Objektorientierte Analyse und konzeptueller Vorentwurf

Christian Kop, Heinrich C. Mayr
Institut für Informatik, Universität Klagenfurt
Universitätsstraße 65 - 67; A-9020 Klagenfurt
e-mail (chris | mayr)@ifi.uni-klu.ac.at

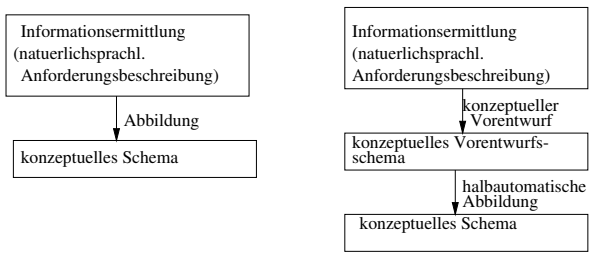
Zusammenfassung

Trotz zunehmender softwaretechnischer Qualität rechnergestützter Informationssysteme werden diese noch immer häufig den Bedürfnissen ihrer Anwender nicht gerecht. Ein Ansatz, einige der Ursachen hierfür zu vemeiden, besteht darin, im Entwurfsablauf dem konzeptuellen Entwurf bzw. der objektorientierten Analyse (OOA) eine Phase (wir nennen sie konzeptuellen Vorentwurf) voranzustellen, in der Systemanforderungen in einer den jeweiligen Bedarfsstellern gerechten und (unmiß-)verständlichen Form beschrieben werden. Die vorliegende Arbeit stellt ein hierfür geeignetes Modell in groben Zügen vor und beschreibt die Schritte, mit denen ein Vorentwurfschema in ein konzeptuelles Schema zu überführen ist. Als Modell der konzeptuellen Ebene wird dasjenige der Object Modelling Technique (OMT) herangezogen.

1. Einleitung

Dank des Fortschritts im Bereich der Methoden und Werkzeuge sowie der Prozessgestaltungsansätze wächst die softwaretechnische Qualität rechnergestützter Informationssysteme. Es werden zunehmend stabile Systeme produziert, die aber dennoch den Bedürfnissen ihrer Anwender häufig nicht gerecht werden: sie 'tun' das, was sie tun richtig, aber sie tun nicht, was gebraucht wird. Dies liegt daran, daß die Kommunikation zwischen Bedarfssteller und Systemanalytiker bzw. Systementwickler nach wie vor unvollständig und mit Mißverständnissen behaftet ist: Natürlichsprachliche Pflichtenhefte werden unterschiedlich interpretiert, konzeptuelle Entwürfe sind aufgrund mangelnder semantischer Mächtigkeit der verwendeten Modelle meist unvollständig und für den Bedarfssteller (Endanwender) meist auch zu abstrakt und in ihrer graphischen Darstellung zu komplex. Dem Problem der Unvollständigkeit konzeptueller Entwürfe (dazu gehören natürlich auch Ergebnisse objektorientierter Analysen) kann man dadurch beikommen, daß man eine vorliegende natürlich-sprachliche Spezifikation auf Basis eines Regelwerks analysiert und schrittweise auf konzeptuelle Elemente wie Klassen und Assoziationen abbildet. Dabei werden die nicht abbildbaren Anteile erkannt. Vorschläge für derartige Abbildungen finden sich beispielsweise in [Ch83, Ei84, KWDB86, PR86, SM94, Sy95]. Unvollständigkeiten der natürlichsprachlichen Spezifikation können damit natürlich kaum erkannt werden, wie auch deren unterschiedliche Interpretation, da sich das entstandene

Abbildung 1



konzeptuelle Schema in der Regel der Beurteilung durch den Bedarfssteller entzieht.

Ein Ansatz, dieses Problem zu überwinden, besteht darin, als erste Formalisierungsebene im Entwurf eine weniger abstrakte Sprachebene einzuziehen, mit Repräsentationskonzepten, die dem Erfahrungshorizont des betreffenden Anwenderkreises nahekommt. Für den Bereich betrieblich/administrativer Informationssysteme ist beispielsweise eine tabellarische Darstellung von Informationszusammenhängen geeignet. Eine hierauf abzielende Methode wurde beispielsweise im Rahmen des DATAID-Projektes [Ce83] entwickelt. Der in der vorliegenden Arbeit vorgestellte Ansatz baut hierauf auf, indem er die dort eingeführten Modellierungsbegriffe ergänzt, teilweise verallgemeinert bzw. verfeinert. Da es sich bei dieser Vorgehensweise letztlich wieder um einen konzeptuellen Entwurf handelt (es werden zur Modellierung Begriffe mit a priori Semantik verwendet), nennen wir sie konzeptuellen Vorentwurf. Natürlichsprachliche Beschreibungen werden demnach nicht direkt auf ein konzeptuelles Schema sondern zunächst auf ein Vorentwurfsschema abgebildet (siehe Abbildung 1).

