

Arbeitsberichte des Instituts für Wirtschaftsinformatik

Herausgeber: Prof. Dr. J. Becker, Prof. Dr. H. L. Grob, Prof. Dr. S. Klein,
Prof. Dr. H. Kuchen, Prof. Dr. U. Müller-Funk, Prof. Dr. G. Vossen

Arbeitsbericht Nr. 69

Semantische Spezifikation Dispositiver Informationssysteme

Roland Holten

Institut für Wirtschaftsinformatik der Westfälischen Wilhelms-Universität Münster,
Steinfurter Str. 107, 48149 Münster, Tel. (0251) 83-38100, Fax (0251) 83-38109
Juli 1999

1	Semantische Modellierung Dispositiver Informationssysteme als Gegenstand	4
2	Ordnungsrahmen für die Information-Warehouse-Entwicklung	5
2.1	Dimension der Entwicklungsphasen	5
2.2	Dimension der Modellierungs- und Abstraktionsebenen	6
2.3	Kombination der Dimensionen: Ordnungsrahmen für Information-Warehouse-Entwicklungsprozesse	8
3	Sprache und Notation für die semantische Modellierung von Dispositiven Informationssystemen	9
3.1	Zum Verhältnis von Sprache und Notation	9
3.2	Metamodell zur semantischen Modellierung Dispositiver Informationssysteme – Explizierung der Sprachdefinition	11
3.3	Notation zur semantischen Modellierung Dispositiver Informationssysteme	17
4	Vorgehen bei der semantischen Modellierung Dispositiver Informationssysteme und Beispiel	25
4.1	Ermittlung von Dispositiven Aufgaben	25
4.2	Erstellung und Nutzung von Bibliotheken	26
4.3	Semantische Modellierung der Informationsobjekte	27
5	Ausblick	29
	Literatur	31

Zusammenfassung

Trotz verfügbarer Technologie und anerkannter Systemarchitektur existieren im Rahmen der Entwicklung von Dispositiven Informationssystemen ungelöste Probleme, die in der *semantischen Modellierung* zu sehen sind. Es mangelt an anerkannten Sprachen und Notationen, die die Voraussetzung zur Ausbildung anerkannter Entwicklungsmethoden darstellen.

Zunächst wird der Problembereich motiviert und es wird in Kapitel 2 ein Ordnungsrahmen vorgeschlagen, der hilft, die Diskussion von Entwicklungsprozessen von Information Warehouses zu strukturieren. Kapitel 3 schlägt sowohl eine Sprache als auch geeignete Notationskonventionen zur Nutzung der Sprache im Rahmen der semantischen Modellierung Dispositiver Informationssysteme vor. Kapitel 4 bietet ein Anwendungsbeispiel.

1 Semantische Modellierung Dispositiver Informationssysteme als Gegenstand

Data Warehouses stellen heute wesentliche Architekturkomponenten zur Entwicklung von Dispositiven Informationssystemen dar.¹ Nach übereinstimmender Meinung sind Daten aus operativen Transaktionssystemen über geeignete Werkzeuge zu extrahieren und für das Management in geeigneter Form im Data Warehouse über einen langen Zeitraum bereitzustellen. In einer weiteren Schicht befinden sich oberhalb des Data Warehouse OLAP-Systeme, die flexibel gewünschte Sichten auf den Datenbestand generieren können. Die oberste Schicht dieser Architektur bilden Reporting-Werkzeuge, die mittels der OLAP-Systeme Daten aus dem Data Warehouse laden und die gewünschten Sichten grafisch aufbereitet anzeigen. Auf dieser obersten Schicht werden auch Programme zur Tabellenkalkulation flexibel in der dargestellten Funktion genutzt. Diese gesamte Architektur soll hier als *Information Warehouse* bezeichnet werden. *Dispositive Informationssysteme* sind Systeme, die Entscheidungsträger im betriebswirtschaftlichen Sinne durch Bereitstellung von Informationen, die im Rahmen der Entscheidungsaufgabe als relevant angesehen werden, unterstützen sollen. Die Namen derartiger Systeme sind trotz der recht jungen Geschichte von Betriebswirtschaftslehre, Wirtschaftsinformatik und Informatik zahlreich.²

