

Konstruktion von Referenzmodellierungssprachen

Ein Ordnungsrahmen zur Spezifikation von Adaptionsmechanismen für Informationsmodelle

Die Autoren

Jörg Becker
Patrick Delfmann
Ralf Knackstedt

Prof. Dr. Jörg Becker
Dipl.-Wirt.Inform. Patrick Delfmann
Dipl.-Wirt.Inform. Ralf Knackstedt
European Research Center for Information Systems
Westfälische Wilhelms-Universität Münster
Leonardo-Campus 3
48149 Münster
{becker | patrick.delfmann | ralf.knackstedt}@ercis.de

Dieser Beitrag ist im Rahmen des vom Bundesministerium für Bildung und Forschung (BMBF) geförderten Forschungsprojekts „Referenzmodellierung 2006 (RefMod06)“ entstanden. Projektträger ist das Deutsche Zentrum für Luft- und Raumfahrt (DLR). Förderkennzeichen: 01 IS C05 A.

spezifische Adaption der Referenzmodelle kann mit erheblichem Aufwand verbunden sein. Der Förderung der Wirtschaftlichkeit der Referenzmodellierung durch eine gezielte methodische Unterstützung des Adaptionsprozesses kommt daher besondere Bedeutung zu.

Vorgehensmodelle zur Referenzmodellierung betonen, dass der eigentlichen Modellierung eine Definition des Problemereichs vorangehen muss [Schü98, 184ff.; Schl00, 78]. Diese umfasst dabei häufig die Eingrenzung der betrachteten betriebswirtschaftlichen Domäne auf Basis der Branche bzw. des Wirtschaftszweiges und sonstiger Unternehmensmerkmale. Weitere Eingrenzungen werden mithilfe von Beschreibungen der adressierten Anwender vorgenommen, wobei die verfolgten An-

wendungszwecke und ergänzende personen- bzw. rollenbezogene Merkmale berücksichtigt werden können. Zum Teil wird hervorgehoben, dass abgestimmt auf die jeweilige Problemstellung eine Konstruktion bzw. Auswahl der anzuwendenden Modellierungssprachen erfolgen muss (vgl. [BDKK02, 34ff.]; vgl. zur Begründung den Grundsatz der Relevanz der Grundsätze ordnungsmäßiger Modellierung [BeSc04, 126]). Vor dem Hintergrund der Bedeutung des Adaptionsprozesses ist zu fordern, dass darüber hinaus eine bewusste Gestaltung der Adaptionsunterstützung vorgenommen wird.

Bestehende Referenzmodellierungsansätze [z. B. Rohl95; Lang97; Remm97; Ohle98; Schü98; Schw99; Hamm99; Hars99; Schl00; Wolf01] bieten dem Referenz-

Kernpunkte

Im Rahmen der Anwendung von Referenzmodellen werden fremde Ausgangslösungen in spezifische Informationsmodelle überführt. Durch Integration von Adaptionsbedarfshinweisen für den Referenzmodellierer lässt sich die Effizienz und Effektivität dieses Adaptionsprozesses gezielt fördern.

- Es wird ein Überblick über die unterschiedlichen Formen der Unterstützung der Adaption von Referenzmodellen geboten. Neben Modellvarianten generierenden Mechanismen der Konfiguration eignen sich die Aggregation, Instanziierung, Spezialisierung und Analogiekonstruktion als nicht generierende Formen der Modelladaption zur Unterstützung der Referenzmodellierung.
- Anhand von Referenzmodellbeispielen, Meta- und Meta-Metamodellen werden konkrete Gestaltungshinweise für den Referenzmodellkonstrukteur zur Durchführung der notwendigen Modellergänzungen gegeben.
- Die unterschiedlichen Ausprägungen der Kombination von Mechanismen der Konfiguration und der nicht generierenden Adaption werden vorgestellt.

Stichworte: Referenzmodellierung, Methodenentwicklung, Modellvariantenmanagement, Multiperspektivität, Konfiguration, Informationsmodellkomponenten, Metamodellierung, Methodische Informationssystemarchitektur

1 Einleitung

Die Effektivität und Effizienz der fachkonzeptionellen Modellierung kann erhöht werden, indem bestehende Modelle als Ausgangslösungen für die Entwicklung spezifischer Modelle genutzt werden [Schü98, 367ff.; BeSc04, 80–83]. In einer entsprechenden Modellbeziehung werden als Ausgangslösung dienende Modelle als Referenzmodelle bezeichnet (zu einer engeren Begriffsfassung vgl. [Schü98, 69ff.]). Die

renzmodellersteller dabei nur eingeschränkte Hilfestellung, da in ihnen in der Regel zwar eine Verzahnung zwischen Adaptionsunterstützungsmechanismen und Modellierungstechniken erfolgt, diese aber nicht expliziert ist (eine Ausnahme bildet z. B. [Broc03], wo auch eine von konkreten Modellierungstechniken losgelöste Diskussion der Adaption erfolgt). Darüber hinaus wird in den bestehenden Referenzmodellierungsansätzen häufig nur ein Teil gängiger Formen der Adaptionsunterstützung berücksichtigt, ohne dass diese Auswahl immer hinreichend nachvollziehbar begründet wird.

Einzelne Formen der Adaption werden darüber hinaus zum Teil als besonders zukunftsweisend propagiert. Eine herausragende Stellung hat dabei die Komponentenorientierung erlangt [Szyp02; Wesk99; Turo01; ABCF02]. Hierbei wird dem Prinzip der „Auswahl und Verbindung“ anstelle z. B. des Prinzips der „Auswahl und Anpassung“ das höchste Entwicklungspotenzial eingeräumt [Ortn00, 107].

Als fundierte Grundlage für den Referenzmodellersteller bei der Gestaltung der Adaptionsaspekte seiner zu spezifizierenden Referenzmodellierungssprache wird im Folgenden ein methodischer Ordnungsrahmen vorgestellt. Auf eine Priorisierung einzelner Adaptionsformen wird zugunsten ihrer übergreifenden Kombination verzichtet. Wir fokussieren dabei die inhaltlichen Anforderungen an die Sprachgestal-

tung, die unter Rückgriff auf bestehende Arbeiten zur Referenzmodellierung und auf der Basis von Projekterfahrungen erhoben wurden. Aspekte der Werkzeugunterstützung werden Gegenstand weiterführender Arbeiten sein. Der Beitrag ist so aufgebaut, dass zunächst die Dimensionen des Ordnungsrahmens identifiziert und dimensionsübergreifend bedeutsame Sprachkonstrukte eingeführt werden (Abschnitt 2). Der Ordnungsrahmen unterscheidet dabei generierende und nicht generierende Adaptionsmechanismen. Die für ihre Integration in Spezifikationen von Modellierungssprachen formulierten Referenzlösungen werden zunächst getrennt voneinander vorgestellt (Abschnitt 3 „Generierende Adaption (Konfiguration)“ bzw. Abschnitt 4 „Nicht generierende Adaption“), bevor Ausprägungsformen ihrer Kombination diskutiert werden (Abschnitt 5). Abschließend formulieren wir Entwicklungsperspektiven (Abschnitt 6).

■ 2 Dimensionen und grundlegende Spezifikationen eines Ordnungsrahmens für die Adaptionsunterstützung

2.1 Dimensionen

Zur Strukturierung der Spezifikation der für die Unterstützung der Referenzmodell-

adaption vorzunehmenden Spracherweiterungen wird auf einen Ordnungsrahmen zurückgegriffen, der von drei Dimensionen aufgespannt wird (vgl. Bild 1) [BDKK02, 69f.; Broc03, 259ff.].

■ Die erste Dimension „Mechanismen der generierenden Adaption“ unterscheidet fünf Kategorien von Konfigurationsmechanismen. Die Konfiguration ist dadurch charakterisiert, dass das Referenzmodell Regeln enthält, die festlegen, wie das Referenzmodell in Abhängigkeit von aktuellen Konfigurationsparameterausprägungen anzupassen ist [Schü98; Schw99; Schl00; BDKK02]. Bei der Konfiguration werden entsprechend der Konfigurationsregeln aus einem Gesamtmodell spezifische Modelle abgeleitet, weshalb diese Adaptionsform im Folgenden auch als generierende Adaption bezeichnet wird.

■ Die zweite Dimension „Mechanismen der nicht generierenden Adaption“ unterscheidet Formen der nicht generierenden Referenzmodelladaption. Für diese ist charakteristisch, dass sie zwar die Erzeugung spezifischer Modellvarianten unterstützen, dabei aber Gestaltungsspielräume lassen, die der Referenzmodellanwender selbstständig ausfüllen muss. Der Adaptionsmechanismus Aggregation setzt voraus, dass das Referenzmodell in Komponenten zerlegt ist. Die Komponenten können im Rahmen der Adaption zu neuen Lösungen zusammengesetzt werden [Lang97; Remm97; Hamm99]. Die Kombinierbarkeit kann dabei durch Schnittstellendefinitionen eingeschränkt werden. Der Mechanismus Instanziierung sieht dagegen vor, dass das Referenzmodell mit Platzhaltern versehen wird, die durch zulässige Ausprägungen ihres Wertebereichs gefüllt werden können [Schü98; Schw99; Wolf01]. In vielen Fällen ist eine Anpassung des Referenzmodells durch den Mechanismus Spezialisierung vorgesehen [Rohl95; BeSc04]. Der Detaillierungsgrad des Referenzmodells wird in diesem Fall bewusst eingeschränkt. Die Adaption sieht vor, dass einzelne Modellteile verändert in das spezifische Modell übernommen werden. Eine besonders geringe Vorwegnahme bzw. Beschränkung der Modellanpassung liegt vor, wenn sich die Anwendung des Referenzmodells allein auf Analogieschlüsse durch den Anwender stützt. Zusammenstellungen von Entwurfsmustern verfolgen z. B. diese Strategie [Risi00; FeYu00].