Wir konzentrieren uns im folgenden auf den Schritt der Überleitung des Vorentwurfschemas in ein konzeptuelles Schema. Als Zielmodell wird dabei die Begriffswelt der konzeptuellen Ebene, also des OOA-Anteils der Object Modelling Technique (OMT [RBPE93, Ru95]) verwendet. Dazu beschreiben wir in Abschnitt 2 zunächst kurz unser Modell (Begriffsystem) für den konzeptuellen Vorentwurf, wobei wir uns hier auf die statischen Aspekte beschränken. Ein erster Vorschlag zur Dynamikmodellierung findet sich in [MS93]. Abschnitt 3 stellt das Vorgehensmodell für die Überleitung auf die konzeptuelle Ebene vor. Ein Ausblick findet sich in Abschnitt 4. Eine Einführung in die Object Modelling Technique würde den Rahmen dieses Papiers sprengen, der Leser sei daher auf die o.g. Literatur verwiesen.

2. Das Modell des konzeptuellen Vorentwurfs

Das Modell des Vorentwurfs muß mehreren Anforderungen genügen: Es muß für Bedarfssteller verständlich sein, damit sie Entwurfsergebnisse beurteilen und ggf. Änderungen durchführen können. Es soll dem Entwerfer bereits eine halbformalisierte Darstellung des Diskursbereichs ermöglichen und ihn bei der Klassifizierung der verschiedenen Anforderungen unterstützen. Unser Ziel ist es außerdem, eine brauchbare Zwischenebene ('Interlingua' in der Begriffswelt der Sprachwissenschaften) für die (halb-)automatische Analyse und Abbildung natürlichsprachlicher Anforderungsspezifikationen auf konzeptuelle Schemata zu schaffen. Im Bereich betrieblich/administrativer Aufgabenstellungen sind unserer Meinung nach verzeichnisorientierte Modelle für diesen Zweck geeignet: Die Bedarfssteller werden hier in ihrer täglichen Arbeit ständig mit Verzeichnissen - wenn auch in einem anderen Kontext - konfrontiert. Verzeichnisse lassen die Unterteilung von Informationen und Anforderungen in verschiedene semantisch vorbelegte Spalten zu und unterstützen damit die Klassifizierung und Formalisierung des Diskursbereichs durch den Systemanalytiker.

Die wichtigsten Konzepte (Begriffe) des erweiterten Verzeichnisansatzes sind **Ding**, **Zusammenhang**, **Aspekt**, **Operation**, **Einschränkung** und **Ereignis**, wobei hier jeweils die Typebene und nicht die Extensionsebene angesprochen wird. Wir beschränken uns hier auf eine nähere Betrachtung derjenigen Begriffe, die der Beschreibung statischer Zusammenhänge dienen:

Dinge können sein

- natürliche oder juristische Personen (*Kunde*, *Mitarbeiter*, ...),
- materielle oder immaterielle Gegenstände (*Produkt*, *Fahrzeug*, *Vertrag* ..),
- abstrakte Begriffe (*Vertragsklausel*).

Diese Beispiele lassen keinen besonderen Unterschied zu den auf konzeptueller Ebene gebräuchlichen Begriffen wie "Entitytyp" [Ch76] oder "Klasse" [RBPE93] erkennen. Letztere verlangen jedoch eine Charakterisierung durch weitere Merkmale (Attribute), z.B. *Person* durch das Merkmal *Name*. Entitytypen müssen darüberhinaus durch ein oder mehrere Attribute explizit identifizierbar sein. Diese Restriktionen gelten nicht für den Begriff des Dings. Eine *Kundennummer*, ein *Name* oder eine *Telefonnummer* werden ebenso als Dinge aufgefaßt. Dies hat den Vorteil daß eine Unterscheidung nach Entitytypen bzw. Klassen und Attributen nicht sofort getroffen werden muß (bei der Analyse eines einzelnen Anforderungssatzes). Sie erfolgt erst, nachdem die Informationen im Laufe des konzeptuellen Vorentwurfs vollständig gesammelt und aufbereitet wurden und somit der Informationsgehalt mehrerer Sätze miteinander verglichen werden kann.