Dispositive Informationssysteme stellen grundsätzlich andere Anforderungen an die Datenbasis als Transaktionssysteme. Dies betrifft insbesondere die Bereitstellung integrierter Sichten sowie die Dokumentation von Entwicklungen im Zeitablauf. Das Information Warehouse im oben genannten Sinne ist in der Lage, die gestellten Anforderungen zu erfüllen. Entsprechende Komponenten zur Umsetzung der Architektur des Information Warehouse sind heute am Markt verfügbar. Nahezu alle großen Hersteller von Datenbankmanagementsystemen wie z. B. Oracle, Sybase, Informix, IBM oder Microsoft bieten entsprechende Werkzeuge an. Außerdem gibt es spezialisierte Anbieter wie MicroStrategy, die beispielsweise ausgereifte Lösungen für OLAP-Systeme auf der Basis Relationaler Datenbanksysteme anbieten. Cognos bietet Werkzeuge, die stattdessen proprietäre Filesysteme nutzen, um multidimensionale Strukturen zu implementieren.

¹ Vgl. beispielsweise Chaudhuri, Dayal (1997), S. 66 ff.; Becker, Holten (Wirtschaftsinformatik) (1998), S. 291 ff.; Holten (1999), S. 40-58; Mucksch, Behme (1998), S. 44-73; Chamoni, Gluchowski (1998), S. 10-21; Vossen (1999), S. 669 ff.

² Vgl. zu einem Überblick entsprechender Konzepte Holten (1999), S. 29 ff.; S. 76 ff.

Trotz der Verfügbarkeit der Technologie und der anerkannten Architektur ist in der *semantischen Modellierung* Dispositiver Informationssysteme ein Engpaß der Entwicklung zu sehen.³ Es sind hier noch keine Methoden etabliert. Ein wesentliches Problem stellt die Bereitstellung geeigneter Sprachen und Notationen für die semantische Modellierung Dispositiver Informationssysteme dar. Diesem Problembereich widmet sich die vorliegende Arbeit. Zunächst wird ein Ordnungsrahmen vorgeschlagen, der hilft, die Diskussion von Entwicklungsprozessen von Information Warehouses zu strukturieren. Darauf aufbauend wird das Verhältnis der Begriffe Sprache und Notation zueinander geklärt und es werden sowohl eine Sprache als auch geeignete Notationskonventionen zur Nutzung der Sprache im Rahmen der semantischen Modellierung Dispositiver Informationssysteme vorgeschlagen. Ein Beispiel rundet den Beitrag ab.

2 Ordnungsrahmen für die Information-Warehouse-Entwicklung

Zur klaren Abgrenzung und Einordnung der folgenden Überlegungen wird zunächst ein Ordnungsrahmen der Entwicklung von Information Warehouses vorgeschlagen, der das Verhältnis von Modellen und Modellierung anhand von akzeptierten Entwicklungsphasen und Abstraktionsebenen definiert.⁴

2.1 Dimension der Entwicklungsphasen

Es ist heute weitgehend akzeptiert, daß Entwicklungsprozesse mit der konzeptuellen Spezifikation dessen, was ein System leisten soll, dem „was“, zu beginnen haben.⁵ Diese fachkonzeptuelle Spezifikation fokussiert die Domäne, in der das System eingesetzt werden soll. Zur Spezifikation der domänenspezifischen Anforderungen werden folglich entsprechende Sprachstrukturen benötigt.

Als zweite Entwicklungsphase gilt das Programmieren im Großen, gelegentlich auch als Design oder DV-Konzeption bezeichnet.⁶ Im Design werden die Komponenten des Systems und ihre Interaktion spezifiziert, was zur Architektur des Systems führt. Typischerweise werden hier Entscheidungen getroffen, die das Datenbankschema oder das Interface des Systems betreffen.⁷ Die Architektur beschreibt die Funktion jeder Komponente und die Bezie-

³ Vgl. Gabriel, Gluchowski (1998), S. 493; Gabriel, Gluchowski (1997), S. 19; Becker, Holten (1998), S. 483 f.; Becker et al. (1994), S. 423; Lehmann, Ellerau (1997), S. 93; Holten (1999), S. 59 f.

