle erlauben manche Handelsplattformen ihren Kunden, selbst Produkte einzustellen bzw. als Anbieter zu agieren. Die charakteristischen Anforderungen für die hier betrachtete Systemgattung sind bereits genannt worden:

- 1) Es sollten prinzipiell beliebige Produkte (genauer: Produktarten oder -klassen) abbildbar sein.
- 2) Die Erfassung neuer Produkttypen sollte dabei keine Änderung des Programmcodes oder des Datenbankschemas erforderlich machen, weil dies angesichts der i.d.R. durchgängigen Verfügbarkeit der Systeme und des häufigen Auftretens neuer Produkttypen nicht akzeptabel wäre.

Daneben gibt es Anforderungen, die weitgehend denen entsprechen, auf die die erwähnten Produktkataloge zielen:

- 3) Die auf der Basis konzeptioneller Produktmodelle verwalteten Objekte sollen den Kunden bei der Suche nach geeigneten Produkten unterstützen.
- 4) Dabei sollte die Beschreibung der Produkte aus der Sicht der Kunden gehaltvoll sein und ihren u.U. variierenden Bedürfnissen nach Detaillierung und Anschaulichkeit gerecht werden.

Daneben gibt es eine Reihe von Anforderungen, die mit der automatisierten Verwaltung und Pflege von Produktdaten zusammenhängen:

- 5) Die Produktbeschreibungen sollten es erlauben, im Zeitverlauf betriebswirtschaftlich relevantes Wissen über das Kaufverhalten der Kunden zu sammeln.
- 6) Das System sollte die Erfassung unsinniger Produktbeschreibungen weitgehend ausschließen.

Gleiches gilt für die Erfassung von Bestellungen der Kunden.

- 7) Es sollte möglich sein, Produktvarianten als solche darzustellen, weil auf diese Weise nicht nur die bereits genannten Analysen des Kaufverhaltens unterstützt werden, sondern auch Redundanz bei der Datenerfassung und -verwaltung vermieden wird.
- 8) Der Kunde sollte in die Lage versetzt werden, individuelle (und gleichzeitig korrekte) Konfigurationen zu spezifizieren.
- 9) Verschiedene Formen der Preisbildung und Preiszuordnung (zu Produktarten und einzelnen Produkten) sollten möglich sein.
- 10) Die Produktbeschreibungen sollten hinreichend differenziert sein, um diverse Kommunikationsprotokolle (z.B. zu Lieferanten, Banken und Logistikpartnern) befriedigen zu können.

Grundsätzlich gibt es eine Vielzahl denkbarer Abstraktionen, um diesen Anforderungen zu begegnen. Im folgenden werden wir drei prototypische Ansätze betrachten, die jeweils in unterschiedlicher Weise konkretisiert werden können.

3. Gängige Ansätze zur Modellierung von Produkten

Die Modellierung von Produkten ist ein zentraler Bestandteil vieler betriebswirtschaftlicher Anwendungen, etwa von Warenwirtschaftssystemen, Produktionsplanungs- und Steuerungssystemen oder auch von Kostenrechnungssystemen. Der Fokus unserer Betrachtung beschränkt sich auf solche Ansätze, die für den Handel mit einer vielfältigen Produktpalette anzutreffen sind.

3.1 "Flache" Produktkonzepte

Um möglichst viele Produktarten mit nur einer Produktklasse abbilden zu können, ist es erforderlich, in der Klassenbeschreibung von den Merkmalen, in denen sich die abzudeckenden Produktarten unterscheiden, zu abstrahieren. Anders gewendet: Man beschränkt sich auf solche Merkmalstypen, die auf alle betrachteten Produktarten zutreffen. Die für eine aus der Sicht des Betrachters nötige Differenzierung unterschiedlicher Produktarten erfolgt dann allein durch die Ausprägung der Merkmalstypen. Die Klassenspezifikation in Abb. 1 ist ein Beispiel für ein solches Produktkonzept. Die verwendete Notation ist an UML angelehnt.

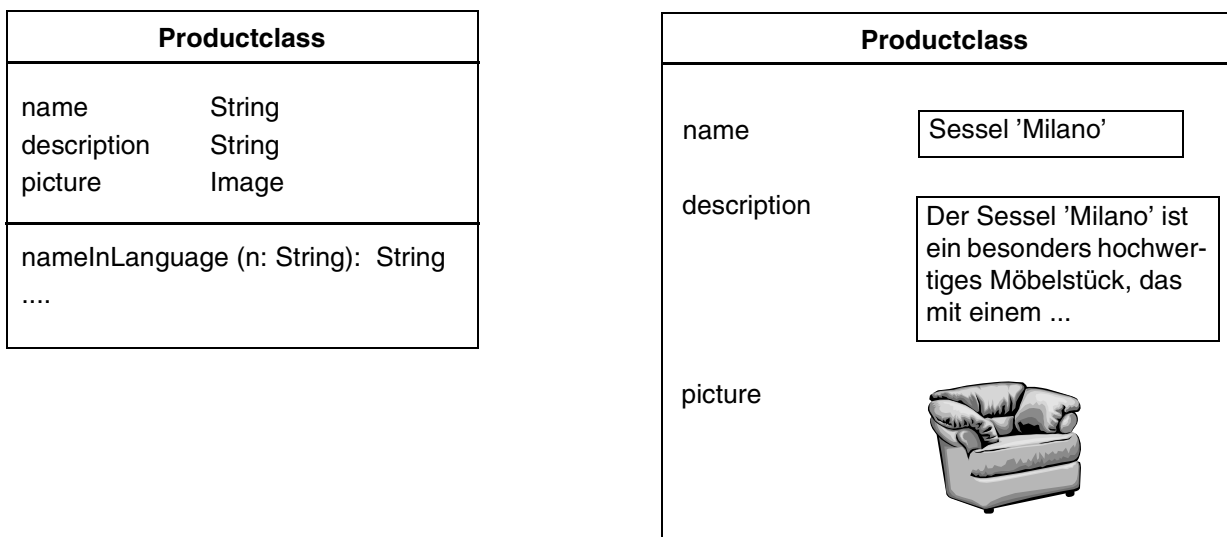


Abb. 1: Generische Produktklasse und Beispielinstantz

Wie die Bezeichnung dieser generischen Klasse andeutet, handelt es sich bei den Instanzen nicht um die Repräsentationen einzelner realweltlicher Objekte, sondern vielmehr um Produktklassen. Das Beispiel verdeutlicht diesen Umstand, der für die Verwendung eines solchen Modells insofern von Bedeutung ist