Ebenso wie auf der konzeptuellen Ebene wird auch hier versucht, Verbindungen zwischen Dingen zu finden. Eine solche Verbindung wird im Unterschied zum konzeptuellen Modell nicht Assoziation sondern **Zusammenhang** genannt. Ein Zusammenhang ist das Aggregat verschiedener Blickwinkel (**Aspekte**) bezogen auf das zuerst genannte beteiligte Ding (vgl. dazu den Begriff der Rollen in manchen konzeptuellen Modellen):

- *Ein **Kunde kauft** mehrere Produkte.*
- *Ein **Produkt kann** von einem Kunden **gekauft werden**.*

Da wir wiederum nicht Klassen sondern Dinge in Zusammenhang stellen, können auch

- *Ein **Kunde hat** mehrere Telefonnummern bzw.*
- *Eine **Telefonnummer gehört** einem Kunden.*

als Aspekte eines Zusammenhang betrachtet werden. Daher auch die abweichende Namensgebung, um Verwechslungen mit dem Begriff der Assoziation des konzeptuellen Modells zu vermeiden. Ein ähnlicher begrifflicher Ansatz, allerdings mit einer anderen Terminologie und Notation, wird in [NiHa89] vorgeschlagen.

Neben diesen allgemeinen Zusammenhängen können auch spezielle Zusammenhänge (Generalisation, Aggregation) beschrieben werden.

Die bisher genannten Begriffe werden mittels zweier Verzeichnisse (**Ding**- und **Zusammenhangsverzeichnis**) gehandhabt. Beide Verzeichnisse sind wiederum durch Spalten untergliedert, welche den Eintrag zusätzlicher Informationen zu Dingen, Zusammenhängen und Aspekten gestattet. Zusätzlich gibt es auch noch ein Operations- und ein Ereignisverzeichnis zur Aufnahme funktionaler und dynamischer Miniweltanforderungen [MS93], ein Einschränkungsverzeichnis in welchem Informationen aufgenommen werden, die keinem der anderen Verzeichnisse zuordenbar sind und ein Satzverzeichnis, damit man aus den Informationen in den anderen Verzeichnisse jederzeit die ursprünglichen natürlichsprachlichen Anforderungen herleiten kann.

3. Konsistenzprüfung und Transformation

In [Ko93] wurde ein Verfahren vorgeschlagen, das Dinge auf Begriffe einer spezifischen Erweiterung des ER-Modells (EER+Modell [Ku92]) abzubilden versucht. In der vorliegenden Arbeit soll das Verfahren herangezogen werden, um Dinge auf Begriffe der Object Modelling Technique (OMT) [RBPE93, Ru95] (namentlich: Klasse und Attribut) abzubilden.

Im folgenden beschränken wir uns auf die Abbildung von statischen Elementen des konzeptuellen Vorentwurfs (Dinge, Zusammenhänge, Aspekte). Diese Abbildung macht nur einen Teil des gesamten Vorgehensmodells für die Entwicklung eines Statikschemas aus. Der Abbildung vorgelagert ist neben der Anforderungssammlung auch ein Initialisierungsschritt, in dem bekannte Informationen aus vorhergehenden Entwürfen in die neuen Verzeichnisse übernommen werden. Damit wird die Wiederverwendung früherer Entwurfsergebnisse unterstützt ('domain knowledge reuse'). Eine detaillierte Beschreibung dieses Schrittes findet sich in [Ko93] und in [KM94].

Die Abbildung erfolgt in den Schritten **Konsistenzprüfung**, **Begriffszuordnung**, **Restrukturierung** und **Vervollständigung**.

Konsistenzprüfung

Bevor eine Transformation in ein konzeptuelles Schema erfolgen kann, muß zunächst einmal festgestellt werden, ob die Informationen in den Verzeichnissen in sich stimmig sind oder ob möglicherweise wesentliche Informationen fehlen. Hierzu zählen unter anderem:

- Überprüfung auf Zyklenfreiheit bei Generalisationen,
- Erkennung synonymer Zusammenhangsbeschreibungen anhand derselben beteiligten Dinge,
- Vervollständigung der Zusammenhänge um Kardinalitäten und Aspektnamen.