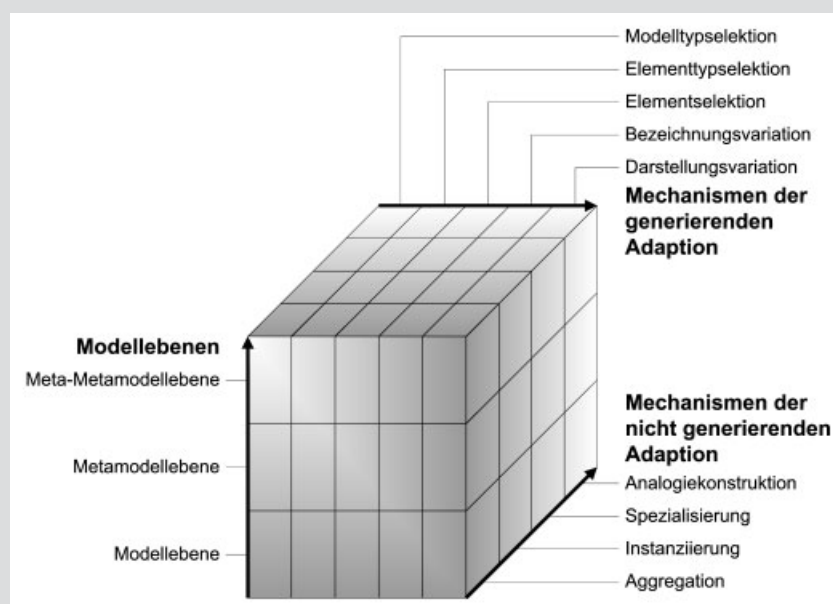


Bild 1 Ordnungsrahmen der Adaptionsunterstützung

■ Die dritte Dimension „Modellebenen“ gliedert sich in drei Modellebenen, wobei die Modelle der jeweils übergeordneten Modellebene die Sprache der Modelle der untergeordneten Ebene beschreiben [Stra96]. Die Spezifikation der Mechanismen umfasst neben der Modell- und der Metamodellebene auch die Meta-Metamodellebene, um Spracherweiterungen, die auf der Metamodellebene z. B. zur Unterstützung der Variantenbildung zu Modelltypen notwendig sind, dokumentieren zu können. Sowohl auf Meta- als auch auf Meta-Metamodellebene werden Entity-Relationship-Modelle (ERM [Chen76]) mit (min, max)-Kardinalitäten [ScSt83, 50f.] verwendet. Dieser Modelltyp wird aufgrund seiner Bekanntheit und einfachen Verständlichkeit sowie seiner Eigenschaft, grundlegende Konstrukte semi-formaler Modellierungssprachen spezifizieren zu können, verwendet. In Fällen, in denen die Mächtigkeit der zugrunde liegenden ERM-Sprache nicht mehr zur Spezifikation einzelner komplexerer Sprachkonstrukte ausreicht, wird zusätzlich auf einfache Grammatiken zurückgegriffen. Diskussionen, welche die Eignung verschiedener Sprachen für die Metamodellierung detailliert beleuchten, sind nicht Gegenstand dieses Beitrages. Ergänzende Aspekte zur Dokumentation von Modellierungstechniken wie Lexika [Ortn97, 95ff.], Gegenstandseinteilungen [Ortn97, 84f.], Metaphern [FeSi01], Prozesse [Stra96, 24ff.] und Organisationsformen [BeKn04, 40] bleiben ebenfalls unberücksichtigt. Für die Modellebene werden bei Bedarf Referenzmodellbeispiele präsentiert.

Die in den Abschnitten 3 und 4 ausführlich vorgestellten Spezifikationen der einzelnen Konfigurationsmechanismen greifen auf den unterschiedenen Modellebenen auf grundlegende Konstrukte zurück, die im Folgenden vorgestellt werden.

2.2 Grundlegende Spezifikationen

Das zentrale Konstrukt der Meta-Metamodellebene ist das *Metamodellelement*, mit dessen Hilfe die Existenz von Modellelementen auf Metamodellebene definiert wird. Da auch für die Metamodellebene ERM verwendet werden, handelt es sich bei diesen Modellelementen um Entitytypen und Relationstypen, die im Meta-Metamodell in dem Entitytypen *Typ* zusammengefasst sind (vgl. für eine vollständige Spezifikation [BDKK02, 131]). Typen

besitzen *Attribute* und können mit *Restriktionen* belegt werden, die eine Einschränkung der den Typen zugehörigen Instanzen erlauben (vgl. Bild 2).

Die Zuordnung von Attributen zu Typen auf Metamodellebene bedingt, dass jedem Modellelement eines Typs (z. B. Prozessfunktionen) die gleichen Attribute zugeordnet werden. Wenn einzelnen Instanzen eines Typs auf Modellebene zusätzliche Attribute erhalten sollen, ist eine Erweiterung des Metamodells notwendig (vgl. Bild 3). Alle Metamodellelemente, für die eine solche instanzbasierte Zuordnung von Attributen vorzusehen ist, werden im Entitytyp *Modellelement*, dem *Instanzattribute zuzuweisen sind*, generalisiert. Diesem wird das *Instanzattribut* zugeordnet, für das wiederum eine bestimmte *Domäne* (Wertebereich) gültig ist.

Der Entitytyp *Prozesselement* gehört beispielsweise zu dem Teil des Metamodells, das die Konstruktion von Prozessmodellen spezifiziert (vgl. Bild 4; zu den aufgeführten Restriktionen vgl. ausführlich [BDKK02, 88ff.]). Das Prozessmetamodell wird im Folgenden exemplarisch und stellvertretend für weitere Modellierungssprachen anderer Sichten verwendet. Es lehnt sich eng an erweiterte Ereignisgesteuerte Prozessketten an (vgl. [Sche01, 125–128], ausführliche Formalisierungen finden sich z. B. in [Kind03; MeNü03], für Anwendungsbeispiele vgl. z. B. [Sche01; BeSc04]).

Für das Verständnis der Spezifikationen zur Adaptionunterstützung sind insbesondere die folgenden Besonderheiten von Bedeutung:

- Eine Prozessfunktion (PF) ist stets eindeutig einer *Funktion* zugeordnet (*PF referenziert Funktion*). Dabei sind Funktionen als solche nicht im Prozessmodell sichtbar. Das Konstrukt der Funktion wird eingeführt, um semantisch identische Aktivitäten in mehreren Prozessen wieder verwenden zu können, ohne prozessspezifische Abhängigkeiten übernehmen zu müssen.
- Funktionen können durch eine *Prozesskomponente* (PK) detailliert werden (*PK detailliert Funktion*). Prozesskomponenten sind *Prozessausschnitte* (PAs). Diesen sind wiederum die für Ereignisgesteuerte Prozessketten grundlegenden Prozesselemente zugeordnet (*PA enthält Prozesselement*).
- Prozesselementen werden bei Bedarf *Ressourcen*, wie z. B. *Organisationselemente*, *Fachbegriffe* etc., zugewiesen. Zur Beschreibung von spezifischen Beziehungen zwischen Ressourcen und Prozesselementen (PE) dient der Relationstyp *PE-Ressourcen-ZuO*, der durch eine trinäre Beziehung zwischen den Entitytypen *Prozesselement*, *Ressource* und *Prozess-Ressourcen-Beziehungstyp* gebildet wird.

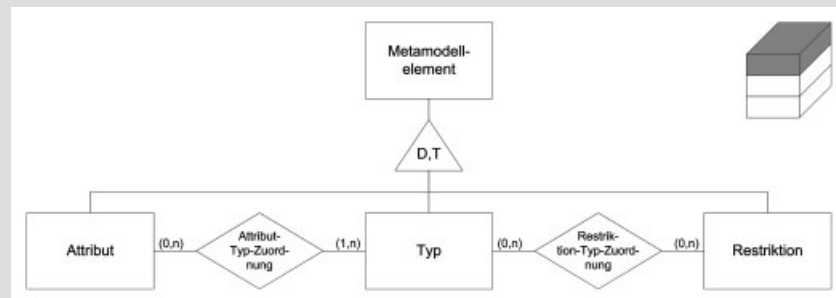


Bild 2 Grundlegende Spezifikationen der Meta-Metamodellebene

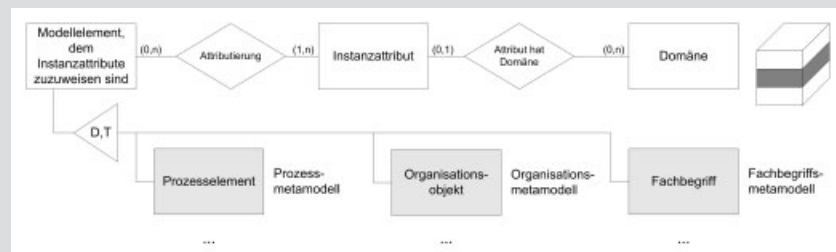


Bild 3 Instanzattribute als grundlegende Spezifikationen der Metamodellebene

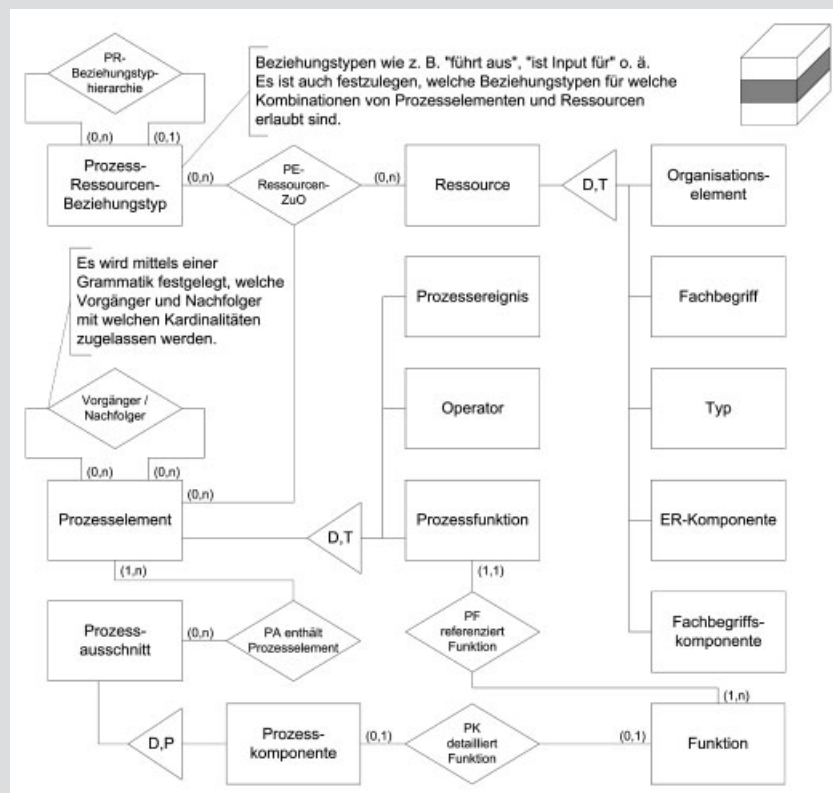


Bild 4 Ausschnitt aus dem sprachbasierten Metamodell der erweiterten Ereignis-gesteuerten Prozesskette (eEPK)

3 Generierende Adaption (Konfiguration)

Der vorgestellte Ordnungsrahmen dient als Navigator für die einzelnen Teile der Spezifikation einer Adaptionunterstützung in Referenzmodellen. Zur besseren Nachvollziehbarkeit werden in einem ersten Schritt die verschiedenen Formen der Konfiguration differenziert nach Modellebenen betrachtet. Die Dimension der nicht generierenden Adaptionformen wird zunächst vernachlässigt. Beschreibungen konkreter Beispiele auf Modellebene lehnen sich dabei an bereits ausführlich in [BDKK02] demonstrierte Beispiele an, die in Bild 5 zusammengefasst dargestellt sind und auf die bei Bedarf verwiesen wird.

Das grundlegende Konzept der Konfiguration von Referenzmodellen beruht auf der konfigurationsparameterabhängigen Generierung von spezifischen Modellen aus einem integrierten Gesamtmodell. Konfigurationsparameter können Unternehmensmerkmale (wie z. B. Branche oder Betriebstyp) [MBGH96; LoHM02] oder

Perspektiven [Woll86, 13; Fran94; DaSh96; Rose98], welche Nutzergruppenanforderungen spezifizieren, sein.

Um die Anwendung der Konfigurationsmechanismen für den Referenzmodellierer effizient zu gestalten, ist es vorteilhaft, Mechanismen mit unterschiedlich weitem Wirkungsgrad bereitzustellen. Deshalb werden Konfigurationsmechanismen nicht allein für die Ebene der Anwendung der Modellierungstechniken vorgesehen (Modellebene), sondern auch für die Ebene ihrer Definition (Metamodellebene). Konfigurative Anpassungen der Metamodelle wirken sich auf alle Modelle aus, die unter Rückgriff auf die jeweilige Modellierungstechnik entstanden sind (vgl. Bild 5; [BKKD03, 905]).