⁴ Vgl. Holten (Framework) (1999).

⁵ Vgl. McMenamin, Palmer (1984); Davis (1990); IEEE-830 (1984).

⁶ Vgl. Balzert (1996a), S.631 ff.; Nagl (1990).

⁷ Vgl. Codd (1990); Balzert (1996b); Date (1990); Embley (1998); Gupta, Horowitz (1991).

hungen zwischen den Komponenten. Kern der Spezifikation des DV-Konzeptes ist somit die Trennung der Schnittstellen der Komponenten von ihrer Implementierung.

Als dritte Phase wird die Implementierungsphase gesehen. Sie behandelt die Realisierung der Komponenten und umfaßt im wesentlichen Aufgaben der Programmierung, der Entwicklung von Algorithmen und Datenstrukturen sowie der entsprechenden Dokumente.

Alle diese Phasen sind selbstverständlich in einen evolutionären Prozeß, der durch vielfältige Rücksprünge gekennzeichnet ist, eingebunden. Es soll hier lediglich die *logische Gruppierung* von Entwicklungsaufgaben betrachtet werden, die zur Bildung der Phasen führt. Die häufig genannte Phase der Wartung bedeutet unter Zugrundelegung des soeben definierten Phasenverständnisses dann nichts anderes als einen Rücksprung in (irgend)eine Entwicklungsphase.⁸ Entwicklungsprozesse werden hier demnach vereinfacht als die logische Abfolge der Phasen *Fachkonzeptuelle Spezifikation, Design (DV-Konzept) oder Programmierung im Großen und Implementierung* betrachtet. Eine Phase stellt in diesem Sinne die Gruppierung von Entwicklungsaufgaben anhand von Entwicklungsentscheidungen dar.

2.2 Dimension der Modellierungs- und Abstraktionsebenen

Jede der bisher skizzierten Phasen produziert gewisse Dokumente als Output und benötigt gewisse Dokumente als Input. Die zweite Dimension des Ordnungsrahmens spezifiziert verschiedene logische Abstraktionsebenen, die das Verhältnis der Dokumente und Modelle, die im Rahmen der Entwicklung anfallen, sowie ihr Verhältnis zur Beschreibung des Entwicklungsprozesses selbst zum Gegenstand haben.⁹ In der Logik werden verschiedenen Sprachebenen unterschieden. Ist eine Sprache Gegenstand einer wissenschaftlichen Untersuchung, wird sie als *Objektsprache* bezeichnet. Demgegenüber wird die Sprache, in der die Untersuchung durchgeführt wird, als *Metasprache* bezeichnet.¹⁰ Das Attribut *meta* beschreibt in diesem Sinne die *Rolle*, die einer Sprache in einem wissenschaftlichen Prozeß zukommt. Diese Überlegungen lassen sich auf den Prozeß der Modellbildung übertragen.¹¹ Sprachen werden bei der Entwicklung von Modellen herangezogen. Eine Sprache S1, die zur Erstellung eines Modells M1 herangezogen wird, kann selbst in einem Modell M2 bezüglich ihrer Syntax beschrieben sein. Gemäß den obigen Überlegungen stellt M2 ein Metamodell bezogen auf den Gegenstand

⁸ Vgl. Nagl (1990), S. 17-23.

⁹ Vgl. Pohl (1996), S.78-90; ISO/IEC 10027; ISO/IEC (1990); Jacobs, Holten (1995), S. 98 ff.; Holten (1999), S. 10-17

¹⁰ Vgl. Kambartel (1996); Lorenz (Metasprache) (1995); Lorenz (Objektsprache) (1995).