Diese Art von Prüfung kann allerdings höchstens halbautomatisch durchgeführt werden, da der Ausgang solcher Überprüfungen vom Informationskontext abhängt. Die Entscheidung ob Informationen konsistent sind, muß in derartigen Fällen nachwievor der Entwerfer treffen. Betrachten wir dazu als Beispiel die Erkennung synonymer Zusammenhangsbeschreibungen: Zwei durch die Sätze "*Ein Kunde **hat** eine Telefonnummer.*" und "*Ein Kunde **besitzt** eine Telefonnummer.*" beschriebene Zusammenhänge können zu einem Zusammenhang zusammengeführt werden, da sie dieselbe Information betreffen. Andererseits können zwei Zusammenhänge die aus den beiden Sätzen: "Ein Mitarbeiter **arbeitet** an einem Projekt." und "Ein Mitarbeiter **leitet** ein Projekt." nicht zusammengeführt werden, da sie unterschiedliche Bedeutung haben.

Konsistenzprüfungen wie z.B. *Jedes Ding, welches in einem Verzeichnis (Zusammenhangsverzeichnis, Operationsverzeichnis, Einschränkungsverzeichnis) referenziert wird, muß auch im Dingverzeichnis aufscheinen* (vgl. [Ce83]) können von einem geeigneten Werkzeug automatisch, also bereits bei der Eingabe der Information, durchgeführt werden.

Begriffszuordnung

Wie der Name schon andeutet, wird hier versucht, den Dingbegriff durch Begriffe des konzeptuellen Schemas zu ersetzen. Vereinfacht läßt sich der Schritt der Begriffszuordnung

wie folgt beschreiben: Analysiere die im Zusammenhangsverzeichnis verknüpften Dinge anhand der syntaktischen Definitionen des konzeptuellen Modells und versuche daraus Rückschlüsse auf die mögliche Klassifizierung der Dinge nach Klassen und Attributen zu ziehen. Beispielsweise wird man Dinge, die mit sich selbst in Zusammenhang stehen wie etwa im Fall "*einem Mitarbeiter unterstehen mehrere andere Mitarbeiter / ein Mitarbeiter wird von einem anderen Mitarbeiter geleitet*", auf Klassen abbilden.

Um die Begriffszuordnung durchführen zu können, wurden einige Entscheidungsgrundlagen ermittelt, die von einem Werkzeug evaluiert werden können. Sie können nach der Art der Begriffsermittlung in **direkte** und **indirekte** Entscheidungsgrundlagen, nach der Aussagekraft in **Regeln** und **Vorschläge** und schließlich nach dem Zuordnungsergebnis in **Entscheidungsgrundlagen für Klassen** und **Entscheidungsgrundlagen für Attribute** unterteilt werden.

Bei den direkten Entscheidungsgrundlagen wird unmittelbar aus den Informationen im Zusammenhangsverzeichnis eine Begriffszuordnung durchgeführt. Bei indirekten Entscheidungsgrundlagen werden zu Informationen im Zusammenhangsverzeichnis zusätzlich auch bereits begrifflich erkannte Dinge als Information zur Entscheidungsfindung herangezogen. Regeln sind Entscheidungsgrundlagen, die zu einer strikten Zuordnung zu einem konzeptuellen Begriff führen. Insbesondere sind es Entscheidungsgrundlagen, deren Nichteinhaltung ein unkorrektes konzeptuelles Schema ergeben würde, da sie beispielsweise eine syntaktische Modelldefinition verletzen würde. Vorschläge hingegen sind Entscheidungsgrundlagen, die eine bestimmte Begriffszuordnung präferieren, deren Einhaltung jedoch nicht erzwingen.

Schließlich gibt es noch sogenannte **Meta-Regeln**. Diese führen zu keiner Begriffszuordnung, ihre Verwendung stellt aber ebenfalls syntaktisch korrekte Modelldefinitionen sicher.

Ein Beispiel für eine **Regel** ergibt sich aus dem bereits oben erwähnten Fall:

> *Ein Ding ist als Klasse zu modellieren, wenn es mit sich selbst in Zusammenhang steht.*

Diese Regel leitet sich daraus ab, daß Attribute in OMT nicht mit sich selbst (also rekursiv) in Beziehung stehen können. Demnach würde in diesem Fall die Klassifikation des Dings als Attribut zu einem Modellierungsfehler auf der Ebene des konzeptuellen Modells führen.