als es mit der Einschränkung verbunden ist, konkrete Produktinstanzen, wie z.B. eine ganz bestimmte Waschmaschine, nicht abbilden zu können. Häufig ist eine solche Abbildung nicht zweckmäßig, weil der Aufwand, der mit der notwendigen Identifikation der realweltlichen Objekte verbunden ist, zumeist viel zu groß ist - man denke beispielsweise an die Flaschen im Lager eines Getränkegroßhandels. Für Handelsplattformen im Internet wird i.d.R. die Repräsentation von Produktklassen genügen. Wenn die Eigenschaften konkreter Produktinstanzen eine Rolle spielen, etwa bei Gebrauchtfahrzeugen, muß die Repräsentation von Produktklassen um Klassen ergänzt werden, die die Abbildung zugeordneter Instanzen erlauben (eine ausführliche Darstellung verschiedener Alternativen findet sich in [Fran99]).

Offensichtlich ist Anforderung 1 insofern erfüllt, als für alle Produkte eine Beschreibung denkbar ist, die diesem Konzept genügt. Anforderung 2 ist uneingeschränkt erfüllt, da das Anlegen einer neuen Produktklasse lediglich die Instanzierung eines Objekts erfordert. Auch die Anforderungen 3 und 4 können durch diesen Ansatz erfüllt werden. Es bleibt allerdings einschränkend festzuhalten, dass sowohl die Suche nach Merkmalen als auch die Information über Produkteigenschaften nicht durch entsprechende, in der Klassenbeschreibung explizit gemachte Konzepte unterstützt wird, sondern lediglich durch die Ausprägung von Zeichenketten bzw. ergänzenden Abbildungen. Das Sammeln von Wissen über Kaufentscheidungsprozesse ist sicher durch ein semantisch flaches Produktkonzept nicht ausgeschlossen, da entsprechende Daten an anderer Stelle abgelegt werden können. Gleichzeitig ist allerdings zu berücksichtigen, dass ein solches Produktkonzept die differenzierte Analyse des Kundenverhaltens auch nicht eben unterstützt, da die angebotenen Produktklassen konzeptionell nicht unterschieden werden. Anforderung 6 markiert eine nachhaltige Schwäche dieses Konzepts, da konzeptionell nahezu beliebig unsinnige Produktbeschreibungen möglich sind. Die Berücksichtigung von Varianten ist zwar durch eine weitere assoziierte Klasse denkbar, wäre aber wenig hilfreich, da nicht dargestellt werden könnte, in welchen Merkmalen sich Varianten voneinander unterscheiden (Anforderung 7). Aus ähnlichen Gründen ist es nicht möglich, Produktkonfigurationen abzubilden (Anforderung 8).

Im Hinblick auf den Handel mit Produkten mag auffallen, dass so wichtige Eigenschaften wie Einkaufs- und Verkaufspreise oder auch Lagerbestände nicht in der Klassenbeschreibung enthalten sind. Da die Lagerverwaltung u.U. vollständig ausserhalb des Unternehmens liegt, nämlich bei einem Logistikpartner, sollte die Verwaltung von Beständen auch dort erfolgen. Es wäre allenfalls zu überlegen, ob auf diesen extern verwalteten Lagerbestand über Operationen zugegriffen wird, die von der Klasse angeboten werden. Ähnliches gilt für Preise. Auch wenn Preisbildungsmechanismen eingesetzt werden, die zu unterschiedlichen Verkaufspreisen der Produkte einer Klasse im Zeitverlauf führen, müssen irgendwo die Einkaufs- bzw. Mindestverkaufspreise abgelegt werden. Das kann durchaus mit Hilfe von Attributen der Klasse geschehen. Es ist aber auch denkbar, auf (Rahmen-) Verträge mit Anbietern bzw. generelle Spezifikationen von Preisbildungsmechanismen zu verweisen, in denen die Konditionen spezifiziert sind. Ähnliche Überlegungen gelten für die zwei noch vorzustellenden Ansätze. Preise und Lagerbestände stellen deshalb kein wesentliches Differenzierungskriterium für unseren Vergleich dar. Insofern ist die in Abb. 1 dargestellte Klassenbeschreibung indifferent gegenüber der Berücksichtigung verschiedener Preisbildungsmechanismen (Anforderung 9). Die Kommunikation mit externen Partnern erfordert aufgabenspezifische Angaben über Produkte, wie z.B. Abmessungen und Gewichte oder auch die jeweilige Gefahrenklasse. Solange diese Angaben einheitlich sind, also in gleicher Form für alle Produkte gelten, könnten sie mittels geeigneter Attribute abgebildet werden. Wenn aber gewisse Angaben für einzelne Produktklassen spezifisch sind, ist der Ansatz nicht mehr überzeugend, da er in diesem Fall zu konzeptioneller Redundanz und damit zur Verwirrung derjenigen führen würde, die für die Eingabe von Produktdaten zuständig sind.

Zusammenfassend - und sicher wenig überraschend - ist dieser Ansatz ein Beleg dafür, dass Konzepte mit relativ geringem semantischem Gehalt (Semantik verstanden als Informationsgehalt) für ein breites Spektrum von Fällen genutzt werden können (anders gewendet: ein hohes Maß an Wiederverwendbarkeit aufweisen), andererseits eben auch vielfältige Interpretationen zulassen und dadurch die Integrität eines Systems gefährden.