3.1 Modelltypselektion

Die Modelltypselektion berücksichtigt die perspektivenspezifische Relevanz von Modelltypen [Rose98, 11–19]. Modelltypen repräsentieren die Ergebnistypen spezieller Modellierungstechniken, die im Rahmen

der Referenzmodellierungstechnik kombiniert werden können, wie z. B. die der erweiterten Ereignisgesteuerten Prozessketten (eEPK), Entity-Relationship-Modelle (ERM), Fachbegriffsmodelle [Rose98, 11–19] und Organigramme [Groc82, 305]. *Teilmetamodelle* (TMM) beschreiben die einzelnen Modelltypen. *Modelltypvarianten* liefern die Basis, um mittels anderer Mechanismen (vgl. Bild 6) Modelltypen modifizieren zu können. Ein Modelltyp enthält das Maximum aller möglichen erlaubten Elementtypen, welche in einer Modelltypvariante beschränkt werden können. Pro Modelltyp existiert grundsätzlich eine Standardvariante. Der Konfigurationsmechanismus ordnet *Perspektiven* den für sie relevanten Modelltypen zu und unterstützt damit eine grobgranulare Konfiguration des Modellsystems (vgl. auch Bild 5, in dem der Anwendungssystemgestaltung über die Modelltypselektion (1) die Modelltypen eEPK, Entity-Relationship-Modell und weitere nicht benannte zugeordnet werden). Bspw. kann vorgesehen werden, dass Anwendungssystemgestalter im Gegensatz zu Organisationsgestalter keine Fachbegriffsmodelle, sondern statt dessen Entity-Relationship-Modelle zur Verfügung gestellt werden.

3.2 Elementtypselektion

Feingranulare Konfigurationsregeln auf Metamodellebene lassen sich mit der Elementtypselektion vornehmen. Modelltypen werden durch die ihnen zugeordneten Elementtypen und deren Beziehungen untereinander charakterisiert. Der Konfigurationsmechanismus erlaubt es, zu Modelltypen Varianten (vgl. Entitytyp *Modelltypvariante* (MTV) in Bild 6) zu bilden und diese Varianten Perspektiven zuzuordnen (vgl. Relationshipstyp *Perspektiven-Varianten-Zuordnung*). Bspw. unterscheiden sich Varianten der eEPK in den an Funktionen annotierbaren Modellelementtypen, wie z. B. Organisationselemente oder Fachbegriffe. Die Perspektive Organisationsgestaltung erfordert z. B. die Annotation von Fachbegriffen an Funktionen, während für die Perspektive Anwendungssystemgestaltung entsprechende Datentypen/Entitytypen oder Datenmodelle bevorzugt werden (vgl. Bild 5, in dem über die Elementtypselektion (2) Organisationseinheiten als Ressourcen von Prozessmodellelementen ausschließlich der Perspektive Organisationsgestaltung zugeordnet werden).

Varianten zu Modelltypen werden gebildet, indem Elemente des Teilmetamodells, das dem Modelltyp zugrunde liegt, elimi-

niert werden. Instanzen des Entitytyps *Metamodellelement* (MME) sind gleichzeitig Sprachelemente auf Metamodellebene. Werden diese Instanzen durch Anlegen entsprechender Instanzen des Relationstyps *MTV-MME-Beschränkung* entfernt, sind die korrespondierenden Metamodellelemente nicht mehr Bestandteil des Metamodells. Gleiches gilt für die Modelle: Modellelemente, die Instanzen eines ausgeblendeten Metamodellelementes sind, werden aus den Modellen ausgeblendet.

3.3 Elementselektion

Im Gegensatz zur Modelltyp- und Elementtypselektion wird mit der Elementselektion die Auswahl von Elementen der Modellebene verfolgt. Für die Elementselektion werden vier verschiedene Selektionskriterien vorgeschlagen, die sich in der Art der Definition der gewünschten Elementuntermengen unterscheiden:

a) *Elementselektion über Typen*: Sowohl Modellelemente als auch deren Beziehungen können im Rahmen der Modellierung typisiert werden. Der Konfigurationsmechanismus der Elementselektion über Typen ermöglicht eine Zuordnung dieser Typen zu Perspektiven. Im Rahmen der Betrachtung organisatorischer Zusammenhänge stellen z. B. „ist übergeordnet“, „kommuniziert mit“ und „führt aus“ mögliche, perspektivenspezifisch zuzuordnende Typausprägungen dar [Rose98, 11–19]. Entsprechend der Zuordnung dieser Typen zur Perspektive Organisationsgestaltung kann z. B. über die Ein- bzw. Ausblendung einzelner Referenzmodellelemente entschieden werden.

b) *Elementselektion mittels Hierarchiestufenaggregation*: Modelle können hierarchische Strukturen von Modellelementen und Beziehungstypen enthalten, z. B. die disziplinarische Hierarchie einer Organisation. Um unterschiedliche perspektivengerechte Detaillierungsgrade unterstützen zu können, ermöglicht der Konfigurationsmechanismus der Elementselektion mittels Hierarchiestufenaggregation die perspektivenspezifische Festlegung von maximalen Hierarchiestufen und die Aggregation von tiefer in der Hierarchie liegenden Elementen auf die als maximal konfigurierte Hierarchiestufe. Der Mechanismus nimmt unter Auswertung der Baumstrukturen den entsprechenden Austausch der Elemente im eigentlichen Referenzmodell vor. Z. B. wird eine in einem Prozessmodell an eine Funktion

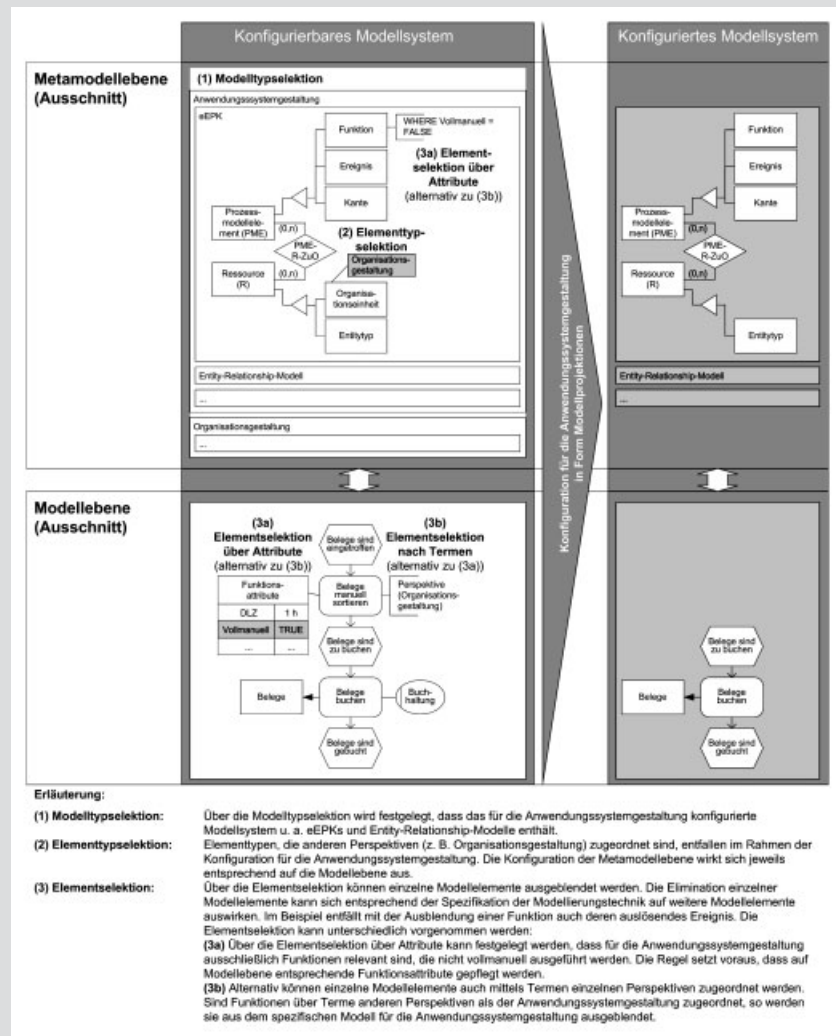


Bild 5 Konfiguration auf Modell- und Metamodellebene

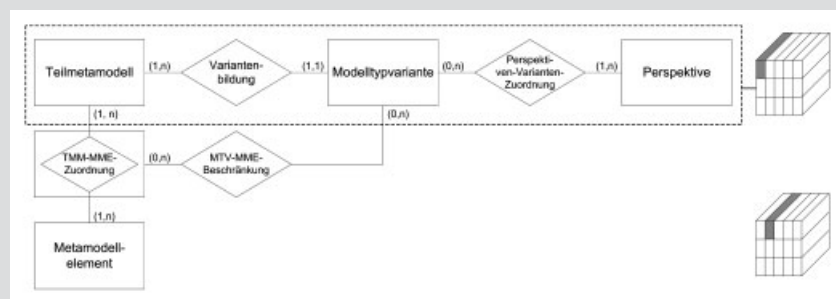


Bild 6 Meta-Metamodellkonstrukte zur Elementtypselektion

annotierte „Stelle X1“ der „Abteilung X“ bei definierter, maximaler Hierarchiestufe der Abteilung fortan in der entsprechenden Perspektive nicht als „Stelle X1“, sondern als „Abteilung X“ angezeigt.

c) *Elementselektion über Attribute*: Die Variantenbildung mithilfe der Elementselektion über Attribute erfolgt über die Auswertung von Eigenschaften, die den Referenzmodellelementen zugeordnet sind. Bspw. kann für die Funktionen ei-

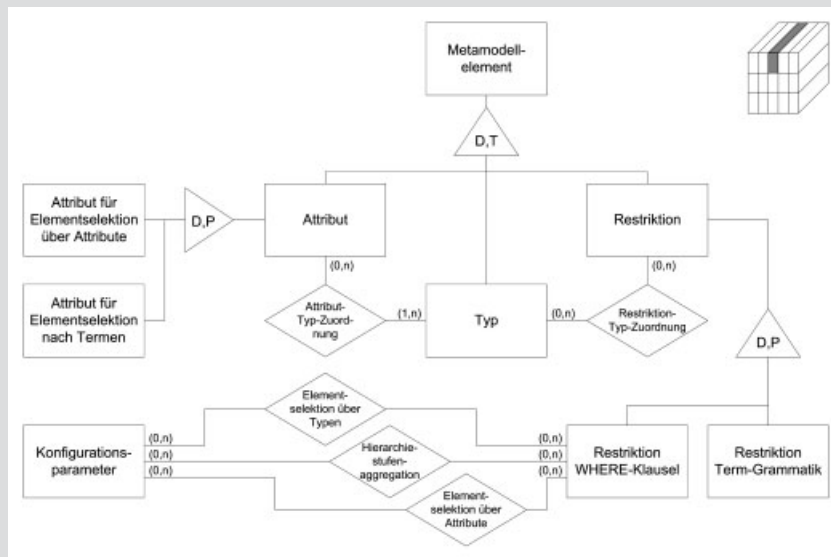


Bild 7 Meta-Metamodellkonstrukte zur Elementselektion

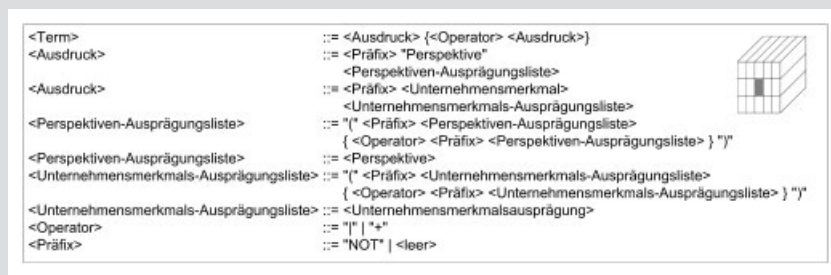


Bild 8 Auszug aus der Grammatik zur Elementselektion nach Termen.

