Prozeßontologie als Faktor der Geschäftsprozeßmodellierung[1]

Roland Kaschek

Institut für Wirtschaftsinformatik und Anwendungssysteme
Universität Klagenfurt

Zusammenfassung

Manche Ansätze zur Geschäftsprozeßmodellierung operationalisieren ein imperatives Modellierungsparadigma. Sie behandeln Geschäftsprozesse als Menge von Aktivitäten, die nach bestimmten Regeln ausgeführt werden. Diese Ansätze helfen nur wenig bei der Identifikation der betreffenden Aktivitäten. In ihrem Metamodell fehlt eine Abstraktionsebene zwischen elementarer Aktivität und Geschäftsprozeß. Ihre Prozeßontologie ist flach. Das erschwert Modellierung sowie Wiederverwendung von Modellteilen. Diese Arbeit argumentiert, daß mit geeigneter Prozeßontologie und zugehörigem Prozeßmetamodell das Herstellen von Modellen und die Wiederverwendung von Modellteilen erleichtert werden kann.

1. Einleitung

Die Diskussion um die Geschäftsprozeßmodellierung hat das Verständnis für Schwierigkeit und Bedeutung der Vorgangsmodellierung verbessert. Ansätze, die den Endanwendern die Modellierung von Geschäftsprozessen gestatten sollen liegen, etwa mit ARIS oder SOM vor, vgl. dazu etwa [KP 95, FS 95]. Sinnvolle Anforderungen an Ansätze zur Geschäftsprozeßmodellierung, wie etwa Unterstützung von Wiederverwendung oder einfaches Vorgehen bei der Modellierung unterstützen sie nicht optimal, da sie von einem imperativen Modellierungsparadigma ausgehen.

Das imperative Modellierungsparadigma erfordert die Konstruktion von Geschäftsprozessen aus Operatoren und elementaren Aktivitäten. Üblicherweise gestatten diese Sequenz, Formen der Parallelität und Synchronisation zu beschreiben. Das bedeutet für die Modelliererin, daß sie, ohne Zwischenschritte machen zu können, benennen muß, wie sich durch Ausführung elementarer Aktivitäten ein Geschäftsprozeß vollzieht. Dies beeinträchtigt Entwicklung und Wiederverwendung von Vorgangsmodellen, da keine hinreichend überschaubaren und generischen Substrukturen geschaffen werden können. Daher erschwert dieses Paradigma auch die Beurteilung von Vorgangsmodellen.

Die in dieser Arbeit intendierten Geschäftsprozesse sind Prozesse in Büros oder Verwaltungseinrichtungen. Ihr Charakteristikum ist, daß sie arbeitsteilig sind, ihre elementaren Bestandteile relativ gut beschrieben werden können, ihre Beendigung relativ verläßlich festgestellt werden kann und die Koordination der verschiedenen Aufgabenträger entscheidend ist. Bei den betreffenden Aktivitäten kann es sich z.B. darum handeln (mit Hilfe eines Software Werkzeuges) Daten, z.B. einen Antrag zu bearbeiten, einen Brief zu schreiben, oder ein Hotelzimmer bzw. einen Flug zu buchen.

Das imperative Paradigma findet man auch in der Literatur über Softwareprozesse, vgl. [Hu 96, S. 78]. In dieser Literatur werden die hier genannten Defizite nicht immer hervorgehoben. Das kann daran liegen, daß man mit einer der Varianten des Wasserfallmodells, vgl. [Bl 92], leicht ein Phasenmodell hinzuziehen kann, das die zu vollziehenden Tätigkeiten vorstrukturiert und die Orientierung erleichtert.

Softwareprozesse, wie auch Produktionsprozesse, weisen einige Gemeinsamkeiten mit den hier behandelten Vorgängen auf. Gelegentlich wird versucht Softwareprozesse mit Hilfe von Workflowsystemen zu unterstützen, vgl. etwa [Gr 96]. Beide Prozeßarten werden hier nicht genauer behandelt, da hier die Ansicht vertreten wird, daß ihre Ontologie vermutlich anders ist als die der Geschäftsprozesse. Hinsichtlich der Softwareprozesse wird dazu auf die Argumentation in Abschnitt 5 verwiesen.