3.2 Generalisierungshierarchien

Wenn im Rahmen der intendierten Anwendung die Unterschiede zwischen verschiedenen Produktarten so groß sind, dass ein flaches Produktkonzept nicht mehr hinreicht, ist es naheliegend, unterschiedliche Klassen einzuführen, um diese Varianz abzudecken. Um darüber hinaus konzeptionelle Redundanz zu vermeiden und damit die Vorteile eines höheren Abstraktionsniveaus nutzen zu können, bietet es sich an, die verschiedenen Produktklassen durch Generalisierungs-/Spezialisierungsbeziehungen zu verbinden. In Abb. 2 ist ein Beispiel für eine entsprechende Generalisierungshierarchie dargestellt.

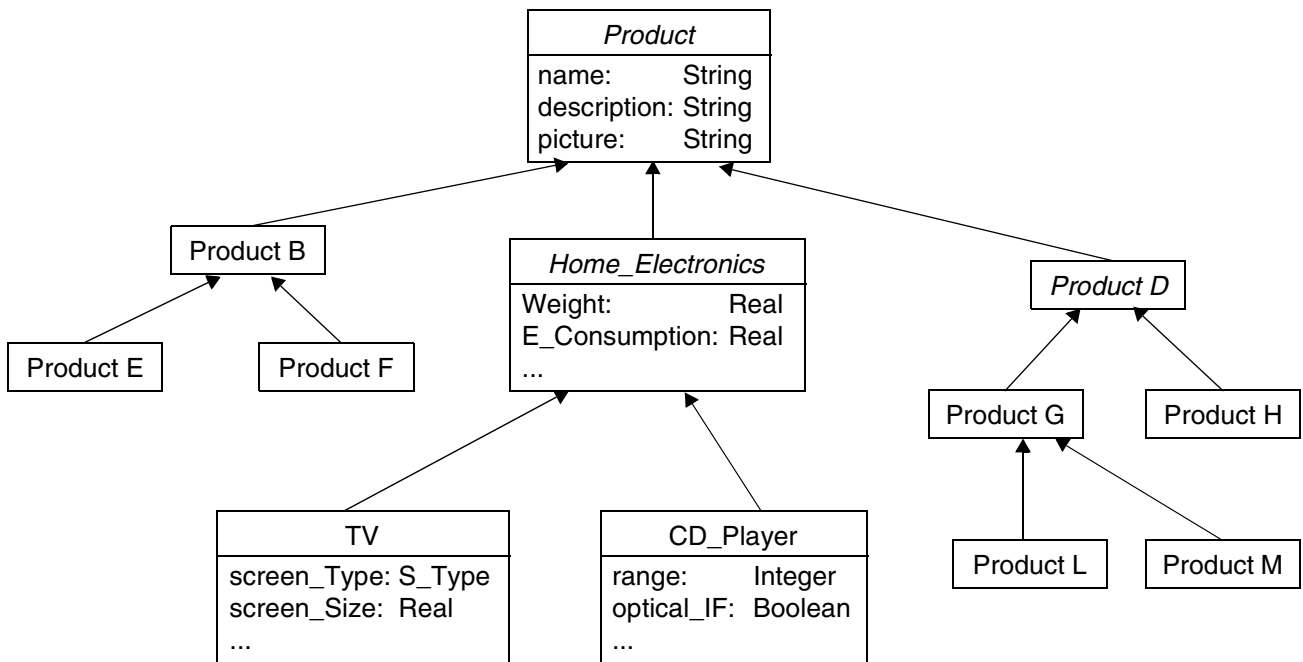


Abb. 2: Beispiel für Generalisierungshierarchie

Im Unterschied zu dem zuvor skizzierten flachen Produktkonzept dienen die Instanzen einer Klasse hier offensichtlich nicht der Einführung neuer Produktarten. Dazu werden vielmehr spezielle Klassen verwendet. Dessen ungeachtet mag es sein, dass es nur eine Instanz einer solchen Klasse gibt, da die Abbildung einzelner realweltlicher Objekte auf eine Instanz zu aufwendig ist (s. 3.1). Grundsätzlich ist die Qualität einer solchen Generalisierungshierarchie abhängig davon, ob die Gemeinsamkeiten, die die Grundlage von Generalisierungsbeziehungen sind, für die jeweilige Anwendungsdomäne wesentlich und im Zeitverlauf stabil sind. Das kann nur im Einzelfall geprüft werden. Im folgenden beschränken wir uns auf die allgemeine Beurteilung eines solchen Ansatzes zur Modellierung von Produkten hinsichtlich der skizzierten Anforderungen.

Da die Zahl der in einer solchen Generalisierungshierarchie verwendeten Produktklassen beliebig ist, können natürlich auch beliebig viele Produktarten abgebildet werden (Anforderung 1). Die Einführung einer neuen Produktart erfordert allerdings die Erweiterung des Datenbankschemas bzw. des Codes (Anforderung 2). Die Anforderungen 3 und 4 lassen sich besser erfüllen als bei flachen Produktkonzepten, da konzeptionelle Besonderheiten von Produktarten als solche abgebildet werden. Aus dem gleichen Grund sind die Voraussetzungen für die Auswertung des Kundenverhaltens (Anforderung 5) günstiger. Da es möglich ist, die Eigenschaften einer Produktart differenziert und ggfs. restriktiv zu beschreiben, kann die Bandbreite unsinniger Initialisierungen deutlich eingeschränkt werden (Anforderung 6). Die Abbildung von Produktvarianten (Anforderung 7) ist in diesem Ansatz zunächst nicht vorgesehen. Er ist allerdings auch für entsprechende Erweiterungen schlecht geeignet. Im einfachsten Fall ist eine Variante dadurch gekennzeichnet, dass einzelne Merkmale bestimmte Zustände aufweisen (etwa eine bestimmte

Farbe). Wenn man diesen Fall mit einer zusätzlichen Klasse, die jeweils mit einer Produktklasse assoziiert ist, abdecken wollte, müßte man alle Attribute der Produktklasse, deren Ausprägungen variieren können, übernehmen - auch wenn man im Einzelfall u.U. nur ein Attribut benötigt. Wenn man darüber hinaus daran denkt, dass eine Variante zusätzliche Merkmale aufweist (wie etwa ein Schiebedach bei einem PKW), die andere Merkmale bedingen (z.B. ein bestimmtes Ausstattungspaket) oder ausschließen, ist die Abbildung einer Produktart auf *eine* Klasse nicht mehr hinreichend. Dieses Argument trifft umso mehr für Konfigurationen (Anforderung 8) zu, denn eine Variante kann ja als eine bestimmte Konfiguration interpretiert werden. Hinsichtlich der Preisbildungsmechanismen (Anforderung 9) ist dieser Ansatz ähnlich zu bewerten wie der vorherige - bis auf den Umstand, dass hier eine differenzierte Zuordnung von Preisbildungsmechanismen zu Produktarten möglich ist. Da dieser Ansatz eine semantisch gehaltvollere Beschreibung von Produktarten erlaubt, sind die Chancen, dem durch externe Schnittstellen spezifizierten Informationsbedarf gerecht zu werden, deutlich besser als beim ersten Ansatz.