ner eEPK ein Attribut zur Kennzeichnung ihres Automatisierungsgrades vorgesehen werden. Während vollautomatisierte Funktionen im besonderen Fokus der Anwendungssystemgestaltung liegen, spielen diese für die Organisationsgestaltung in der Regel keine Rolle, ebenso wie vollständig manuell zu bearbeitende Funktionen außerhalb des Interessenbereichs der Anwendungssystemgestaltung liegen (vgl. Bild 5, in dem im Rahmen der Elementselektion über Attribute (3a) auf Metamodellebene festgelegt wird, dass vollmanuelle Funktionen nicht relevant für die Anwendungssystemgestaltung sind und auf Modellebene entsprechende Funktionsattributausprägungen gepflegt werden).

d) *Elementselektion nach Termen*: Eine direkte Zuordnung von Modellelementen zu Konfigurationsparameterausprägungen ermöglicht die Elementselektion nach Termen. Mittels des einem Mo-

dellelement zugewiesenen booleschen Terms, z. B. „Perspektive (Organisationsgestaltung)“, wird festgelegt, dass das entsprechende Modellelement nicht zur Perspektive Anwendungssystemgestaltung gehört (vgl. Bild 5, in dem die Funktion „Belege manuell sortieren“ mittels Elementselektion nach Termen (3b) als ausschließlich relevant für die Perspektive Organisationsgestaltung deklariert wird).

Die Elementselektionen über Typen, mittels Hierarchiestufenaggregation und über Attribute basieren auf einem gemeinsamen Konzept. Sie operieren auf den Ausprägungen der Modellelemente. Zur Repräsentation der Elementselektionen wird auf Meta-Metamodellebene die Existenz von Instanzenrestriktionen für Typen auf Metamodellebene definiert (vgl. Entitytyp *Restriktion WHERE-Klausel* in Bild 7). Eine solche Restriktion ist abhängig von der jeweiligen Konfiguration, weswegen sie im

Meta-Metamodell über die drei Relationstypen *Elementselektion über Typen*, *Hierarchiestufenaggregation* und *Elementselektion über Attribute* dem Entitytyp *Konfigurationsparameter* zugeordnet ist. Für die Elementselektion über Attribute wird im Meta-Metamodell zusätzlich ein Entitytyp *Attribut für Elementselektion über Attribute* als Spezialfall des Entitytyps *Attribut* definiert. Diese Spezialisierung ermöglicht es, in der Definition einer Modellierungssprache auf Metamodellebene festzulegen, ob ein Attribut für die Elementselektion genutzt werden kann. Als Konsequenz der Beschränkung von Typausprägungen auf Metamodellebene werden auf Modellebene alle Elemente, die diesen Ausprägungen entsprechen, sowie deren Beziehungen ausgeblendet.

Die Relevanz eines Modellelements für eine ausgewählte Konfiguration hängt bei der Elementselektion nach Termen von einem dem Element zugeordneten Term ab. Terme sind hierbei Attribute in Textform, deren Ausprägungen beschreiben, in welchen Konfigurationen das jeweilige Modellelement zur Verfügung steht. Terme haben zu Auswertungszwecken einer vordefinierten Term-Grammatik zu gehören (vgl. Bild 8; in konkreten Anwendungen sind zusätzlich die zulässigen Ausprägungen von <Perspektive>, <Unternehmensmerkmal> und <Unternehmensmerkmalsausprägung> zu ergänzen).

Realisiert wird die Elementselektion nach Termen, indem im Meta-Metamodell ein spezieller Attributtyp (vgl. Entitytyp *Attribut* und dessen Spezialisierung *Attribut für Elementselektion nach Termen* in Bild 7) für Typen auf Metamodellebene eingeführt wird. Um die formale Korrektheit der Terme sicherzustellen, muss die Existenz einer *Restriktion* definiert werden, die die Formulierung von Termen innerhalb von Attributen dieses Typs in der vorgestellten Grammatik sichert (vgl. die Spezialisierung *Restriktion Term-Grammatik* von Entitytyp *Restriktion*).

Die im Meta-Metamodell definierte Sprachkomponente macht sich auch auf *Metamodellebene* im speziell für die Elementselektion nach Termen angelegten Teilmetamodell bemerkbar (vgl. Bild 9). Hier werden alle Modellelemente aus allen Modellierungssprachen, für die Termzuweisungen möglich sind, zu einem Modellelement generalisiert (vgl. Bild 9, in dem aus Gründen der Übersichtlichkeit nur ein Teil aller für eine Termzuweisung geeigneten Modellelemente aufgeführt ist). Ein für eine Termzuweisung geeigneter *Elementtyp* steht mit dem *booleschen Kon-*

figurationsparameterterm in Beziehung. Dieser hat wiederum ein Attribut, welches den *Termtex*t enthält und das eben dem auf Meta-Metamodellebene definierten Typ (*Attribut für Elementselektion nach Termen*) angehört (weshalb es mit einem „&“ gekennzeichnet ist). Die Restriktion, die an den Term annotiert ist, ist vom Typ *Restriktion Term-Grammatik* und sichert die Formulierung des Termtex

tes in der bereits vorgestellten Grammatik. Die auf Meta-Metamodellebene definierte *Restriktion Term-Grammatik* besitzt also genau eine Instanz auf Metamodellebene, nämlich diejenige, welche die Formulierung von Termen in der vorgestellten Grammatik fordert. Im vorgestellten Fall handelt es sich um die erweiterte Backus-Naur-Form (EBNF) [Loud94, 83ff.].

3.4 Bezeichnungsvariation

Je nach Perspektive können in Referenzmodellen verschiedene Bezeichnungskonventionen gelten. Je nach Benutzergruppe wird bspw. von Rechnung oder von Faktura bzw. von Auftrag oder Bestellung gesprochen, wobei mit den verschiedenen Bezeichnungen jeweils die gleichen Sachverhalte angesprochen werden. Der Konfigurationsmechanismus der Bezeichnungsvariation ermöglicht einen von der Konfigurationsparameterausprägung abhängigen Austausch der Bezeichnungen. Für eine solche Bezeichnungsmenge gleicher Bedeutung (Synonyme) wird bei der Nutzung im konfigurierbaren Referenzmodell ein entsprechender Platzhalter verwendet, an dessen Stelle nach der Konfiguration die zugeordnete Bezeichnung tritt.

Im Meta-Metamodel werden zur Realisierung der Bezeichnungsvariation *Konfigurationsparameter* in Beziehung zu *Synonymgruppenbezeichnungen* gesetzt (vgl. Bild 10) und repräsentieren somit über diese Beziehung die Umbenennungstabelle für die Modellelemente. Diese Tabelle ist modellübergreifend (daher auf Metamodellebene) zu spezifizieren. Synonymgruppenbezeichnungen definieren den führenden Standardbegriff (z. B. „Faktura“), der in Abhängigkeit von der Konfigurations-

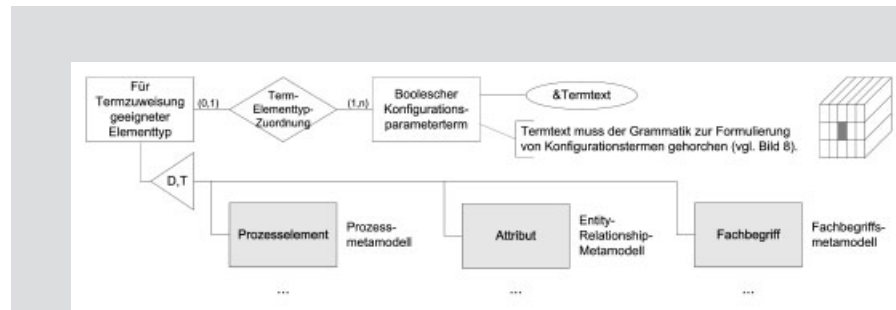


Bild 9 Metamodellkonstrukte zur Elementselektion nach Termen

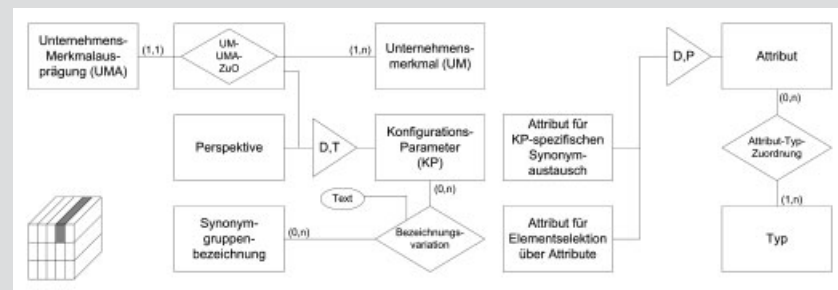


Bild 10 Meta-Metamodellkonstrukte zur Bezeichnungsvariation

parameterausprägung durch den im Attribut *Text* gespeicherten Term (z. B. „Rechnung“) ersetzt wird.

Da die Umbenennungstabelle nicht pauschal für alle Attribute der Modellelemente gelten soll, müssen diejenigen Attribute gekennzeichnet werden, auf welche die Umbenennungstabelle anzuwenden ist. Bspw. kann dies durch ein dem Attributnamen voran gesetztes Sternchen (*) geschehen. Diese Spracherweiterung der Metamodellebene muss konsequenterweise auf der Meta-Metamodellebene durch eine Spezialisierung des Entitytyps *Attribut* in *Attribut für KP-spezifischen Synonymaustausch* modelliert werden.

3.5 Darstellungsvariation

Während mit den bisher vorgestellten Konfigurationsmechanismen die konzeptionellen Sprachaspekte der Referenzmodelle modifiziert werden können, wird mit dem Konfigurationsmechanismus der Darstellungsvariation eine Variation der repräsentationellen Sprachaspekte ermöglicht. Dabei unterscheiden wir die folgenden drei Typen der Darstellungsvariation:

a) *Darstellungsvariation der Symbole:* Durch den Konfigurationsmechanismus der Darstellungsvariation können dem gleichen konzeptionellen Sprachaspekt unterschiedliche Symbole zugeordnet

werden. Bspw. bietet es sich an, für anwenderorientierte Benutzergruppen die klassischen Symbole der EPK durch anschaulichere Bildzeichen auszutauschen, z. B. arbeitende Personen an Schreibtischen anstelle von abgerundeten Rechtecken für Funktionen [Allw98].

b) *Darstellungsvariation der Topologie:* Zusätzlich zu Symbolvariationen sind häufig Variationen in der topologischen Anordnung der Modelle zu beobachten. Bspw. unterscheiden sich Benutzer in den von ihnen präferierten Leserichtungen. Eine von der Standardnotation abweichende Darstellung der eEPK ist die Spaltendarstellung, bei der die Annotation von Ressourcen an Prozessfunktionen in eigens dafür vorgesehenen Spalten erfolgt [Rose98, 11–19].

c) *Darstellungsvariation der Konfigurationsansatzpunkte:* Die Konstrukte zur Repräsentation der Konfigurationsregeln lassen sich ebenfalls unterschiedlich darstellen. Zur Visualisierung der Konfigurationsansatzpunkte eignen sich z. B. sog. Build-Time-Operatoren [Schü98, 235–291], Spaltendarstellungen [BHKS00, 102] und die Annotation von Selektionskriterien an einzelne Modellelemente (vgl. den Term „Perspektive (Organisationsgestaltung)“ an der Funktion „Belege manuell sortieren“ in Bild 5).