¹¹ Vgl. Holten (1999), S. 11 ff.

des Modells M1 dar (Abbildung 1).¹² Dieselbe Idee der Unterscheidung verschiedener Abstraktionsebenen, die zu mehreren Typ-Instanz-Beziehungen führt, ist im ISO/IEC IRDS Framework angewendet. In diesem Framework führen die Beziehungen zu verbundenen Paaren von Abstraktionsebenen (Abbildung 2).¹³

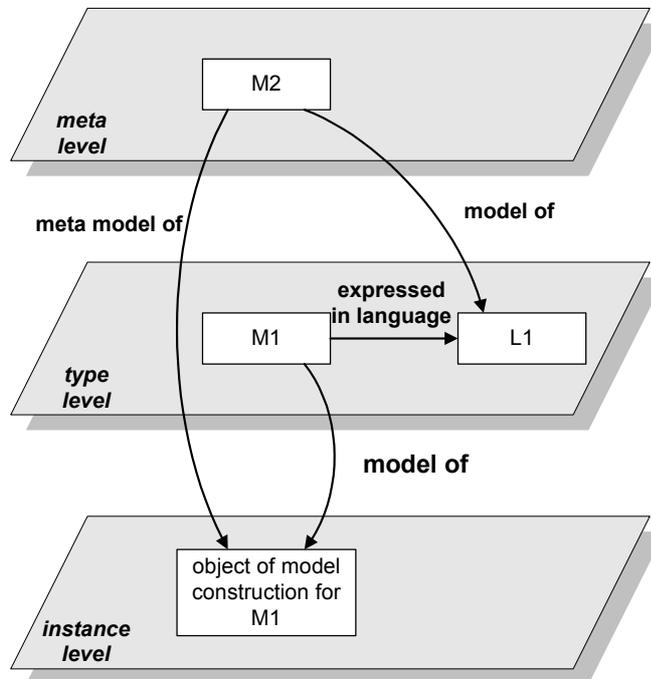


Abbildung 1: Abstraktions- und Modellierungsebenen

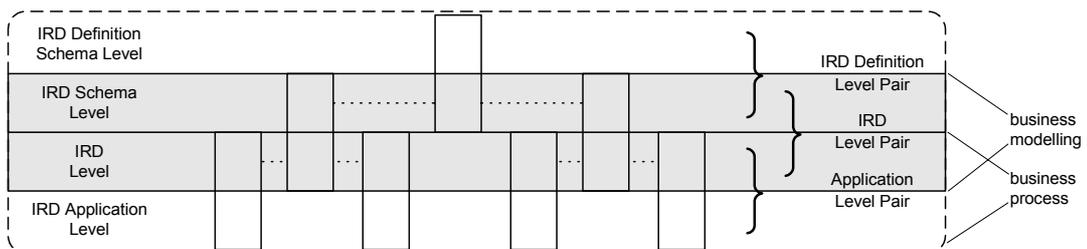


Abbildung 2: Ebenen und Ebenenpaare im IRDS-Framework

Ausgehend von diesen Überlegungen können drei Abstraktionsebenen des Modellierungsprozesses unterschieden werden:

¹² Vgl. Nissen et al. (1996), S. 37 ff.; Holten (1999), S. 11 ff.

¹³ Vgl. Jacobs, Holten (1995), S. 98 ff.; Pohl (1996), S. 80 ff.; ISO/IEC 10027; ISO/IEC (1990).

Die *Instanzenebene* umfaßt domänenrelevante Daten, z. B. *den Verkauf von PKWs in der Region Euopa an Händler im Zeitraum Juni 1999, der zu einem Umsatz von 3 Millionen EUROS führte*. Die nächste *Ebene* definiert den *Typ* von Instanzdaten im Sinne eines Datenbankschemas oder von Variablen. Für das obige Beispiel muß es eine Variable geben, die in der Lage ist, einen *gewissen Umsatzbetrag in EURO* für den *realisierten Verkauf von PKWs in der Region Euopa an Händler im Zeitraum Juni 1999* aufzunehmen. Die Konzepte und Begriffe, die auf der *Typeebene* benutzt werden, um das entsprechende Modell zu beschreiben, werden auf der nächst höheren Ebene der Abstraktion, der *Metaebene*, definiert. Im Beispiel muß auf der *Metaebene* definiert werden, daß es gewisse *Bezugsobjekte* gibt, die Gegenstand von betrieblichen Entscheidungen sind, und die mit *Kennzahlen* kombiniert werden können und dann zu sinnvollen Aussagen über die Bezugsobjekte in Form von *Fakten* führen können. Begriffe der *Metaebene* müssen, der Methodologie von WEDEKIND folgend mittels definierter Konstruktionsoperatoren eingeführt (rekonstruiert) werden.¹⁴