Modellieren, Ontologie und Wiederverwendung werden in Abschnitt 2 diskutiert. Abschnitt 3 stellt das imperative Modellierungsparadigma und seine Prozeßontologie dar. Abschnitt 4 behandelt den Vererbungsbegriff. Abschnitt 5 präsentiert eine reichere Prozeßontologie. Den Kern eines Geschäftsprozeßmetamodells bespricht Abschnitt 6. Im 7. Abschnitt werden die Grundzüge eines Vorgehensmodells für die Geschäftsprozeßmodellierung vorgestellt. Abschnitt 8 beinhaltet einen Rückblick sowie einen Ausblick auf die weitere Behandlung des Themas. Die Literaturliste folgt im 9. Abschnitt.

---

[1] Meinungsäußerung anläßlich des Workshops Modellierung'98 in Münster.

2. Modellieren, Ontologie und Wiederverwendung

Unter Modell wird nach [LM 78] eine Vorstellung verstanden, die sich jemand von einem Ding oder Vorgang in seiner Umwelt macht. Unter Modellieren verstehen wir hier das Herstellen, Nutzen oder Modifizieren von Modellen. Die Begriffe, aus denen ein Modell gebildet wird, und ihre Beziehungen zueinander werden gelegentlich mit Hilfe von Metamodellen, das sind Modelle von Modellen, vgl. dazu etwa [KP 97a], präzisiert. Ihre Reichhaltigkeit und Struktur haben Auswirkung darauf, wie das Modellieren vonstatten geht.

Modelle, also auch Metamodelle, vereinfachen. Wie im Fall von Modellen muß man sich auch für Metamodelle fragen, ob die Vereinfachungen angemessen sind. Wie oben gezeigt wurde ist das für Metamodelle zum imperativen Paradigma, das keinen Begriff hat für Prozeßteile zwischen Prozeß und Aktivität nicht der Fall. Um das imperative Paradigma zu verbessern kann die Ontologie der Geschäftsprozesse daraufhin untersucht werden, ob sie zusätzliche Aspekte enthält, die Anlaß zu geeigneten Begriffen im Metamodell geben können. Unter Paradigma wird hier in Anlehnung an [BM 97, S. 151f.] ein Metamodell für Prozeßmetamodelle verstanden.

Wir folgen [KP 95] darin, daß die Wiederverwendbarkeit eines Modellteils insbesondere davon abhängt, ob seine Bedeutung klar und der Teil, sowie seine Einbettung in die Modelle, überschaubar ist. Metamodelle, die Aggregation und Generalisation in geeigneter Weise enthalten, können sich günstig auf die Wiederverwendbarkeit auswirken, da sie Generizität und Kompositionalität fördern, vgl. dazu auch [EW 96].

3. Das imperative Modellierungsparadigma und seine Prozeßontologie

Das imperative Paradigma wird am Beispiel ereignisgesteuerter Prozeßketten erläutert, siehe dazu [KP 95]: Prozesse werden als Netze von Ereignissen und Aktivitäten aufgefaßt, die sich gleichzeitig oder nacheinander ereignen können. Eine Aktivität oder ein Ereignis wird ausgelöst, falls eine entsprechende Bedingung erfüllt ist. Dann wird entweder ein geeigneter Akteur informiert oder das Ereignis bzw. die Aktivität automatisch ausgelöst.

Da [KP 95] sich zur Prozeßontologie nicht explizit äußert wurde die Ontologie des imperativen Paradigmas anhand des Metamodells für die ereignisgesteuerten Prozeßketten geraten. Sie besagt, daß ein Geschäftsprozeß eine Menge von Aktivitäten ist, die nach bestimmten Regeln ausgeführt wird. Über einen Geschäftsprozeß kann daher im wesentlichen gesagt werden, aus welchen Aktivitäten er besteht, welche Regeln in seinem Verlauf angewendet werden können und ob er sich gerade vollzieht oder nicht. Genau genommen kann nicht einmal festgestellt werden in welcher Situation die betreffenden Regeln zur Anwendung kommen, da die Prozeßontologie weder Prozeßzustand noch Prozeßzeit oder ähnliches aufweist.

Eine derart arme Ontologie eignet sich weder zum Modellieren noch zur Wiederverwendung. Im Grunde können mit dem imperativen Modellierungsparadigma nur relativ kleine Modelle erarbeitet werden. Wenn in der Praxis mit diesem Paradigma Erfolge erzielt werden, so liegt dies eher an der Qualifikation der Modellierer, die sich da, wo es nötig ist, über das Paradigma hinwegsetzen, als an der Eignung des Paradigmas bzw. seiner Metamodelle.