4 Nicht generierende Adaption

Für die Spezifikation der nicht generierenden Adaptionenmechanismen wird im Folgenden soweit möglich auf dieselben Konzepte wie bei der Konfiguration zurückgegriffen. Durch Aggregation, Instanziierung, Spezialisierung und Analogiekonstruktion unterstützte Referenzmodellteile werden deshalb im Folgenden als Teil eines Gesamtmodells verstanden. Zur besseren Nachvollziehbarkeit betrachten wir im Folgenden die nicht generierenden Adaptionenmechanismen zunächst einzeln auf den verschiedenen sprachlichen Adaptionesebenen. Entsprechend wird jetzt die Dimension der Konfigurationsmechanismen beim Aufspannen des weiterhin als Navigator dienenden Ordnungsrahmens unterdrückt.

4.1 Aggregation

Da auch Referenzmodellkomponenten als Teil eines Gesamtmodells aufgefasst werden, können diese als Metamodellelement spezifiziert werden, was einer Instanz *Komponente* des Entitytyps *Metamodellelement* auf Meta-Metamodellebene gleich kommt (vgl. nochmals Bild 7). Die Berücksichtigung verschiedener Modelltypen unterschiedlicher methodischer Informationssystemarchitektursichten [Sche01] führt zur Definition von modelltyp- bzw. sichten-spezifischen Komponenten einerseits (vgl. die mehrstufigen Spezialisierungen des Entitytyps *Komponente* auf Metamodellebene in Bild 11) und modelltyp- bzw. sichtenübergreifenden Komponenten andererseits. Letzteren dient der Relationshiptyp *Komponentenhierarchie*, der Komponenten erlaubt, die wiederum Komponenten verschiedener Modelltypen enthalten.

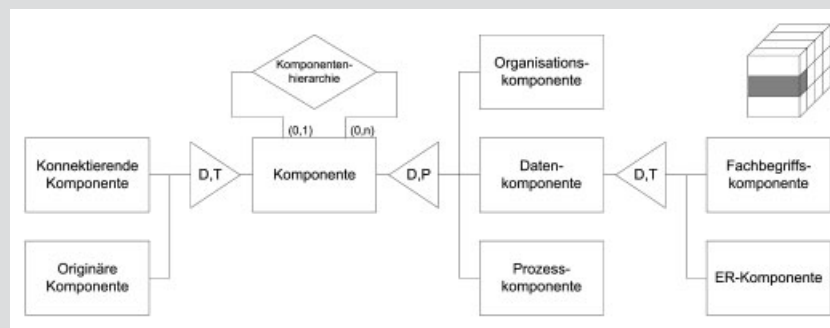


Bild 11 Metamodellelemente zur Aggregation

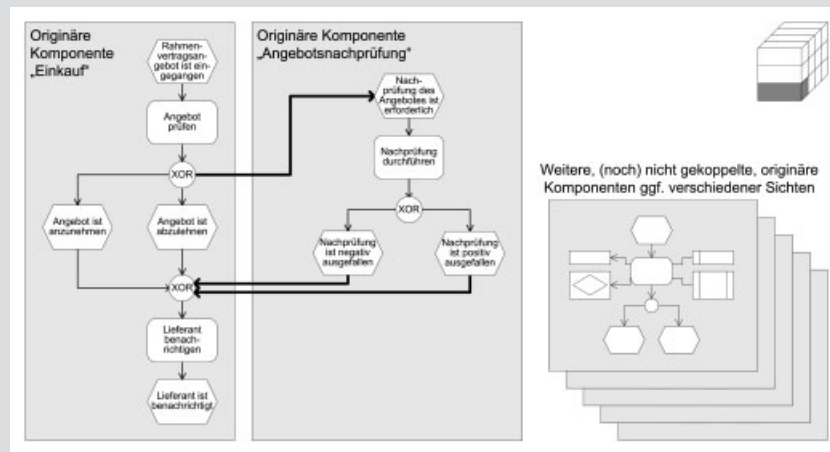


Bild 12 Exemplarische Aggregation zweier Prozesskomponenten auf Modellebene

Die Konnektierung von Komponenten verschiedener Modelltypen bzw. Sichten ist durch die Syntaxbeschreibungen der Modelltypenintegration in den Teilmodellen der Metamodellebene restringiert (vgl. hierzu exemplarisch den Relationshiptyp *PE-Ressourcen-Zuordnung* in Bild 4). Es wird zwischen Komponenten unterschieden, die bereits im Vorfeld vom Referenzmodellierer angelegt wurden (*Originäre Komponente*) und solchen, die dem Nutzer bei Anwendung der Komponenten (Aggregation) dazu dienen, die ausgewählten originären Komponenten zu konnektieren (*Konnektierende Komponente*). Im einfachsten Fall bestehen letztere aus nur einem Modellelement.

Eine exemplarische Anwendung der Aggregation auf Modellebene zeigt Bild 12. Hier werden die Prozesskomponenten *Einkauf* und *Angebotsnachprüfung* durch Kontrollflüsse über Schnittstellen (hier die Ereignisse *Nachprüfung des Angebots ist erforderlich* sowie *Nachprüfung ist positiv* bzw. *negativ ausgefallen* und zwei *XOR-Operatoren*) konnektiert, wobei die syntaktische Konsistenz durch die Metamodellelemente zur Aggregation sowie zur Definition der Sprache von Ereignisgesteuerten Prozessketten sichergestellt wird.

4.2 Instanziierung

Die Instanzierbarkeit von Attributen durch Attributausprägungen und von Modellelementen durch Attribute oder ganze Modelle ist dadurch gekennzeichnet, dass bei der Konstruktion des Referenzmodells vom konkreten Anwendungskontext insofern abstrahiert wird, als Aspekte zunächst vage formuliert oder frei gelassen werden. Der Referenzmodellierer ist auf Stellen im Referenzmodell hinzuweisen, die bei Anwendung folglich konkretisiert werden müssen. Für Modellelementtypen sind entsprechende Attribute bereitzustellen, die einen solchen Hinweis geben können. Zu diesem Zweck wird auf Meta-Metamodellebene ein Attribut *Markierung abstraktes Element* eingeführt, das der Markierung von instanzierbaren Modellelementen dient (vgl. Bild 13).

Analog sind instanzierbare Attribute von Modellelementen zu kennzeichnen. Instanzierbare Attribute unterscheiden sich auf Modellebene von Element zu Element und können daher nicht auf Meta-Metamodellebene eingeführt werden, da entsprechende Spezifikationen für alle Elemente eines Typs gelten würden (vgl. auch Abschnitt 2.2). Deshalb wird auf Metamodellebene auf die Spezifikation von in-

stanzbasierten Attributen zurückgegriffen (vgl. Bild 14). Ein Instanzattribut kann an dieser Stelle, wie auch andere Elementtypen, mit einem Attribut vom Typ *Kennzeichnung abstraktes Element* markiert werden. Der Anwender kann so auf die notwendige Konkretisierung des jeweiligen Attributes hingewiesen werden. Soll die *Domäne* des Attributes ebenfalls vom Anwender im Rahmen einer Instanziierung festgelegt werden, ist dem *Instanzattribut* bei der Modellkonstruktion zunächst noch keine Domäne zuzuordnen. Dies ist möglich, da die Kardinalität des Instanzattributes in Bezug auf die Domäne als (0,1) deklariert ist. Einem Instanzattribut muss also keine Domäne zugewiesen werden. Um zu vermeiden, dass eine Domänenzuordnung bei nicht zu instanzierenden Attributen vergessen wird, wird die Restriktion am Relationstyp *Attribut hat Domäne* eingeführt, welche die Optionalität der Zuweisung von Domänen auf als abstrakt markierte Instanzattribute beschränkt.

Bild 15 zeigt die exemplarische Instanziierung eines Prozesses auf Modellebene. Ein Prozessmodell für die nachträgliche Prüfung eines zunächst nicht bekannten Sachverhaltes wird allgemein gehalten und mit entsprechenden Attributen gekennzeichnet. Diese sind mit den Präfixen *\$A* für den Attributtyp *Markierung abstraktes Element*, *\$ID* für Instanzattribute mit fester Domäne und *\$IF* für Instanzattribute mit freier Domäne markiert.

Für sämtliche Modellelemente wird die Bezeichnung allgemein formuliert und im Zuge der Instanziierung auf einen konkreten Sachverhalt angepasst – in diesem Falle die Nachprüfung eines Angebotes für einen Rahmenvertrag im Einkauf eines Handelsunternehmens. Die Funktion *Nachprüfung durchführen* ist weiterhin durch ein Prozessmodell zu verfeinern.

4.3 Spezialisierung und Analogiekonstruktion

Spezialisierung und Analogiekonstruktion sind durch die größten Freiheiten bei der Anpassung des Modells gekennzeichnet, da Vorgaben zur Abänderung des Modells – abgesehen von syntaktischen Restriktionen – weitgehend fehlen. Bei beiden Adaptionsformen kann der Anwender nichtsdestotrotz darauf hingewiesen werden, für welche Ausschnitte eines Modells eine Spezialisierung bzw. Analogiekonstruktion sinnvoll (oder aber auch zu vermeiden) ist.