2.3 Kombination der Dimensionen: Ordnungsrahmen für Information-Warehouse-Entwicklungsprozesse

Da in jeder Entwicklungsphase gewisse Modelle als Dokumente produziert werden, können die skizzierten Dimensionen sinnvoll miteinander kombiniert werden, um die Entwicklung von Information Warehouses gemäß den Entwicklungsphasen und Abstraktionsebenen einordnen zu können. Das entsprechende Rahmenmodell ist in Abbildung 3 dargestellt.

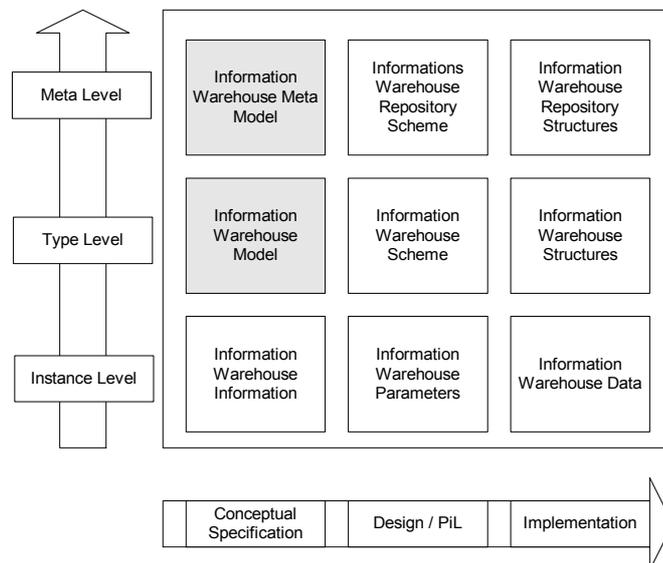


Abbildung 3: Ordnungsrahmen der Entwicklung von Information Warehouses

¹⁴ Vgl. Wedekind (1981); Wedekind (1980); Wedekind (1979); Wedekind, Ortner (1977); Holten (1999), S. 19-27.

Jede Abstraktionsebene beschreibt die Entwicklung von Information-Warehouse-Objekten von ihrer konzeptuellen Spezifikation bis hin zu ihrer Implementierung. Die Typebene beschreibt die üblichen Entwicklungsphasen des Information Warehouse selbst. Die Metaebene beschreibt die Entwicklung eines Information-Warehouse-Repositories von der Sprachdefinition bis hin zur Repository-Implementierung. Die Instanzenebene schließlich beschreibt die schrittweise Formalisierung von Informationsobjekten, die im Geschäftsprozeß anfallen und letztlich als Entitäten in Datenbanken ihren Niederschlag finden.

Jede Box des Ordnungsrahmens für Information-Warehouse-Entwicklungsprozesse ist durch eine eindeutige Kombination von Dokumenten und Entwicklungsaktivitäten charakterisiert. Es gibt Abhängigkeiten zwischen den Entwicklungsaktivitäten in einer Box und den Dokumenten, die in einer anderen Box erzeugt werden. Der Ordnungsrahmen soll diese Abhängigkeiten klar strukturieren und die Diskussion von Entwicklungsprozessen unterstützen. Im Vordergrund der weiteren Betrachtungen steht das Verhältnis von Sprachen zur semantischen Modellierung von Dispositiven Informationssystemen zu Notationen für die semantische Modellierung Dispositiver Informationssysteme. Bezugnehmend auf den Ordnungsrahmen läßt sich zunächst formulieren, daß entsprechende Sprachen und Notationen in die Box *Information Warehouse Meta Model* einzuordnen sind, wohingegen das semantische Modell selbst in die Box *Information Warehouse Model* gehört.