4. Vererbung

In objektorientierten Sprachen wird Vererbung zur Ableitung neuer Klassen aus alten genutzt. Diese Ableitung kann als Spezialisierung verstanden werden. Sie wird vielfach als geeignet angesehen Wiederverwendung von Modellbestandteilen zu vereinfachen. Es ist daher naheliegend diesen Begriff zu untersuchen, wenn man Wiederverwendung in der Vorgangsmodellierung betreiben möchte.

Vertreter der Objektorientierung geben gelegentlich, wie etwa in [Ku 95], für den Klassenbegriff folgende Ontologie: Eine Klasse ist eine Einheit aus sogenannten Instanzvariablen und sogenannten Methoden. Sie wird durch Klassifikation[2] gleichartiger Objekte gleichartiger Bedeutung im Diskursbereich gewonnen. Die Instanzvariablen einer Klasse stehen für beobachtbare Eigenschaften oder Merkmale der betreffenden Objekte. Die Methoden hingegen für Operationen, die auf die Objekte einwirken können. Merkmalsausprägungen können nur festgestellt oder verändert werden, indem die Operationen eines Objektes auf dieses angewendet werden. Bezeichnet man für eine Klasse K die Menge der Methoden mit $M(K)$ und die Menge der Instanzvariablen mit $I(K)$ und sind G und S Klassen, so gelten, falls G eine Generalisation von S ist, $I(G) \subseteq I(S)$, und $M(G) \subseteq M(S)$.

Da man Klassen als Aggregation ihrer Instanzvariablen und Methoden auffassen kann liegt folgende Verallgemeinerung nahe: Sind D,D' gleichartige Aggregationen aus Komponenten $K(D)$ bzw. $K(D')$ und ist D' eine Spezialisierung von D, so gilt $K(D) \subseteq K(D')$.

5. Eine reichere Prozeßontologie

Von Menschen genutzte Prozesse vollziehen sich in Abschnitten oder Phasen und sollen ein Ziel erreichen. Phasen sind durch typische Ausprägungen von Prozeßmerkmalen und zeitliches Nacheinander gekennzeichnet.

---

[2] Für eine Definition der wichtigsten Abstraktionskonzepte, insbesondere von Aggregation, Generalisation und Klassifikation, siehe [PM 88].

Ein Vorgang ist eine Einheit aus den Phasen, in denen er sich vollzieht. In jeder seiner Phasen erreicht der Prozeß ein die Phase kennzeichnendes Subziel seines Ziels. In den Phasen eines Prozesses werden von Akteuren Tätigkeiten durchgeführt. Eine Tätigkeiten kann in verschiedenen Phasen erfolgen.

Geschäftsprozesse sind Prozesse. Akteure kooperieren im Prozeßverlauf. Sie wirken in einem gemeinsam Kontext auf ein Ziel hin. Dieses Wirken hat in der Phase, in der es sich ereignet, einen spezifischen Sinn, der in der Erreichung eines Teilziels des betreffenden Subziels besteht. Die Akteure leisten in koordinierter Weise jeweils kooperationslose Beiträge zum Prozeß oder lösen einen Geschäftsprozeß aus. Die kooperationslosen Beiträge können der Ausführungsart nach mehr oder weniger in das Ermessen des Akteurs gestellt sein. Ebenso kann die Koordinationsart den Akteuren hinsichtlich der Koordination Ermessensspielraum lassen.

Phasen und Kontributionen enden mit der Erreichung eines (untergeordneten) Organisationsziels. Die Erreichung des betreffenden Ziels wird durch Urteil des betreffenden Akteurs festgestellt. In Geschäftsprozessen ist dieses Urteil oft relativ sicher, d.h. es muß nachträglich nicht korrigiert werden. Obwohl durch diesbezügliche Irrtümer ausgelöste Rückkehr zu bereits absolvierten Prozeßteilen nicht ausgeschlossen werden kann wird hier angenommen, daß ihr Auftreten signifikant seltener als in Softwareprozessen ist und deshalb zu einer anderen dynamischen Struktur des Vorgangs führt. Dies wiederum führt zu der Annahme, daß das hier gezeigte Metamodelle für Softwareprozesse nur bedingt geeignet ist.

Kooperationen sind Einheiten zielgerichteten Zusammenwirkens von Akteuren zugunsten des Ziels des Vorgangs in dem sie Zusammenwirken. Sie können nacheinander oder gleichzeitig beginnen. Dies trifft auch für Kontributionen, d.h. kooperationslose Prozeßbeiträge von Akteuren, zu. Der Prozeßfortschritt kann in jeder Phase und in jeder ihrer Kooperationen im Arbeitsfortschritt in den betreffenden Kontributionen identifiziert werden.