Die folgende Tabelle stellt die vergleichende Bewertung in vereinfachter Form da:

Anforderung	"flaches Konzept"	Spezialisierung
1	+	+
2	+	-
3	~	~
4	~	~
5	o	+
6	-	+
7	-	-
8	-	-
9	~	~
10	-	+

+ Anforderung erfüllt - Anforderung nicht erfüllt o Anforderung eingeschränkt erfüllt ~ Ansatz indifferent

4. Ein Metamodell-orientierter Ansatz

Beide bisher diskutierten Ansätzen weisen die erhebliche Schwäche auf, dass sie die Beschreibung von Varianten und individueller Produktkonfigurationen nicht in befriedigender Weise erlauben. Daneben bleibt der für den ansonsten leistungsfähigeren Ansatz (Spezialisierungshierarchie) der u.U. schwerwiegende Nachteil, dass die Einführung neuer Produktklassen einen Eingriff in den Code bzw. das Schema erfordert. Beide Ansätze sind also nicht zufriedenstellend. Da die Vermeidung von Änderungen am Programmcode bzw. am Schema im Hinblick auf den reibungslosen Betrieb eines Systems eine nicht zu vernachlässigende Forderung darstellt, sollte sie auf jeden Fall erfüllt werden. Die Suche nach einem alternativen Ansatz zielt damit vor allem auf eine Überwindung des Konflikts zwischen Anwendungsbreite (möglichst alle denkbaren Produktarten sollten von Beginn an abgedeckt sein) und semantischem Gehalt der verwendeten Konzepte. Angesichts des Umstands, dass die zu modellierenden Produkte ex ante nicht bekannt sind, scheint es kaum möglich, diesen Konflikt aufzulösen. Bei genauerer Betrachtung läßt sich diese Einschränkung allerdings relativieren: Auch wenn uns konkrete Erscheinungsformen bestimmter Produktarten im voraus nicht bekannt sein mögen, können wir durchaus gehaltvolle Vorstellungen darüber entwickeln, welche Produktarten prinzipiell möglich sind oder - anders gewendet - welche Sprachkonzepte man benötigt, um eine Produktklasse im Sinne der genannten Anforderungen zu beschreiben.

4.1 Zentrale Konzepte

Die dargestellte Problematik ist nicht neu. Sie gilt in gewisser Weise, wenn auch auf einem deutlich höheren Abstraktionsniveau, grundsätzlich für die Erstellung von Modellen im Rahmen der Software-Entwicklung: Auch da weiß man zunächst nicht um die Besonderheiten der zu modellierenden Domänen. Wenn man eine Vorstellung davon hat, wie solche Domänen im Sinne der jeweiligen Aufgabenstellung strukturiert bzw. beschrieben werden können, kann man zumindest eine aufgabenspezifische Modellierungssprache spezifizieren. Auf diese Weise wird zumindest indirekt eine zielgerechte Beschreibung noch nicht bekannter Sachverhalte unterstützt. Die Spezifikation von Beschreibungskonzepten setzt eine Metasprache voraus, die z.B. durch ein Metamodell konkretisiert wird. Daneben weist das Problem eine deutliche Ähnlichkeit zur Beschreibung der Komposition von Aggregaten auf, wie sie etwa bei der Produktionsplanung und -steuerung unter Rückgriff auf Stücklistenmodelle eine Rolle spielt. Der im folgenden skizzierte Ansatz weist denn auch Parallelen sowohl zur Metamodellierung als auch zur Modellierung von Aggregaten auf. Abb. 3 zeigt die zentralen Klassen eines entsprechenden Modells.