3 Sprache und Notation für die semantische Modellierung von Dispositiven Informationssystemen

3.1 Zum Verhältnis von Sprache und Notation

Gemäß WEDEKINDS Entwicklungsmethodologie für Datenbanken, besteht ein grundlegender Schritt der Entwicklung von Programmlösungen in der Normierung sprachlicher Mittel, die als Rekonstruktion von Begriffen dem operationalen Herstellen der Problemlösung in Form von Programmen vorausgehen muß.¹⁵ Mittels der *Objektypenmethode* können benötigte Begriffe, sogenannte *Objektypen*, schrittweise eingeführt werden.¹⁶ *Objektypen* sind abstrakte Mengen von Daten wie z. B. *Rechnung*, *Posten*, *Kunde* oder *Teil*.¹⁷ Sie erhalten ihre pragmatische Bedeutung für die programmiersprachliche Lösung des gegebenen Problems durch einen Konstruktionsprozeß.¹⁸ Beispielsweise kann der Begriff *Bezugsnachweis* im Rahmen der Entwicklung eines Warenwirtschaftssystems nur verstanden werden, wenn vorher die Begriffe

¹⁵ Vgl. Wedekind (1981), S. 74, 67 f.; Holten (1999), S. 21 f.

¹⁶ Vgl. Wedekind (1981), S. 112-124.

¹⁷ Vgl. Wedekind (1980), S. 666.

¹⁸ Vgl. Wedekind (1981), S. 69, S. 86.

Artikel und *Lieferant* eingeführt wurden, und *Bezugsnachweis* aus diesen beiden Grundbegriffe komponiert wird.

Die Idee WEDEKINDS wird hier auf die Lösung von Problemen für einen gesamten Problembereich, nämlich die Entwicklung Dispositiver Informationssysteme, übertragen. Es wird somit die Frage gestellt, welche *Menge Objekttypen* für die Entwicklung von Dispositiven Informationssystemen generell aus domänenspezifischer Sicht benötigt wird, und wie diese Menge beschrieben werden kann.

In der Literatur werden verschiedene Sprachkonstrukte zur semantischen Modellierung von Dispositiven Informationssystemen wie Dimensionen, Hierarchien, Netze von komplexen und mehrdimensionalen Informationsobjekten, Teilräume, Kennzahlen, Fakten, Anfragen an die Informationsbasis sowie zur Zuordnung von Fakten zu Berichten gefordert.¹⁹ Die geforderten Begriffe werden jedoch in der Regel nicht konstruktiv im Sinne von Objekttypen hergeleitet sondern mittels pragmatischer und intuitiver Ausführungen eingeführt. Gemessen an der Methodologie WEDEKINDS und der allgemein anerkannten Bedeutung Dispositiver Informationssysteme ist diese Situation unbefriedigend.

In einigen Quellen werden Sprachkonstrukte lediglich eingeführt, um Notationsvorschläge für die semantische Modellierung von Dispositiven Informationssystemen vorzuschlagen oder zu diskutieren.²⁰ Dabei wird nicht immer klar zwischen dem eingeführten Begriff (dem Objekttyp) und dem einem Begriff zugeordneten Notationselement unterschieden. Deshalb wird hier die Ansicht vertreten, daß im Rahmen der semantischen Modellierung Dispositiver Informationssysteme zwischen *Sprache* und *Notation* (bzw. *Repräsentation*) zu unterscheiden ist. Diese erforderliche Trennung ist die Konsequenz aus der Unterscheidung von Typebene und Metaebene, die im vorigen Kapitel eingeführt wurde, und der Anwendung von WEDEKINDS Objekttypenansatz im Sinne seiner Methodologie auf *beide* Ebenen. Sprachkonstrukte nämlich, die auf der Typebene zur semantischen Modellierung eines konkreten Dispositiven Informationssystems genutzt werden sollen, müssen vorher auf der Metaebene als Begriffe eingeführt worden sein. Die Modellierung auf der Typebene selbst bedeutet dann aber eine Definition der *konkret* benötigten *Meta-Objekttypen* (die ja Mengen darstellen) durch *Aufzählung ihrer Elemente*. Die *pragmatische Bedeutung dieser Elemente* ist jedoch vorher aufgrund der Konstruktionen auf der Metaebene spezifiziert worden. Dennoch müssen auch für die Elemente, die das Modell der Typebene ausmachen, geeignete Notationen und Notationskonventionen getroffen werden.