Entsprechende *Modellausschnitte*, die als Kandidaten für die Spezialisierung oder Analogiekonstruktion gekennzeichnet wer-

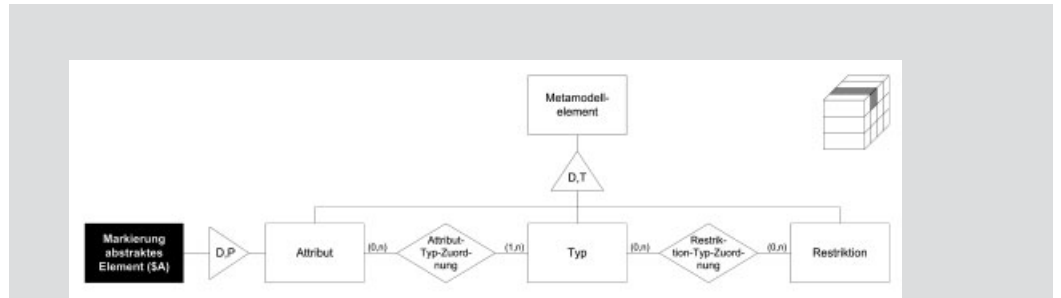


Bild 13 Meta-Metamodellkonstrukte zur Instanziierung

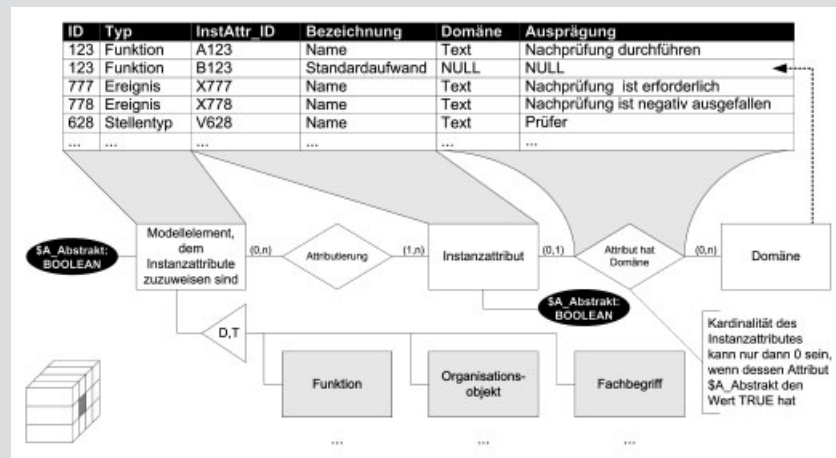


Bild 14 Metamodellkonstrukte zur Instanziierung

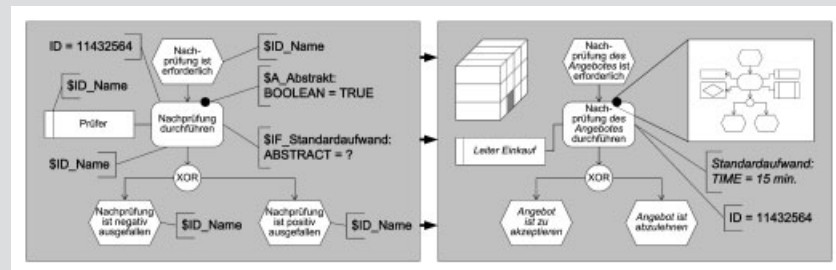


Bild 15 Exemplarische Instanziierung auf Prozessmodellebene

den, sind als Instanzen des Entitytyps *Metamodellelement* auf Meta-Metamodellebene zu spezifizieren. Die Kennzeichnung eines Modellausschnitts für Zwecke der Spezialisierung oder Analogiekonstruktion wird durch Attribute vorgenommen, die dem Typ *Modellausschnittmarkierung* angehören. Auf diese Weise ist sichergestellt, dass Modellausschnitte für andere Zwecke wieder verwendet werden können, wenn sie nicht als spezialisierbar bzw. für die Analogiekonstruktion geeignet gekennzeichnet sind (vgl. Bild 16).

Modellausschnitte werden für Modelle sämtlicher Modellierungssichten spezifi-

ziert. Um ein zusätzliches, getrenntes Metamodellkonstrukt zu vermeiden, werden Modellausschnitte zu Modellkomponenten spezialisiert und so wieder verwendet (vgl. exemplarisch und stellvertretend für die übrigen Sichten die Spezialisierung des *Prozessmodellausschnitts* zur *Prozesskomponente* in Bild 17). Die Attribute zur Kennzeichnung der Modellausschnitte sind lange Textfelder (MEMO), um dem Referenzmodellierer die Möglichkeit zu geben, ausführliche Hinweise für den Modellanwender zur Spezialisierung bzw. Analogiekonstruktion in textueller Form zu verfassen. Sobald diese Attribute leer

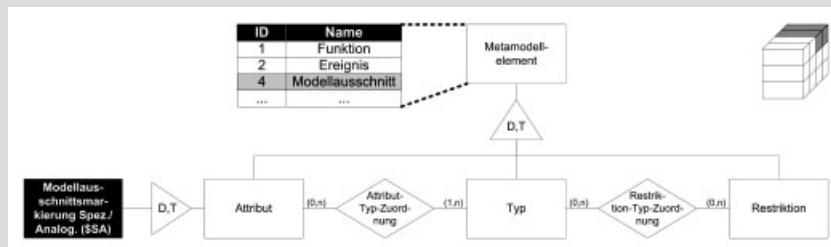


Bild 16 Meta-Metamodellkonstrukte zur Spezialisierung und Analogiekonstruktion

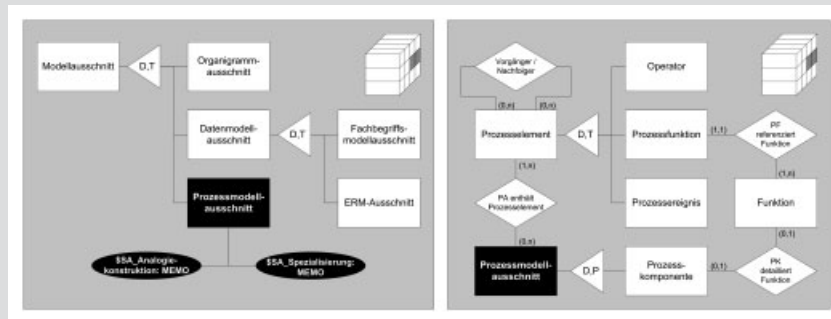


Bild 17 Exemplarische Instanzen der Spezialisierungs- und Analogiekonstrukte im Prozessmetamodell

sind, ist wiederum für den Anwender erkennbar, dass es sich beim jeweiligen Modellausschnitt nicht um einen zu spezialisierenden bzw. zur Analogiekonstruktion vorgesehenen Abschnitt handelt.

Auf Modellebene kann eine Kennzeichnung der für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Abschnitte bspw. durch textuell oder farblich gekennzeichnete Polygone erfolgen, die dem Modellabschnitt unterlegt sind und denen der entsprechende Hinweistext hinterlegt ist.

5 Kombination der generierenden und nicht generierenden Adaption

Die Kombination der verschiedenen Adaptionenmechanismen kann sich je Kombinationspaar verschiedenartig auf die Modelle auswirken. Hierbei wird zwischen Innensicht und Außensicht unterschieden. Die Innensicht betrachtet die Anwendung eines Konfigurationsmechanismus auf die durch nicht generierende Adaptionenmechanismen

markierte Modellausschnitte (z. B. Elementselektion innerhalb einer Modellkomponente). Die Außensicht beschreibt Auswirkungen der Anwendung eines Konfigurationsmechanismus auf den für die nicht generierende Adaption relevanten Modellausschnitt in seiner Gesamtheit (z. B. Nutzung der Elementselektion über Attribute zur Auffindung von relevanten Modellkomponenten). Die Unterscheidung zwischen Innen- und Außensicht wird im Folgenden anhand eines Beispiels ausführlich erläutert. Vor dem Hintergrund der großen Bedeutung, die der Komponentenorientierung zukommt (vgl. Abschnitt 1), wird hierfür die Kombination des Konfigurationsmechanismus der Elementselektion über Attribute mit der Aggregation gewählt.

In der Außensicht kann diese Kombination für die Wiederauffindung von Komponenten in Bibliotheken verwendet werden. Auf Meta-Metamodellebene werden zu diesem Zweck die Konstrukte der Elementselektion über Attribute wieder verwendet (vgl. Bild 18).

Da eine Komponente im Meta-Metamodell als Instanz des *Metamodellelements* definiert ist, können ihr auf Metamodellebene Attribute zugewiesen werden, die dem *Typ Attribut für Elementselektion über Attribute* angehören. Typischerweise sollten dies Attribute sein, welche die Eigenschaften der Komponente charakterisieren [Lang97, 32ff.]. Weiterhin kann zur Einschränkung der Instanzen der Komponente, die den für den Anwender relevanten, ausgewählten Komponenten entsprechen sollen, der Komponente eine Restriktion zugewiesen werden, die diese Einschränkung vornimmt (vgl. *Restriktion WHERE-Klausel* in Bild 18 sowie ihre an die Prozesskomponente annotierte Instanz in Bild 19).

Die Restriktion wird abhängig vom *Konfigurationsparameter* angewendet, welcher wiederum den Anwendungskontext beschreibt, der zur Auswahl von Komponenten führen soll. Im Beispiel soll nach Komponenten gesucht werden, die eine Dokumentprüfung vornehmen, weshalb in deren Namen die Silbe „prüf“ vorkommen soll und die als Output ein hier noch nicht näher spezifiziertes Dokument generieren.

Für die Modellebene bedeutet dies, dass eine in Komponentenbibliotheken abgelegte Menge von Komponenten nach den genannten Attributausprägungen durchsucht wird. Alle Komponenten, die diese Attributausprägungen nicht besitzen, werden dem Prinzip der Konfiguration folgend ausgeblendet. Die übrigen Komponenten

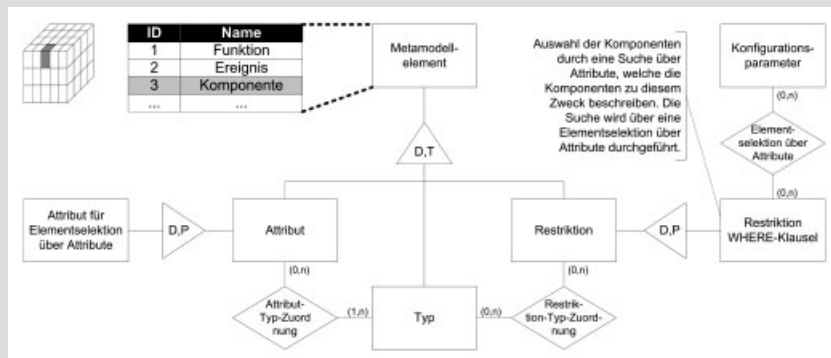


Bild 18 Meta-Metamodellkonstrukte der mit der Elementselektion über Attribute kombinierten Aggregation

stellen das Ergebnis der Suche dar und dienen dem Referenzmodellierer als Auswahlalternativen. Nach einer Einschränkung der Auswahl entsprechend den Vorstellungen des Referenzmodellierers kann eine Konnektierung gemäß den Regeln der Aggregation erfolgen (vgl. Bild 20).

In der *Innensicht* ermöglicht diese Kombination der Mechanismen die Konfiguration von Modellkomponenten selbst. Im Falle der Elementselektion über Attribute lassen sich bspw. innerhalb einer Prozesskomponente Stränge ausblenden, die für den aktuellen Anwendungskontext nicht relevant sind. Die Ausblendung richtet sich dabei nach den Attributen, welche die einzelnen Elemente des Prozessstranges als nicht relevant kennzeichnen (vgl. auch Abschnitt 3.3). Bei einer perspektivenorientierten Betrachtung können z. B. rein manuell auszuführende Prozessstränge ausgeblendet werden, wenn die Modellkomponente Anwendungssystemgestalter zur Verfügung gestellt werden soll (vgl. das Beispiel zur Elementselektion in Bild 5). Die Metamodell- und Meta-Metamodellkonstrukte der Elementselektion über Attribute sowie die der Aggregation können für die Kombination der beiden Mechanismen 1:1 wieder verwendet werden. Der Kombinationscharakter wird hierbei über die Zuordnung der konfigurierten Modellelemente zu Komponenten, bzw. im allgemeinen Fall zu Modellausschnitten der nicht generierenden Adaption, hergestellt.