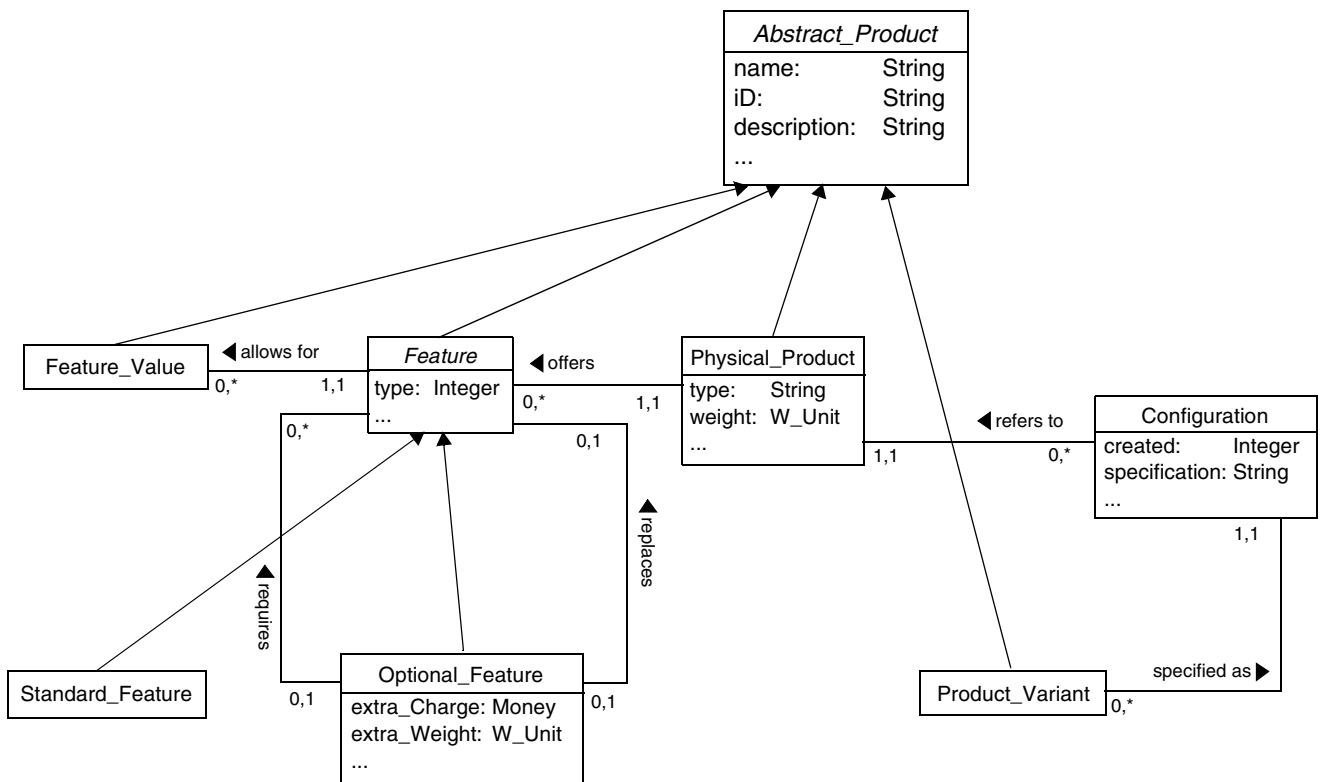


Abb. 3: Metamodell-basierter Ansatz für die Beschreibung von konfigurierbaren Produktklassen

Dieses Modell gilt für alle zu verwaltenden Produktklassen. Eine bestimmte Produktart wird durch miteinander verbundene Instanzen der im Modell enthaltenen Klassen dargestellt. Angesichts der Mehrdeutigkeit des Begriffs "Metamodell" erfordert das Verständnis der hier fokussierten "Meta-" Ebene eine Abgrenzung gegenüber anderen Ebenen. Das hier dargestellte Metamodell ist eine Instanz des Metamodells der verwendeten Modellierungssprache (aus diesem Blickwinkel könnte man also im letzteren Fall von einem Meta-Metamodell sprechen). Dieses Verhältnis ist vergleichbar mit dem zwischen einer Fachsprache und der natürlichen Sprache: Man verwendet Begriffe der einen Sprache, um Begriffe einer spezielleren Sprache einzuführen. Im Hinblick auf den Systementwurf ist allerdings zu berücksichtigen, dass eine solche Differenzierung nicht unmittelbar umzusetzen ist, da übliche Systemarchitekturen bzw. mit ihnen einhergehende Werkzeuge i.d.R. nur zwei Sprachebenen unterscheiden. Deshalb ist das dargestellte Modell in einem systemtechnischen Sinn kein Metamodell, sondern ein Objektmodell. Die angemessene Interpretation der Konzepte muss also durch die jeweilige Anwendung sichergestellt werden. Damit sind Einschränkungen verbunden, auf die noch einzugehen ist.

Sämtliche Anforderungen, die in 2 formuliert sind, werden durch diesen Ansatz erfüllt, wobei wie bei den anderen Ansätzen auch von Preisbildungsmechanismen abstrahiert wird. Im Hinblick auf die Beschreibung von Konfigurationen ist allerdings einschränkend zu berücksichtigen, dass mit diesem Modell nur ein Teil der denkbaren Integritätsbedingungen abgebildet werden (durch die Beziehungen zwischen `Optional_Feature` und `Feature`). Hier stößt man einerseits auf Grenzen der verwendeten Modellierungssprache, andererseits macht eine solche Einschränkung auch im Hinblick auf den intendierten Anwendungsbereich Sinn. Es wird i.d.R. nicht wirtschaftlich sein, wenn eine Handelsplattform die detaillierten Konfigurationsmöglichkeiten eines komplexen Produkts (wie etwa eines PKW) in der gleichen Detailliertheit nachbildet wie der jeweilige Hersteller. Das Modell beschreibt mögliche Konfigurationen. Alle individuellen Konfigurationen werden als Instanzen von `Configuration` verwaltet, wobei das Attribut `specification` der Beschreibung einer individuellen Konfiguration in einer geeigneten Sprache (etwa als Instanz eines korrespondierenden XML-Dokumenttyps) dient. Abb. 4 verdeutlicht diesen Sachverhalt mit einem Beispiel.