6. Ein Metamodell für Geschäftsprozesse

Ein Vorgang ist eine Einheit seiner Phasen aber selbst keine Phase. Also kann er als Aggregation seiner Phasen verstanden werden. Bezeichnet man die Menge der Phasen eines Vorgangs V mit $P(V)$ und sind G,S Vorgänge, so ist $P(G) \subseteq P(S)$, falls G eine Generalisation von S ist. S wird daher aus G dadurch abgeleitet, daß die Phasen von G, gegebenenfalls in modifizierter Weise, übernommen, d.h. überschrieben werden und, daß evtl. neue hinzugefügt werden. In analoger Weise ist die Generalisation von Phasen bzw. Kooperationen zu verstehen.

In der Abbildung 1 ist der Kern eines Metamodells dargestellt, das der oben vorgestellten Prozeßontologie entspricht. Es ist in der neuen Notation von OMT, siehe [R* 91], dargestellt und zeigt, daß Geschäftsprozesse aus Phasen, diese aus Kooperationen jene aus Kontributionen und selbige aus Akteur und Aufgabenbeschreibung bestehen. Spezialisierung der wichtigsten Prozeßteile ist mit Hilfe von Assoziationen ins Metamodell integriert. Aus Platzgründen sind hier nicht dargestellt die Einbindung eines Geschäftsprozesses in die Organisation, die ihn nutzt, die Koordination der Akteure und die Zuordnung von Mitteln zu Akteuren. Aus diesem Grund ist hier nicht sichtbar, wie Geschäftsprozesse, Phasen und Kooperationen in dieser Reihenfolge zu Organisationszielen, Subzielen des Organisationsziels und Teilzielen von Subzielen zugeordnet sind. Die Abbildung zeigt unter Nutzung des Modellierungsbegriffs „Qualifyer“ wie Subziele in Geschäftsprozessen, Phasen und Teilziele in Phasen Kooperationen bestimmen.

Die besondere Betonung der Organisationsziele und des Zusammenhangs dieser Ziele in Abbildung 1 kann dadurch motiviert werden, daß erstens Zielorientierung in der Literatur über Entwurfsprozesse als ein wichtiger Ansatz zur Generierung von Entwürfen auftritt, vgl. [BM 97, S.150], und daß zweitens dieser Ansatz zu einem Vorgehensmodell ausgebaut werden kann.

Da Aggregation und Generalisation auf allen Stufen des Metamodells zur Verfügung stehen kann Geschäftsprozeßmodellen der Charakter der Generizität sowie der Kompositionalität verliehen werden. Das vereinfacht die Wiederverwendung von Modellteilen, vgl. dazu [Vl 96, S. 399 f.], weil vorhandene Prozeßteile leicht in neue Prozeßbeschreibungen eingebaut werden können und weil neue Prozeßteile aus vorhandenen leicht abgeleitet werden können. Anhaltspunkt für das Auffinden von gesuchten Prozeßteilen kann das Ziel sein, dessen Erreichung sie sichern. Über die Implementierung einer Umgebung zur Geschäftsprozeßmodellierung, die das Metamodell aus Abbildung 1 operationalisiert, berichtet [Ri 97].

Eine Kooperation ist eine Einheit sinnvollen kooperativen Arbeitsverhaltens der Akteure in einer Phase eines Geschäftsprozesses. Der betreffende Sinn ist die Erreichung eines Teilziels dieser Phase. Eine Kontribution ist der kleinste, nicht kooperative, vollständige Beitrag eines Akteurs zu einem Vorgang.

Die hier dargestellten Ideen sind schon in [K* 95] enthalten. In [L* 96] treten einige dieser Ideen in anderer Form auf. Diese Arbeit beruht auf der Idee wiederverwendbare Prozeßbausteine einzuführen und versucht mit deren Hilfe Wiederverwendung zu ermöglichen. Sie gibt weder eine Prozeßontologie noch ein Vorgehensmodell zum Herstellen bzw. Auswählen von Prozeßbausteinen an und beseitigt daher die grundlegenden Schwachstellen des imperativen Paradigmas nicht.