Wie der Ordnungsrahmen zur Adaptionunterstützung nahe legt, kann grundsätzlich jeder Konfigurationsmechanismus (vgl. Abschnitt 3) mit jedem nicht generierenden Adaptionmechanismus (vgl. Abschnitt 4) kombiniert werden (vgl. Bild 1). Dies belegen die in Tabelle 1 erläuterten Ausprägungen der Adaptionstechnikkombinationen. Für die Innensicht der Instanziierung finden sich dabei keine sinnvollen Kombinationen mit Konfigurationsmechanismen, da hier zum Zeitpunkt der Referenzmodellkonstruktion kein Modell für die Innensicht vorliegt.

6 Fazit und Ausblick

Für die Referenzmodelladaption ist eine Vielzahl von Mechanismen gebräuchlich, die sich in mannigfacher Weise miteinander kombinieren lassen. Dem Referenzmodellkonstrukteur stehen damit umfangreiche Gestaltungsoptionen zur Auswahl, wie er seine Referenzmodellierungstechnik mit ei-

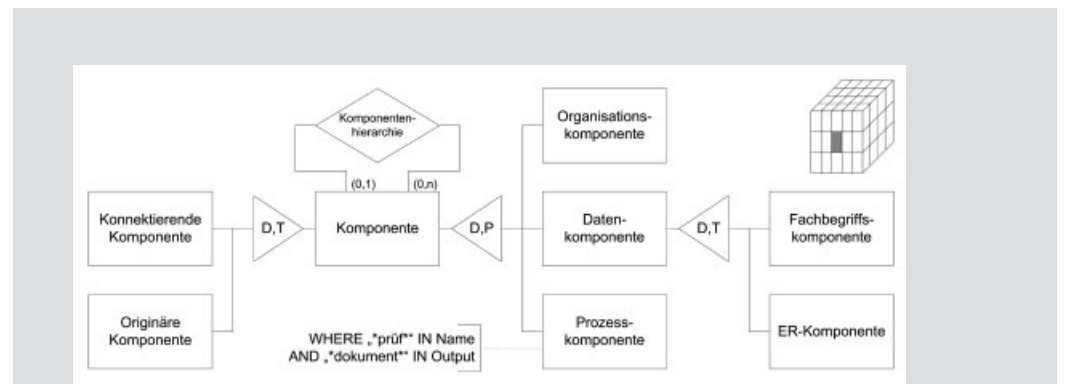


Bild 19 Exemplarische Auswirkung der Elementselektion über Attribute auf die Metamodellkonstrukte der Aggregation

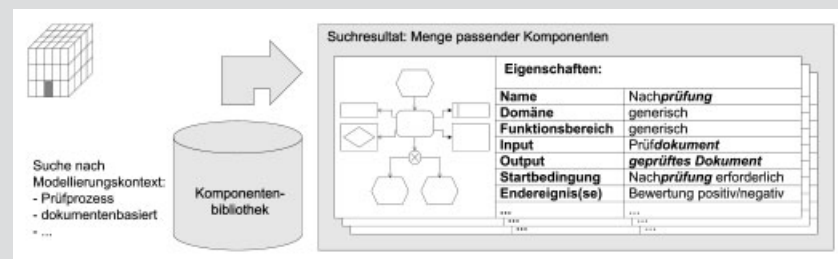


Bild 20 Auffindung von Modellkomponenten mit der Elementselektion über Attribute mit dem Ziel der Aggregation

ner geeigneten Unterstützung für den Referenzmodellierer versehen kann. Der vorgestellte Ordnungsrahmen mit dem in ihm enthaltenen Modellsystem kann durch die Strukturierung dieses Entscheidungsfeldes und dem Angebot konkreter Ausgangslösungen dazu beitragen, bei der Konstruktion von Referenzmodellierungssprachen zu fundierten Lösungen zu gelangen. Dabei musste sich der Beitrag auf die fachkonzeptionellen Aspekte der Spezifikation von Referenzmodellierungssprachen beschränken.

Eine Modellierungstechnik erweitert eine Modellierungssprache um eine prozessorientierte Handlungsanleitung [Stra96, 24ff.]. Hinsichtlich der Handlungsanleitung ist im Zusammenhang mit den Adaptionmechanismen vor allem danach zu fragen, in welcher Reihenfolge diese bei der Konstruktion und Anpassung der Referenzmodelle eingesetzt werden sollen bzw. ob eine simultane Anwendung der Adaptionmechanismen möglich ist und sinnvoll erscheint. Dabei ist insbesondere die Reduktion bzw. Beherrschung der Modellierungs- und Anwendungscomplexität zu berücksichtigen.

Aufgrund der Kompliziertheit und Komplexität von Referenzmodellen mit

Adaptionsunterstützung ist die Akzeptanz der entsprechenden Modellierungstechniken in der Anwendung nur herzustellen, wenn für sie geeignete Modellierungswerkzeuge bereitgestellt werden. Der Ordnungsrahmen ist daher durch eine weitere Dimension der Ebenen der Entwicklung von Anwendungssystemen (hier des Modellierungswerkzeugs) zu ergänzen, für deren Einteilung sich die Unterscheidung von Fachkonzeption, DV-Konzeption und Implementierung anbietet (vgl. [Sche01]). Zu Werkzeugen, die bereits einzelne Bereiche des Ordnungsrahmens umsetzen, vgl. [GeME03; JKSK00; LMBK01; ThMS03; Reit99]. Hier sind konfigurative Anpassungen hauptsächlich in Form der Modelltyp- und Elementtypselektion möglich, bzw. es werden Umgebungen für die Anpassung von Modellen in Form von Metamodelleditoren geschaffen, denen aber eine umfassende Regelbasis zur automatisierten Anpassung fehlt.

Weiterer Forschungsbedarf lässt sich hinsichtlich der Frage prognostizieren, ob für verschiedene Anwendungskontexte nur bestimmte Adaptionformen geeignet sind. Entsprechende Zusammenhänge würden der Konfiguration von Modellierungssprachen hinsichtlich der von ihnen unterstütz-

Tabelle 1 Kombination generierender und nicht generierender Referenzmodelladaption				
Kombinationsausprägungen	Nicht generierende Adaption	Aggregation	Instanziierung	Spezialisierung bzw. Analogiekonstruktion (durch Begründung der Spezialisierungsnotwendigkeit bzw. der Eignung zur Analogiekonstruktion oder durch Anleitungen zur Spezialisierung bzw. zur Analogiekonstruktion unterstützt!)
Konfigurationsarten	Sichten			
Modelltypselektion	Außensicht	Relevanzbeurteilung für Komponenten auf der Basis der Zuordnung von Modelltypen zu Kontexten, sowie Veränderung von einer Komponente zugeordneten Schnittstellenregeln und Suchmerkmalsausprägungen, die von der Einschränkung der Modelltypen des in der Komponente gekapselten Modells abhängt	Eliminierung von instanzierbaren Modellelementen, die sich in ausgeblendeten Modellen befinden, sowie Veränderung des zulässigen Wertebereichs durch Einschränkung der zulässigen Modelltypen, sofern eine modelltypübergreifende Instanziierung vorgesehen ist	Vollständige Eliminierung von für die Spezialisierung bzw. Analogiebildung vorgesehenen Bereichen durch Ausblenden aller Modelle des zugrundeliegenden Modelltyps
	Innensicht	Veränderung des in der Komponente gekapselten, hier notwendigerweise modelltypübergreifenden Modells durch Modelltypselektion		Selektion von Modelltypen des für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen, hier notwendigerweise modelltypübergreifenden Modellbereichs
Elementtypselektion	Außensicht	Veränderung der grundsätzlichen Funktionsweise des Mechanismus, die sich z. B. in unterschiedlichen Sprachkonstrukten zur Schnittstellenspezifikation bzw. zur Unterstützung der Wiederauffindung von Komponenten ausdrückt, was eine Elementtypselektion auch außerhalb von Modelltypenspezifikationen bedingt	Eliminierung aller instanzierbarer Elemente eines ausgeblendeten Elementtyps sowie Veränderung der grundsätzlichen Funktionsweise des Mechanismus, die sich z. B. in unterschiedlichen Sprachkonstrukten zur Schnittstellenspezifikation ausdrückt, was eine Elementtypselektion auch außerhalb von Modelltypenspezifikationen bedingt	Veränderung der Sprachkonstrukte zur Begründung oder Anleitung durch Selektion von entsprechenden Elementtypen, die zu Metamodellbereichen gehören, die nicht der Modelltypspezifikation sondern der Spezifikation von Adaptionsmechanismen dienen, Eliminierung des markierten Bereichs, falls seine Elemente vollständig entfallen
	Innensicht	Veränderung des in der Komponente gekapselten Modells durch Auswahl von Modelltypvarianten		Veränderung (im Extremfall vollständige Eliminierung) des für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Modellbereichs durch Auswahl von Modelltypvarianten
Elementselektion	Außensicht	Selektion ganzer Komponenten im Modell bzw. in Komponentenbibliotheken und von Suchmerkmalen und ihren Ausprägungen sowie Konfiguration der Aussagen zur inhaltlichen Kombinierbarkeit der Komponenten	Ein- bzw. Ausblenden der instanzierbaren Stelle selbst sowie Anpassung des für die Instanziierung gültigen Wertebereichs	Selektion vollständiger, für die Spezialisierung bzw. Analogiekonstruktion vorgesehener Modellbereiche, sowie Anpassung der Begründung oder Anleitungen durch Selektion inhaltlicher Elemente
	Innensicht	Veränderung des in der Komponente gekapselten Modells durch Selektion einzelner Modellbestandteile		Veränderung des für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Modellbereichs durch Selektion einzelner Modellbestandteile
Bezeichnungsvariation	Außensicht	Veränderung der Bezeichnungen von Komponenten und von Suchmerkmalen und ihren Ausprägungen sowie von Modellelementen, die in die Formulierung von Schnittstellenregeln eingehen	Veränderung der Bezeichnungen instanzierbarer Modellelemente und von Modellelementen, die die zulässigen Wertebereiche der Instanziierung definieren	Veränderung der Bezeichnungen der für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Modellbereiche sowie von Modellelementen, die in Begründungen oder Anleitungen inhaltlich eingehen
	Innensicht	Veränderung der Bezeichnungen von Elementen des in der Komponente gekapselten Modells		Veränderung von Bezeichnungen von Elementen des für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Modellbereichs

Tabelle 1 (Fortsetzung)

Kombinationsausprägungen	Nicht generierende Adaption	Aggregation	Instanziierung	Spezialisierung bzw. Analogiekonstruktion
Konfigurationsarten	Sichten			(durch Begründung der Spezialisierungsnotwendigkeit bzw. der Eignung zur Analogiekonstruktion oder durch Anleitungen zur Spezialisierung bzw. zur Analogiekonstruktion unterstützt!)
Darstellungsvariation	Außensicht	Veränderung der Darstellung von Modellelementtypen, die als Komponenten ausgezeichnet werden können, sowie von Suchmerkmalsausprägungen und Ausprägungen der Schnittstellenregeln	Veränderung der Darstellung der Modellelementtypen, die als Platzhalter ausgezeichnet werden können, sowie von Wertebereichsausprägungen	Veränderung der Darstellung von Modellelementen, die in Begründungen oder Anleitungen inhaltlich eingehen
	Innensicht	Veränderung der Darstellungen des in der Komponente gekapselten Modells		Veränderung der Darstellung des für die Spezialisierung bzw. Analogiekonstruktion vorgesehenen Modellbereichs

ten Adaptionenmechanismen den Weg bereiten. Dabei könnten verschiedene Adaptionenmechanismen abhängig von Modellierungskontexten (dies können z. B. betriebswirtschaftliche Domänen oder Modellierungsperspektiven sein) zur Verfügung gestellt werden.