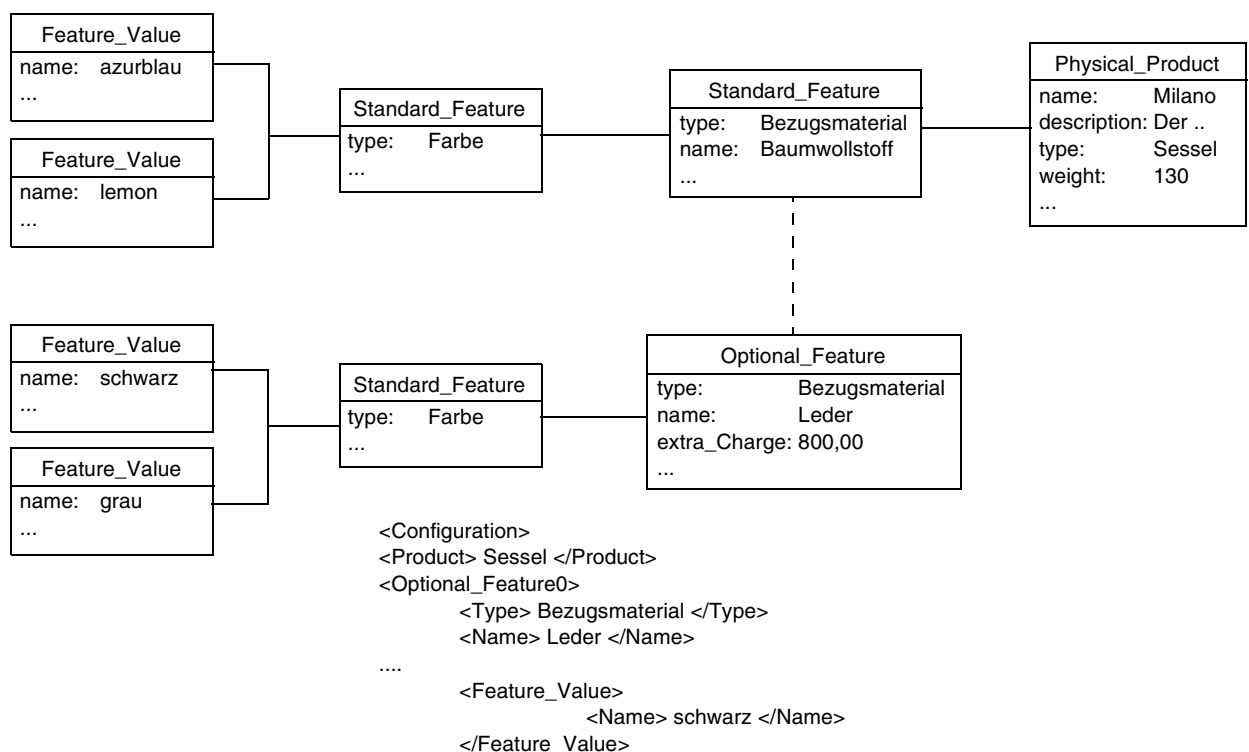


Abb. 4: Beispielhafte Instanziierung des Modells mit korrespondierender Spezifikation einer Konfiguration

Auch wenn dieser Ansatz ein hohes Maß an Flexibilität bei gleichzeitig hoher Integrität unterstützt und insgesamt die dargestellten Anforderungen weitgehend erfüllt, hat er auch Schattenseiten. Sie sind vor allem darauf zurückzuführen, dass er eine Abstraktion einführt, die den Benutzern eines Systems u.U. nicht zugemutet werden kann und die - wie bereits erwähnt - von gängigen Werkzeugen nicht unterstützt wird. Wenn z.B. alle Fahrzeugtypen gesucht werden, deren Motor eine Leistung von mehr als 100 KW hat, wäre eine entsprechende Anfrage, entweder in SQL oder einer objektorientierten Programmiersprache, bei Vorhandensein einer speziellen Klasse "PKW" trivial. Hier ist die Spezifikation der Anfrage weniger intuitiv, weil die Anfrage zunächst auf Metakonzpte gerichtet ist. Erst danach erfolgt die Suche auf Instanzen dieser Metakonzpte. Erschwerend kommt hinzu, dass eine wichtige Funktion des Datenbankschemas bzw. systemweiter Klassenspezifikationen entfällt: die Definition einheitlicher Bezeichner (Namen von Klassen, Attributen und Operationen). Um einem Wildwuchs an Bezeichnern entgegenzuwirken, sind zusätzliche Maßnahmen, wie die Einführung eines einheitlichen Wörterbuchs und Regeln für die Datenerfassung, nötig.

Die mit dem Modell verbundene Abstraktion kann Kunden sicher nicht zugemutet werden. Das stellt allerdings auch kein schwerwiegendes Problem dar. Schließlich können die Objekte, die zusammen die Beschreibung einer Produktart darstellen, in ähnlicher Weise auf eine dynamisch erzeugte Benutzungsschnittstelle abgebildet werden wie dies für die speziellen Produktklassen des zweiten hier diskutierten Ansatzes gilt. Demgegenüber kann gegenüber denjenigen, die neue Produktarten erfassen, die Struktur des Modells nicht völlig verborgen werden. So können zwar die rekursiven Beziehungen zwischen `Optional_Feature` und `Feature` ausgeblendet werden, die Erfassung von Produktmerkmalen wird allerdings nicht in allen Fällen allein durch schrittweise Kumulation möglich sein. Vielmehr müssen diejenigen, die neue, komplexere Produktarten erfassen, ein erhebliches Abstraktionsvermögen mitbringen. Sie müssen nämlich entscheiden, welche Merkmale für die Dekomposition eines Produktes auf den verschiedenen Stufen angemessen sind. Bei dem in Abb. 4 dargestellten Beispiel ist etwa zu entscheiden, ob zuerst nach Farbe und dann nach Material oder umgekehrt differenziert werden soll.

4.2 Ergänzende Konzepte

Das in Abb. 3 dargestellte Modell ist nicht hinreichend, um alle produktbezogenen Daten abzubilden. Das gilt u.a. für Preisangaben, Lagerbestände und Verpackungen. Um verschiedene Formen der Preisbildung berücksichtigen zu können, ist es nicht sinnvoll, Verkaufspreise direkt den Produktklassen zuzuordnen. Stattdessen ist es angeraten, Preise bzw. die Preisbildungsmechanismen (wie etwa Auktionen) in assoziierten Klassen zu beschreiben. Auch die Verwaltung von Lagerbeständen empfiehlt ergänzende Abstraktionen, da hier verschiedene Ausprägungen denkbar sind: Die Verwaltung eines eigenen physischen Lager, der Rückgriff auf ein externes, von einem Logistikpartner verwaltetes Lager oder aber die Beschränkung auf Rahmenverträge, in denen die Lieferanten die Lieferung von Mindestmengen in einem bestimmten Zeitraum zusichern. Dadurch erhalten die Verträge gleichsam die Funktion eines virtuellen Lagers.

Da im Unterschied zum traditionellen Einzelhandel die Auslieferung an den Kunden ein obligater Bestandteil der Auftragsabwicklung ist, kommt der Berücksichtigung solcher Daten, die für die Durchführung entsprechender Logistikprozesse benötigt werden, eine hohe Bedeutung zu. Das schließt z.B. Gewicht und Abmessungen der Verpackung ein. Abb. 5 zeigt eine Erweiterung des Modells um die Beschreibung von Verpackungseinheiten. Sie entspricht weitgehend dem bekannten "Composite Pattern". Die der Klasse `Package` in der Assoziation mit `Aggregated_Package` zugeordnete Kardinalität 1,1 ist beabsichtigt. Die in einer Sammelpackung enthaltenden Verpackungen werden nicht jeweils als ein Objekt abgebildet, da ihre Unterscheidung in der Realwelt zu aufwendig wäre (s.o.). Es wird also nur die Verpackungsart referenziert, die jeweilige Stückzahl wird durch das Attribut `no_Of_Items` verwaltet.