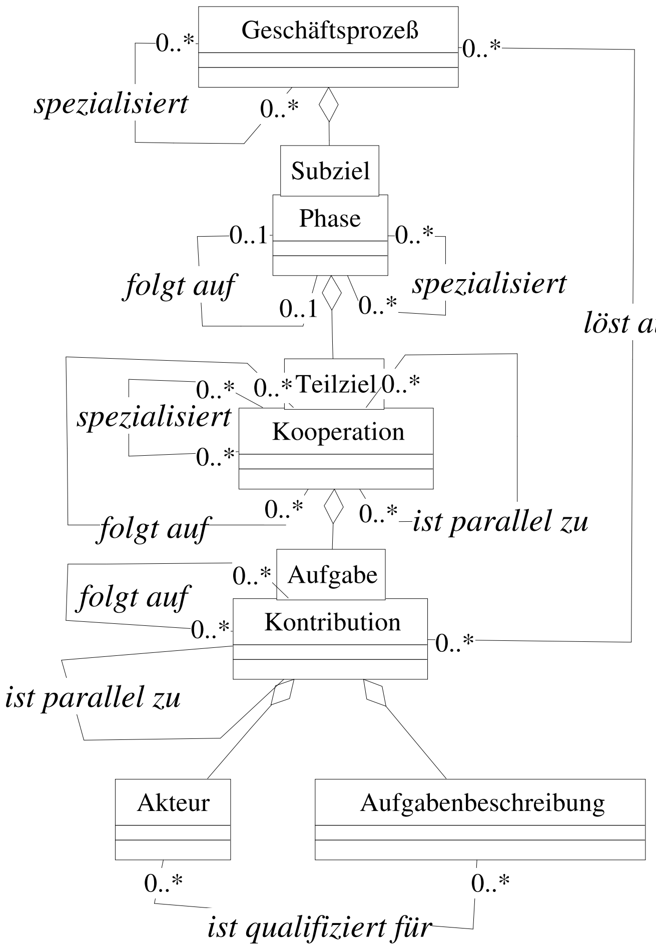

Abbildung 1: Kern eines Geschäftsprozeßmetamodells

7. Ein Vorgehensmodell für die Geschäftsprozeßmodellierung

Zunächst sind Ziele für Geschäftsprozesse festzulegen. Um ein Geschäftsprozeßmodell zu erarbeiten wird festgelegt wie dem Geschäftsprozeßziel Subziele so zugeordnet werden können, daß die sukzessive Erreichung der Subziele zur Zielerreichung führt. Jedes Subziel eines Ziels wird daraufhin in Teilziele zerlegt. Jedem Teilziel wird dann eine Kooperation zugeordnet und festgelegt wie die Kooperationen eines Subziels voneinander abhängen. Unter einem Teilziel wird hier ein Ziel verstanden, das gegebenenfalls gleichzeitig mit anderen Teilzielen, unabhängig von diesen, angestrebt wird um das Subziel zu erreichen. Schließlich wird jedes Teilziel so in Aufgaben zerlegt, daß deren Erfüllung die Erreichung des Teilziels bewirkt. Die Kooperation, die dem Teilziel zugeordnet ist, wird so in Kontributionen und Kooperationsregeln zerlegt.

In [KP 95] wird ein Vorgehensmodell zur Geschäftsprozeßmodellierung mit Hilfe von ereignisgesteuerten Prozeßketten angegeben. Zur Organisation von Arbeitssitzungen werden Angaben gemacht, darüber wie das betreffende Netz, d.h. das eigentliche Geschäftsprozeßmodell, herzustellen ist jedoch nicht. Damit klammert es den entscheidenden Teil eines Vorgehensmodells aus.

OGM, vgl. [Ro 96], ist ein objektorientierter Ansatz, dessen Metamodell Prozeßzustände und Prozeßschritte aufweist. Sein Vorgehensmodell ist dem hiesigen ähnlich. Es orientiert sich stark an konkreten Anhaltspunkten zur Identifikation von Kontributionen, ohne diesen Begriff zu kennen. Es verwendet den im Metamodell fehlenden Begriff Subprozeß, was hier als Schwäche des Metamodells gewertet wird. Da OGM einen Begriff wie

Kooperation nicht kennt und Regeln zur Identifikation von Prozeßschritten fehlen ist seine Unterstützung zur Herstellung qualitativ hochwertiger Modelle sowie zur Wiederverwendung von Prozeßteilen nicht optimal[3].

8. Resümee und Ausblick

Diese Arbeit behandelte das Problem, daß imperatives Paradigma und flache Prozeßontologie Modellierung und Wiederverwendung erschweren. Die betreffende Vorgangsontologie wurde angereichert und der Vererbungsbegriff für die Vorgangsmodellierung nutzbar gemacht. Dann wurden ein Geschäftsprozeßmetamodell und ein dazu passendes Vorgehensmodell skizziert.