Ein Beispiel für einen **Vorschlag** ist:

> *Ein Ding dessen Dingname mit Bezeichnungen wie "name", "typ", "art" ... endet, könnte ein Attribut sein.*

Da Attribute zumeist durch oben genannte Bezeichnungen charakterisiert sind, ist dies ein guter Hinweis darauf, daß das Ding als Attribut modelliert werden sollte. Es könnte aber auch als Klasse beschrieben werden. Deshalb wird hier nur ein Vorschlag als Entscheidungsgrundlage angewandt.

Ein Beispiel für eine **indirekte Regel** ist:

> *Wurde in einem binären Zusammenhang eines der beiden beteiligten Dinge als Attribut*

erkannt, so ist das zweite beteiligte Ding als Klasse zu modellieren.

Bei dieser indirekten Regel spielt ebenfalls die Überlegung eine Rolle, daß Attribute nicht miteinander in Beziehung stehen.

Ein Beispiel für eine **Metaregel** ist:

In einem mehrstelligen Zusammenhang muß mindestens zwei der beteiligten Dinge der Begriff Klasse zugeordnet werden.

Auch hier wird davon ausgegangen, daß eine Klasse und zwei Attribute in einem mehrstelligen Zusammenhang wenig Sinn machen. Wir haben die Option offengelassen, daß ein mehrstelliger Zusammenhang zwei Klassen und ein oder mehrere Attribute haben kann. Eine solche Konstruktion ist zwar noch keine reguläre Konstruktion in OMT, man kann sie aber während des Restrukturierungsschrittes in eine solche umwandeln.

Tabelle 1

Klasse	Attribut	Begriffszuordung
Regel/Vorschlag	——	Klasse
___	Regel/Vorschlag	Attribut
Regel	Vorschlag	Klasse
Vorschlag	Regel	Attribut
Vorschlag	Vorschlag	Anzahlermittlung
Regel	Regel	Widerspruch

Mit Hilfe derartiger Entscheidungsgrundlagen werden nun für die Zuordnung Klasse bzw. Attribut einfache Gesetzmäßigkeiten definiert (siehe Tabelle 1).

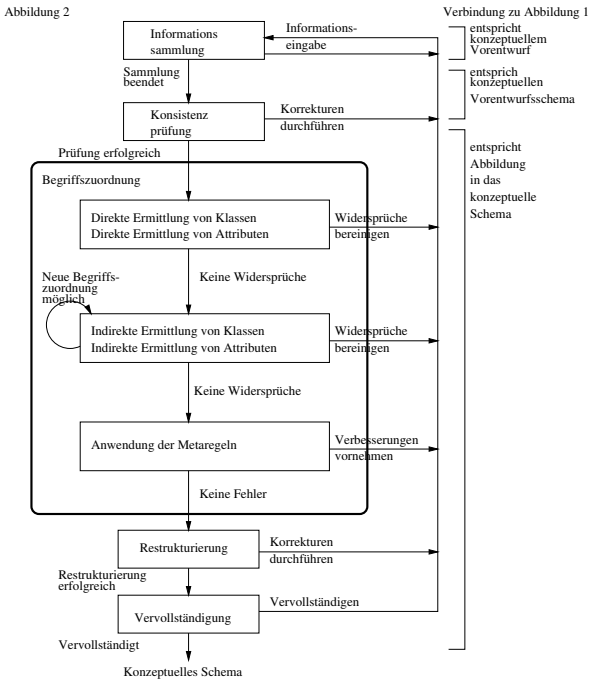
Die obige Tabelle, läßt sich folgendermaßen interpretieren: wenn immer auf eine Klasse bzw. ein Attribut zumindest eine Regel angewandt werden kann, erhält das Ding die entsprechende Zuordnung. Werden nur Vorschläge für eine bestimmte Begriffszuordnung gefunden, so wird jener Begriff präferiert, der mehr Vorschläge für sich verbuchen kann. Wird aus den Informationen im Ding und Zusammenhangsverzeichnis für beide Begriffe zumindest eine Regel gefunden, so entsteht ein Widerspruch. Dieser ist durch den Entwerfer durch Hinzfügen oder Entfernen von Informationen aufzulösen. Nachdem das Hinzufügen bzw. Entfernen von Informationen bedingt, daß die Verzeichnisse selbst wieder einer Konsistenzprüfung unterzogen werden müssen, wird der in Abbildung 2 veranschaulichte Ablauf für die Begriffszuordnung vorgeschlagen. Zuerst wird versucht Klassen und Attribute direkt zu ermitteln. Enthalten die Verzeichnisse keine widersprüchlichen Informationen ("Keine Widersprüche" in Abb. 2) erfolgt die indirekte Ermittlung und nach deren erfolgreichem Abschluß die Anwendung von Metaregeln.