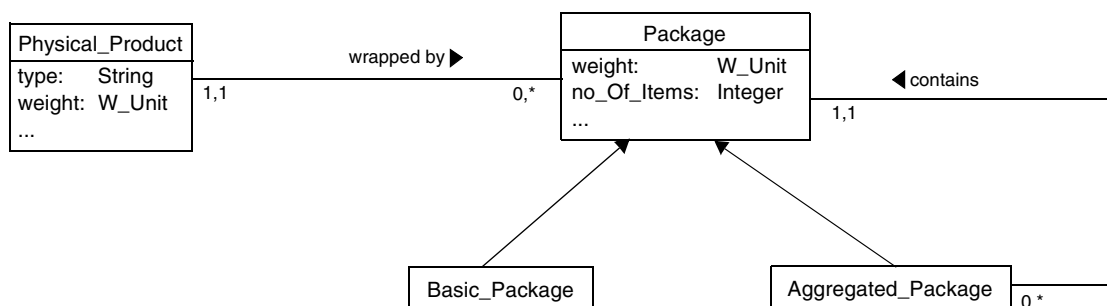


Abb. 5: Ergänzung des Modells um Verpackungen

Falls es nötig wird, einzelne Instanzen der modellierten Produktklassen zu verwalten, sind die Klassen `Physical_Product`, `Configuration` und `Product_Variant` jeweils mit einer Klasse zu assoziieren, deren Instanzen diese Funktion übernehmen.

5. Arbeiten im Umfeld

Angesichts ihrer zentralen Bedeutung für den elektronischen Handel ist es wenig verwunderlich, dass Produktbeschreibungen Gegenstand einer Reihe einschlägiger Forschungsaktivitäten sind. Auch einige Betreiber von Handelsplattformen verfolgen z.T. ambitionierte Ansätze. Dabei geht es vor allem um drei nicht überschneidungsfreie Bereiche: die Kategorisierung von Produkten, die Erstellung von Produktkatalogen und darauf basierenden Austauschformaten sowie die Entwicklung von Ontologien. Die Bildung von hierarchisch geordneten Produktkategorien ist nicht auf eine Beschreibung von Produktarten mittels konzeptioneller Merkmale gerichtet. Stattdessen werden Kategorien durch die Aufzählung der zugeordneten Produktarten festgelegt. Solche Verzeichnisse, die vor allem die Suche nach Produkten unterstützen sollen, sind also mit den Schlagwortkatalogen einer Bibliothek vergleichbar. Ein Beispiel für solche Kategoriensysteme ist das von der UN gepflegte Verzeichnis [UNSP99], das in Teilen in Abb. 6 dargestellt ist. Es entspricht der im Internet gebräuchlichen Verwendung des Begriffs "Meta", dass solche Verzeichnisse auch als Metadaten [MiFe99] bezeichnet werden: "Meta-data are defined lists of keywords describing features of available information." [StSc00a] Es muss nicht betont werden, dass dies ein deutlich anderes Begriffsverständnis ist als das, das dem hier vorgestellten Ansatz zugrundeliegt.

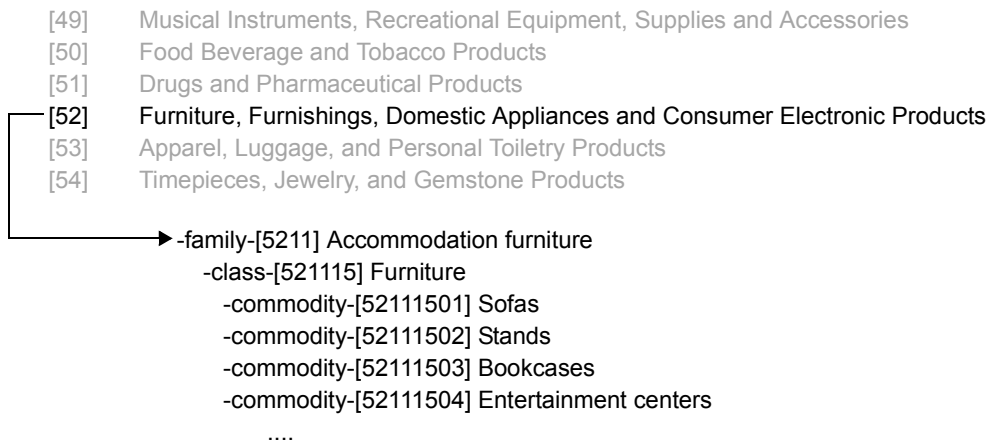


Abb. 6: Ausschnitt aus den Produktkategorien der UN

Produktkataloge (die bereits erwähnten IEPCs) zielen darauf, Produktbeschreibungen zu vereinheitlichen, um Standardwerkzeuge für die Bearbeitung und Übertragung von Produktdaten einsetzen zu können. Sie sind insofern vergleichbar mit dem hier vorgestellten Ansatz, als sie zunächst eine Sprache zur Beschreibung von Produkten festlegen. Es gibt eine Reihe von Unternehmen und Konsortien, die versuchen, solche Kataloge als Standard zu etablieren. Dazu gehören u.a. die "Common Business Library" [Comm99], cXML [Arib99] und die Spezifikation des Konsortiums "Open Buying on the Internet" [OBI98]. In Deutschland hat BMEcat ([HRS99], [HüSc00]) in kurzer Zeit eine erstaunliche Beachtung gefunden. BMEcat spezifiziert eine Menge von XML-Dokumenttypen für den Austausch von Produktdaten. Dabei werden auch Metakonzepte verwendet ("Article_Features", "Feature_Group" ...). Es ist allerdings nicht möglich, Produktvarianten oder Konfigurationsmöglichkeiten zu beschreiben. Grundsätzlich sind all diese Ansätze im Unterschied zu den in 3 und 4 diskutierten nicht unmittelbar der konzeptionellen Modellierung als Bestandteil der Systementwicklung zuzuordnen. Dabei ist die Modellierung letztlich ergiebiger, da es - ein hinreichendes semantisches Niveau der Modelle vorausgesetzt - relativ einfach ist, aus einem instanziierten Modell eine Instanz etwa eines XML-Dokumenttyps zu erzeugen.