Gegenwärtig wird eine Softwareumgebung zur Modellierung von Geschäftsprozessen implementiert. Diese bietet, die im obigen Metamodell dargestellten, Klassen als grundlegende Modellierungsbegriffe an und erlaubt die Simulation von Geschäftsprozessen sowie deren Analyse nach elementaren ökonomischen Aspekten. In Zukunft soll dies mit Hilfe eines professionellen Simulationswerkzeuges geschehen. Rechnergestützt soll aus einem Vorgangsmodell ein Simulationsmodell abgeleitet und dann untersucht werden.

9. Literatur

[Bl 92] Blum B.; Software Engineering a Holistic View; Oxford University Press; New-York, Oxford; 1992.

[BM 97] Braha D., Maimon O.; The Design Process: Properties, Paradigms, and Structure, IEEE Transactions on Systems, Man, and Cybernetics-Part A 27,2(1997):146-166.

[B* 92] Batini C., Ceri S., Navathe S.; Conceptual Database Design; The Benjamin/Cummings Publishing Company, Inc.; Redwood City, California et al; 1992.

[EW 96] Ernst M., Wolff M.-R.; Ein Vorschlag zum Einsatz von Wiederverwendungskonzepten im Workflow Management; HMD 188(1996):116-124.

[FS 95] Ferstl O., Sinz E.; Der Ansatz des Semantischen Objektmodells (SOM) zur Modellierung von Geschäftsprozessen; Wirtschaftsinformatik 37(1995):209-220.

[Gr 96] Gruhn V.; Geschäftsprozeß-Management als Grundlage der Software-Entwicklung. Informatik Forschung und Entwicklung 11,2(1996):94-101.

[Hu 96] Humphrey W.; Using a Defined and Measured Personal Software Process; IEEE Software 13,3(1996):77-90.

[J* 97] Jablonski S., Böhm M., Schulze W. (Hrsg.); Workflow-Management, Entwicklung von Anwendungen und Systemen; dpunkt Verlag; Heidelberg; 1997.

[Ku 95] Kuo Y.; When to Inherit And When Not; International Journal of Software Engineering and Knowledge Engineering 5,3(1995):391-405.

[KP 95] Keller G., Popp K.; Referenzmodelle für Geschäftsprozesse; HMD 187(1995):94-117.

[KP 97a] Kaschek R., Paech B.; Grundlagen der Modellierung von Vorgängen; [J* 97, S. 33-46].

[KP 97b] Kaschek R., Paech B.; Metamodellierung von Arbeitsabläufen; [J* 97, S. 47-64].

[K* 95] Kaschek R., Kohl C., Mayr H.C.; Cooperations-An Abstraction Concept Suitable For Business Process Reengineering; In Györkös J., Krisper M., Mayr H.C. (Hrsg.); Re-Technologies for Information Systems; R. Oldenbourg Verlag; Wien, München; 1995; S. 161-172.

[LM 78] Lockemann P., Mayr H.; Rechnergestützte Informationssysteme; Springer Verlag; Berlin, Heidelberg. 1978.

[L* 96] Lang K., Taumann W., Bodendorf F.; Business Process Reengineering with Reusable Reference Process Building Blocks; [SS 96, S. 265-290].

[PM 88] Peckham J., Marjanski F.; Semantic Data Models; ACM Computing Surveys 20,3(1988):153-189.

[Ro 96] Rohloff M.; An Object Oriented Approach to Business Process Modelling; [SS 96, S. 251-264].

[Ri 97] Rinderer T.; CoCA- Ein objektorientierter Ansatz und eine Umgebung zur Modellierung von Arbeitsvorgängen; Klagenfurt; 1997.

[R* 91] Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorensen W.; Object-Oriented Modeling And Design; Prentice-Hall International, Inc.; Englewood Cliffs; New Jersey; 1991.

[SS 96] Scholz-Reiter B., Stickel E. (Hrsg.); Business Process Modelling; Springer Verlag; Berlin et al.; 1996.

[Vl 96] Vliet van H.; Software Engineering, Principles and Practice; John Wiley & Sons, Chichester et al.; dritter Nachdruck; 1996.

[3] Einen Versuch Qualitätsfaktoren von Vorgangsmodellen zu bestimmen findet man in [KP 97a, S.42, 43].