Restrukturierung

In der Restrukturierung erfolgt eine weitere Überprüfung der Verzeichnissinhalte unter Berücksichtigung des konzeptuellen Modells. Überprüft werden kann z.B., ob Klassen verwaist sind, d.h. weder Attribute noch Beziehungen (Assoziationen, Generalisationen u.ä.) zu anderen Klassen besitzen. Unterschiedliche Klassen können daraufhin untersucht werden, ob sie ein oder mehrere Attribute gemeinsam besitzen.

Ebenso kann es zur Umordnung von Verzeichnisinhalten kommen. Umstrukturierungen können z.B. erfolgen, wenn eine Klasse eine generalisierte Klasse mehrerer Subklassen ist und diese Subklassen ein oder mehrere gemeinsame Attribute besitzen. Diese Attribute könnten dann der

generalisierten Klasse zugeordnet werden. Eine andere Umstrukturierung wurde bereits in der Begriffszuordnung angesprochen (Attribute in mehrstelligen Zusammenhängen). Solche Konstruktionen könnten in OMT gelöst werden, indem die Assoziation selbst zu einer Klasse wird (vgl. [RBPE93] S. 39ff). Die Attribute, die vormals dem mehrstelligen Zusammenhang angehörten, würden dann zu Attributen der neuen Klasse.



Nachdem bereits im vorangegangenen Schritt Dingen Begriffe des konzeptuellen Modells zugeordnet werden konnten, erfolgt mit diesem Wissen schließlich während der Restrukturierung auch die Abbildung des Zusammenhangs auf einen Begriff des konzeptuellen Modells. Zusammenhänge zwischen erkannten Klassen werden zu Assoziationen. Zusammenhänge zwischen einer Klasse und einem Attribut dienen als Information, welches Attribut zu welcher Klasse gehört.

Vervollständigung

Im Vervollständigungschritt werden die Verzeichnisse um Informationen angereichert, die nicht aus der Analyse der natürlichsprachlichen Anforderungen gewonnen werden können, die aber dennoch eine wichtige Rolle im konzeptuellen Schema einnehmen. Informationen wie z.B. die Erweiterung der Attribute um die Information ihrer Sichtbarkeit, die Parameterbelegung von Klassen sowie das Zusammenfassen von Schemata in Subsysteme nach OMT (siehe [Ru95]) zählen zu den Tätigkeiten in diesem Schritt. Der Entwerfer hat diese Information manuell nachzutragen.

4. Ausblick

In dieser Arbeit wurde ein Modell vorgestellt, welches dem Entwerfer eine formalisierte Darstellung und Klassifizierung der statischen Aspekte einer Diskurswelt gestattet. Das Modell kann ihm gleichsam als Notizblock zur Ordnung der Anforderungen dienen, das auch von den Bedarfsstellern und Endbenutzern validiert werden kann. Weiterhin wurde ein Verfahren vorgestellt, wie Informationen aus diesem Schema in ein konzeptuelles Schema überführt und widersprüchliche Anforderungen entdeckt werden können. Erste Erfahrungen aus einem Anforderungsanalyseprojekt, bei dem Verzeichnisse mit mehreren hundert Dingen und Zusammenhängen in einem Kärntner Unternehmen erstellt wurden, haben gezeigt, daß unser Modell für den konzeptuellen Vorentwurf (nach einigen kleineren Modifikationen) von Endanwendern mit unterschiedlichsten Tätigkeitsbereichen gut aufgenommen und verstanden wurde.

Die nächsten Arbeitsschwerpunkte zielen auf eine Verfeinerung des Regelwerks zur Abbildung in das OMT-Modell unter Einschluß der dynamischen Aspekte, eine formale Absicherung des Verzeichnisansatzes, sowie eine verfeinerte Anbindung des Konzepts für die grammatikalische

Analyse natürlichsprachlicher Anforderungen ab.