Der Begriff "Ontology" wird nicht einheitlich verwendet. Es scheint allerdings die Auffassung vorzuherrschen, dass eine Ontology die (weitgehend) formalisierte Spezifikation von Begriffen einer Domäne enthält. Eine Produktontologie würde also mit ähnlichem Anspruch antreten wie die in diesem Beitrag diskutierten Modellierungsansätze. Dabei steht der Entwurf von Informationssystemen nicht im Vordergrund, sondern die Etablierung einer allgemein akzeptierten Referenzsprache - die aber natürlich für die

Systementwicklung verwendet werden kann. Es bleibt allerdings einschränkend festzuhalten, dass bisher keine elaborierten Ontologien über Produkte vorzuliegen scheinen.

6. Ausblick

Die in diesem Beitrag präsentierten Konzepte zur Modellierung von Produkten sind im Rahmen des Entwurfs eines Referenzmodells für Handelsplattformen im Internet entstanden. Das Referenzmodell besteht gegenwärtig aus einem Objektmodell mit ca. 100 Klassen sowie zahlreichen Modellen von korrespondierenden Geschäftsprozessen. Die Geschäftsprozessmodelle sind auf verschiedenen Abstraktions- und Detaillierungsstufen dokumentiert und wurden zum Teil in Petri-Netze überführt. Neben dem Handel mit physischen Produkten unterstützt das Referenzmodell auch den Handel mit Finanzdienstleistungen. Im Unterschied zu physischen Produkten werden Finanzdienstleistungen mit Hilfe von Spezialisierungshierarchien modelliert. Das basiert auf der Annahme, dass die Varianz von Produkttypen bei Finanzdienstleistungen deutlich geringer ist als bei physischen Produkten. Das Referenzmodell ist gegenwärtig auf Konzepte beschränkt. Die Instanzierung des Modells ist allerdings auch Bestandteil des Forschungsvorhabens. Dazu gehört u.a. die Einführung multilingualer Verzeichnisse für Bezeichner von Produkten und Produkteigenschaften. Zu diesem Zweck werden wir zunächst bereits existierende Verzeichnisse betrachten. Im Hinblick auf die Modellierung von Produkten bietet die Instanzierung zudem die reizvolle Perspektive, aus den im Zeitverlauf erfaßten Produktarten Vorschläge für die Erstellung spezieller Produktklassen, die in eine Generalisierungshierarchie eingefügt sind, zu erzeugen. Das in 3.2 dargestellte Modell wäre dazu allerdings nicht hinreichend. Vielmehr müßten zusätzliche Klassen zur Abbildung spezifischer Produktmerkmale eingeführt werden, um auch Konfigurationsmöglichkeiten darstellen zu können. Eine weitere reizvolle Aufgabe ergibt sich durch die Ergänzung der verwendeten Modellierungssprache um Konzepte zur Beschreibung zusätzlicher Integritätsbedingungen, deren Handhabung die Nutzer eines solchen Systems nicht überfordert.

Literatur

- [Arib99] Ariba Inc.: cXML/1.1. Ariba Inc. 1999, (<http://www.cxml.org/home/>, 16.6.2000)
- [Comm99] Commerce One: Common Business Library, Version 2.0, rel. 3, (<http://www.commerceOne.com/xml/>, 16.6.2000)
- [Fran99] Frank, U.: Applying the MEMO-OML: Guidelines and Examples. Arbeitsberichte des Institut für Wirtschaftsinformatik der Universität Koblenz-Landau. Nr. 11, Universität Koblenz-Landau 1999
- [HRS99] Hümpel, C.; Renner, T.; Schmitz, V.: BMECat Specification. Version 1.01 (<http://www.bmecat.org/>, 16.6.2000)
- [HüSc00] Hümpel, C.; Schmitz, V.: BMEcat - an XML standard for electronic product data interchange. In: Turowski, K.; Fellner, K. K. (Hg.): XML Meets Business: 1. Deutsche Tagung XML 2000, Tagungsband. 2000, S. 1-11
- [KeGe97] Keller, A.; Genesereth, M.: Using Infomaster to Create a Housewares Virtual Catalogs. In: The International Journal of Electronic Commerce and Business Media, 7. Jg., Nr. 4, 1997, S. 41-44
- [MiFe99] Milstead, J.; Feldman, S.: Metadata: Cataloging by Any Other Name In: ONLINE, January 1999
- [OBI 99] OBI Consortium: Open Buying on the Internet. Technical Specifications, Release V2.1. OBI Consortium (<http://www.openbuy.org/obi/specs/>, 16.6.2000)
- [StSc00a] Stanoevska-Slabeva, K.; Schmid, B.: Internet Electronic Product Catalogs: An Approach Beyond Simple Keywords and Multimedia. In: Computer Networks, 32. Jg., Nr. 6, 2000, S. 701-715

- [StSc00b] Stanoevska-Slabeva, K.; Schmid, B.: Cross-Supplier Bundling of Tourist Products with Multi-Vendor Electronic Catalogs. Erscheint in: Hansen, H. (Hg.): Proceedings of the European Conference on Information Systems 2000
- [TiRo98] Timm, U. J.; Rosewitz, M.: Electronic Sales Assistance for Product Configuration. In: (Hg.): Proceedings of the 11th International Bled Electronic Commerce Conference : Electronic Commerce in the Information Society. 1998.
- [UNSP99] United Nations Standard Products and Services Classification Code Organization: United Nations Standard Products and Services Classification. United Nations Standard Products and Services Classification Code Organization, 1999 (<http://www.spssc.org/browse/>, 16.6.2000)