Literatur

- [ABCF02] Ackermann, J.; Brinkop, F.; Conrad, S.; Fette, P.; Frick, A.; Glistau, E.; Jaekel, H.; Kotlar, O.; Loos, P.; Mrech, H.; Ortner, E.; Raape, U.; Overhage, S.; Sahn, S.; Schmietendorf, A.; Teschke, T.; Turowski, K.: Vereinheitlichte Spezifikation von Fachkomponenten. Memorandum des Arbeitskreises 5.10.3 Komponentenorientierte betriebliche Anwendungssysteme. <http://wi2.wiwi.uni-augsburg.de/downloads/gi-files/MEMO/Memorandum-final-2-44-mit-literatur-Web.pdf>, Abruf am 2004-05-17.
- [Allw98] Allweyer, T.: Modellbasiertes Wissensmanagement. In: Information Management & Consulting 13 (1998) 1, S. 37–45.
- [BDKK02] Becker, J.; Delfmann, P.; Knackstedt, R.; Kuropka, D.: Konfigurative Referenzmodellierung. In: Becker, J.; Knackstedt, R. (Hrsg.): Wissensmanagement mit Referenzmodellen. Konzepte für die Anwendungssystem- und Organisationsgestaltung. Heidelberg 2002, S. 25–144.
- [BeKn04] Becker, J.; Knackstedt, R.: Referenzmodellierung im Data-Warehousing. State-of-the-Art und konfigurative Ansätze für die Fachkonzeption. In: WIRTSCHAFTSINFORMATIK 46 (2004) 1, S. 39–49.
- [BeSc04] Becker, J.; Schütte, R.: Handlungsinformationssysteme. Domänenorientierte Einführung in die Wirtschaftsinformatik. 2. Aufl., Frankfurt am Main 2004.
- [BHKS00] Becker, J.; Holten, R.; Knackstedt, R.; Schütte, R.: Referenz-Informationsmodellierung. In: Bodendorf, F.; Grauer, M. (Hrsg.): Verbund-

- tagung Wirtschaftsinformatik 2000. Aachen 2000, S. 86–109.
- [BKKD03] Becker, J.; Knackstedt, R.; Kuropka, D.; Delfmann, P.: Konfiguration fachkonzeptioneller Referenzmodelle. In: Uhr, W.; Esswein, W.; Schoop, E. (Hrsg.): Wirtschaftsinformatik 2003/Band II. Medien – Märkte – Mobilität. Heidelberg 2003, S. 901–920.
- [Bro03] Vom Brocke, J.: Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen. Berlin 2003.
- [Chen76] 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.
- [DaSh96] Darke, P.; Shanks, G.: Stakeholder Viewpoints in Requirements Definition. In: Requirements Engineering 1 (1996) 1, S. 88–105.
- [FeSi01] Ferstl, O. K.; Sinz, E. J.: Grundlagen der Wirtschaftsinformatik. Band 1. 4. Aufl., München, Wien 2001.
- [FeYu00] Fernandez, E. B.; Yuan, X.: Semantic Analysis Patterns. In: Laender, A. H. F.; Liddle, S. W.; Storey V. C. (Hrsg.): Conceptual Modeling – ER 2000 – 19th International Conference on Conceptual Modeling, Salt Lake City, Utah,

Abstract

Construction of Reference Modeling Languages – A Framework for the Specification of Adaptation Mechanisms for Conceptual Information Models

Reference modeling languages differ from each other as they use different model types (such as process or data models) and as they provide different mechanisms that allow an adaptation of the reference model to specific contexts. The developer of the reference modeling language has to decide which adaptation technique he wants to use (e. g. configuration, aggregation, instantiation, specialization or analogy construction) and which of these techniques he wants to integrate into the language specification. In this paper, these adaptation techniques are compared, and reference solutions for the specification of extended reference modeling languages are proposed. The introduced solutions are structured by a methodical framework that assigns modeling examples and meta-models as well as meta-meta-models to the different adaptation techniques. Based on this framework, possible combinations of configurative adaptation mechanisms with aggregative, instantiation based, specialization based and analogy construction based mechanisms are discussed.

Keywords: Reference Modeling, Multi-perspective Information Modeling, Configuration, Model Variant Management, Information Model Components, Meta Modeling, Method Engineering

- USA, October 9–12. Berlin et al. 2000, S. 183–195.
- [Fran94] *Frank, U.*: Multiperspektivische Unternehmensmodellierung. Theoretischer Hintergrund und Entwurf einer objektorientierten Entwicklungsumgebung. München 1994.
- [GeME03] Generischer Modelleditor. <http://wiseweb.wiwi.tu-dresden.de/systementwicklung/projekte/gme/>, Abruf am 2004-05-12.
- [Groc82] *Grochla, E.*: Grundlagen der organisatorischen Gestaltung. Stuttgart 1982.
- [Hamm99] *Hammel, C.*: Generische Spezifikation betrieblicher Anwendungssysteme. Aachen 1999.
- [KSK00] *Junginger, S.; Kühn, H.; Strobl, R.; Karagiannis, D.*: Ein Geschäftsprozessmanagement-Werkzeug der nächsten Generation – ADONIS: Konzeption und Anwendungen. In: WIRTSCHAFTSINFORMATIK 42 (2000) 5, S. 392–401.
- [Kind03] *Kindler, E.*: On the semantics of EPCs: A framework for resolving the vicious circle. Extended abstract. In: *Nüttgens, M.; Rump, F. J.* (Hrsg.): EPK 2003. Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten. S. 7–18. <http://epk.et-inf.fho-emen.de/epk2003/epk2003-proceedings.pdf>, Abruf am 2004-02-27.
- [Lang97] *Lang, K.*: Gestaltung von Geschäftsprozessen mit Referenzprozessbausteinen. Wiesbaden 1997.
- [LMBK01] *Ledeczki, A.; Maroti, M.; Bakay, A.; Karsai, G.; Garrett, J.; Thomason, C.; Nordstrom, G.; Sprinkle, J.; Volgyesi, P.*: The Generic Modeling Environment. In: Proceedings of WISP'2001. Budapest 2001.
- [LoHM02] *Lohmann, M.; Hau, M.; Mertens, P.*: Anforderungsanalyse auf der Basis von Unternehmensmerkmalen. In: *Becker, J.; Knackstedt, R.* (Hrsg.): Wissensmanagement mit Referenzmodellen. Heidelberg 2002, S. 279–289.
- [Loud94] *Louden, K. C.*: Programmiersprachen: Grundlagen – Konzepte – Entwurf. Bonn et al. 1994.
- [MBGH96] *Mertens, P.; Bissantz, N.; Geyer, H.; Hagedorn, J.; Holzner, J.; Ludwig, M.*: IV-Anwendungsarchitekturen für Branchen und Betriebstypen – erörtert am Beispiel der Ergebnisrechnung. In: WIRTSCHAFTSINFORMATIK 38 (1996) 5, S. 485–495.
- [MeNü03] *Mendling, J.; Nüttgens, M.*: EPC Syntax Validation with XML Schema Languages. In: *Nüttgens, M.; Rump, F. J.* (Hrsg.): EPK 2003. Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten. S. 19–30. <http://epk.et-inf.fho-emen.de/epk2003/epk2003-proceedings.pdf>, Abruf am 2004-02-27.
- [Ohle98] *Ohlendorf, E.*: Architektur betrieblicher Referenzmodellsysteme. Konzept und Spezifikation zur Gestaltung wieder verwendbarer Normsoftwarebausteine für die Entwicklung betrieblicher Anwendungssysteme. Aachen 1998.
- [Ortn00] *Ortner, E.*: Wissensmanagement. Teil 1: Rekonstruktion des Anwendungswissens. In: Informatik Spektrum 23 (2000) 4, S. 100–108.
- [Ortn97] *Ortner, E.*: Methodenneutraler Fachentwurf. Stuttgart, Leipzig 1997.
- [Reit99] *Reiter, C.*: Toolbasierte Referenzmodellierung – State-of-the-Art und Entwicklungstrends. In: *Becker, J.; Rosemann, M.; Schütte, R.* (Hrsg.): Referenzmodellierung. State-of-the-Art und Entwicklungsperspektiven. Heidelberg 1999, S. 45–68.
- [Remm97] *Remme, M.*: Konstruktion von Geschäftsprozessen. Ein modellgestützter Ansatz durch Montage generischer Prozesspartikel. Wiesbaden 1997.
- [Risi00] *Rising, L.*: The Pattern Almanac 2000. Boston et al. 2000.
- [Rohl95] *Robloff, M.*: Produktionsmanagement in modularen Organisationsstrukturen. Reorganisation der Produktion und objektorientierte Informationssysteme für verteilte Planungssegmente. München, Wien 1995.
- [Rose98] *Rosemann, M.*: Die Grundsätze ordnungsmäßiger Modellierung. Intention, Entwicklung, Architektur und Multiperspektivität. In: *Maicher, M.; Scheruhn, H.-J.* (Hrsg.): Informationsmodellierung. Referenzmodelle und Werkzeuge. Wiesbaden 1998, S. 1–21.
- [Sche01] *Scheer, A.-W.*: ARIS – Modellierungsmethoden, Metamodelle, Anwendungen. 4. Aufl., Berlin et al. 2001.
- [Schl00] *Schlagbeck, B.*: Objektorientierte Referenzmodelle für das Prozess- und Projektcontrolling. Grundlagen – Konstruktion – Anwendungsmöglichkeiten. Wiesbaden 2000.
- [Schü98] *Schütte, R.*: Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Wiesbaden 1998.
- [Schw99] *Schwegmann, A.*: Objektorientierte Referenzmodellierung. Theoretische Grundlagen und praktische Anwendung. Wiesbaden 1999.
- [ScSt83] *Schlageter G.; Stucky, W.*: Datenbanksysteme – Konzepte und Modelle. 2. Aufl., Stuttgart 1983.
- [Stra96] *Strahringer, S.*: Metamodellierung als Instrument des Methodenvergleichs. Eine Evaluierung am Beispiel objektorientierter Analysemethoden. Aachen 1996.
- [Szyp02] *Szyperski, C.*: Component Software – Beyond Object-Oriented Programming. 2. Aufl., London et al. 2002.
- [ThMS03] The Metaview System. <http://web.cs.ualberta.ca/~softeng/Metaview/system/system.shtml>, Abruf am 2003-01-29.
- [Turo01] *Turovski, K.*: Spezifikation und Standardisierung von Fachkomponenten. In: WIRTSCHAFTSINFORMATIK 43 (2001) 3, S. 269–281.
- [Wesk99] *Weske, M.*: Business-Objekte: Konzepte, Architekturen, Standards. In: WIRTSCHAFTSINFORMATIK 41 (1999) 1, S. 4–11.
- [Wolf01] *Wolf, S.*: Wissenschaftstheoretische und fachmethodische Grundlagen der Konstruktion von generischen Referenzmodellen betrieblicher Systeme. Aachen 2001.
- [Woll86] *Wollnik, M.*: Implementierung computergestützter Informationssysteme. Berlin, New York 1986.