Zu letzterem wurden in [Fe92] bereits erste Ideen und Vorgehensweisen entwickelt. In diesem Zusammenhang verspricht ein linguistischer Ansatz, der eine NTS Grammatik [FMa91, FM93] zur Analyse von natürlichsprachlichen Sätzen heranzieht, gute Ergebnisse zu erzielen.

Literaturverzeichnis

[Ce83] Ceri, S. (ed): Methodology and Tools for Database Design. North Holland, 1983.

[Ch76] Chen, P.P.: The Entity Relationship Model - Towards a Unified View of Data. ACM TODS Vol.1, No. 1, 1976 pp. 9-36.

[Ch83] Chen, P.P.: English Sentence Structure and Entity Relationship Diagrams. In: Int. Journal of Information Siences, Vol 29, 1983, pp 127-149.

[Ei84] Eick, C. F.: From natural language to good database definitions - a database design methodology. In: Proceedings of the International Conference on Data Engineering, Apr. 1984, pp 324-331.

[Fe92] Felderer A.: Zur Tabellarisierung natürlichsprachlicher Anforderungs-beschreibungen. Diplomarbeit, Institut für Informatik, Universität Klagenfurt, 1992.

[FMa91] Fliedl, G.; Maratschinger, M.: Natürlichkeitstheoretische Syntax. Ein Modell und seine Anwendungen. In: Betriebslinguistik und Linguistikbetrieb, Band 1. Tübingen Niemeyer.

[FM93] Fliedl, G.; Mayerthaler, W.: Natürlichkeitstheoretische Syntax. In: Jakobs, J.; v. Stechnow, A.; Sternefeld, W.; Vennemann, Th. (Hrsg.): Syntax. Ein internationales Handbuch zeitgenössischer Forschung. De Gruyter, Berlin, pp. 610-635.

[KM94] Kop, C.: Mayr, H.C.: Reusing Domain Knowledge in Requirements Analysis. In (Györkös, J. et. al. eds): Proc. Re'94, Bled, 1994, pp. 133-147.

[Ko93] Kop, C.: Herleitung von EERM+-Schemata aus Ding-, Zusammenhangs- und Operationsverzeichnisssen. Diplomarbeit, Insittut für Informatik, Universität Klagenfurt, 1993.

[Ku92] Kupper T.: Relationale Verwaltung von EER+-Schemata und deren Überleitung in relationale Datenbankschemata. Diplomarbeit, Institut für Informatik, Universität Klagenfurt, 1992.

[KWDB86] Kersten M. L; Weigand H.; Dignum F.; Doom J.: A conceptual modelling expert system. In: [Sp86], pp 35-48.

[MeSe86] Meersman R.; Sernadas A.C. (eds.): Data and Knowledge (DS2): Proceedings of the 2nd IFIP 2.6. Working Conference on Database Semantics, North Holland Publ. Comp. Nov. 1986.

[MS93] Mayr, H.C.; Schnattler, M.: Vorkonzeptuelle und konzeptuelle Dynamik Modellierung. In [SR93], pp. 3-18.

[NiHa89] Nijssen G. M.; Halpin T. A.: Conceptual Schema and Relational Database Design - A fact oriented approach. Prentice Hall Publ. Comp., 1989.

[PR86] Proix, C.; Rolland, C.: A knowledge base for information systems design. In: [MeSe86], pp 293-307.

[RBPE93] Rumbaugh, J.: Blaha, M.: Premerlani, W.: Eddy, F.; Lorensen, W.: Objektorientiertes Modellieren und Entwerfen. Carl Hanser Verlag, München, Wien, 1993.

[Ru95] Rumbaugh J.: OMT: The object model. In: Journal of objectoriented programming. Jan. 1995.

[SM94] Sunil, V.; Farid M.: From English to Formal Specifications. In.: The Computer Journal. Vol. 37, No. 9. 1994.

[Sp86] Spaccapietra S.(ed): Proceedings of the 5th International Conference on Entity-Relationship Approach. North Holland, Nov. 1986.

[SR93] Scheschonk, G.; Reisig, W. (eds): Petri-Netze im Einsatz für Entwurf und Entwicklung von Informationssystemen. Springer Verlag 1993.

[Sy95] Sykes, J.A.: English Grammmar as a Sentence Model for Conceptual Modelling using NIAM. In: ISCO 3 International Conference on Information System Concepts, Marburg, March 1995.