¹⁹ Vgl. Codd et al. (1993); Schelp (1998), S. 264-270; Golfarelli et al. (1998); Bulos (1996); Ehrenberg, Heine (1998); Gabriel, Gluchowski (1998), S. 494-496; Stöhr et al. (1999); Wietek (1999); Holthuis (1998).

²⁰ Vgl. Golfarelli et al (1998); Bulos (1996); Ehrenberg, Heine (1998); Gabriel, Gluchowski (1998), S. 494-496.

Es soll hier ausdrücklich betont werden, daß den Ausführungen WEDEKINDS, der im Sinne einer konstruktiven Wissenschaftstheorie verlangt, jedem Zeichen direkt bei seiner Einführung eine pragmatische Bedeutung zu geben,²¹ gefolgt wird. Allerdings wird WEDEKINDS methodischer Vorschlag in *zweifacher* Weise angewandt, was wie gezeigt zur Trennung der Typebene von der Metaebene führt. Folglich müssen im folgenden zunächst die Begriffe der Metaebene eingeführt und mittels einer geeigneten Notation dargestellt werden. Diese Begriffe bilden in ihrer Gesamtheit eine *Sprache*, die auf der Typebene genutzt wird, um die Objekttypen, die das semantische Modell eines Dispositiven Informationssystems spezifizieren, zu konstruieren. Auch auf der Typebene sind entsprechende Notationen erforderlich.

Zielgruppe der Notation auf der Metaebene sind Personen, die über Instrumente der Entwicklung von Dispositiven Informationssystemen (wie z. B. geeignete Sprachen) reden und nachdenken. Zielgruppe der Notation auf der Typebene sind Personen, die ein konkretes Dispositives Informationssystem in ihrem Arbeitsfeld im Unternehmen benötigen und entwickeln. Auf der Metaebene werden die Objekttypen im Sinne WEDEKINDS demnach fast gleichzeitig definiert und in einer geeigneten Notation dargestellt. Auf der Typebene jedoch sind (andere) Zeichen, die aufgrund der Definitionen der Metaebene bestimmte pragmatische Bedeutungen haben, bei der Konstruktion von (neuen) Objekttypen des semantischen Modells zu verwenden. Folglich gibt die Konstruktion der Metaebene in Verbindung mit den Notationskonventionen der Typebene im Sinne der semiotischen Dreiteilung in Syntax, Semantik und Pragmatik von MORRIS²² die Syntax, Semantik und Pragmatik der zu verwendenden Notationselemente der Typebene vor. *Jedes* semantische Modell der Typebene, das die eingeführte Notation vor dem Hintergrund der Konstruktion der Metaebene verwendet, hat dann eine bestimmte pragmatische Bedeutung. Die Konstruktion des semantischen Modells auf der Typebene selbst hat jedoch die *zweckadäquate und pragmatische* Spezifikation der *Elemente von Objekttypen* zum Gegenstand, die auf der Metaebene eingeführt wurden. Syntax und Semantik dieser *Elemente* sind (bei jeder beliebigen Auswahl) aufgrund der Konstruktionen der Metaebene und der Beziehungen zwischen Meta- und Typebene jedoch *vorgegeben*. Die Elemente der Typebene stellen außerdem im Sinne WEDEKINDS selber wieder Objekttypen dar, da es sich hierbei auch um Mengen von abstrakten Daten handelt.

3.2 Metamodell zur semantischen Modellierung Dispositiver Informationssysteme – Explizierung der Sprachdefinition

An dieser Stelle sollen nur die Sprachelemente skizzenartig als Objekttypen konstruiert werden, die für die semantische Modellierung Dispositiver Informationssysteme unmittelbar

²¹ Vgl. Wedekind (1981), S. 64.

benötigt werden. Ausführliche Herleitungen dieser und weiterer Objekttypen des Metamodells finden sich bei HOLTEN.²³ Bei der Konstruktion des Metamodells für die semantische Modellierung Dispositiver Informationssysteme wurde der betriebswirtschaftlich abgesicherten Bereitstellung von Komponenten für die Darstellung multidimensionaler Informationsobjekte für das Management besondere Bedeutung beigemessen.

Der erste Objekttyp, der als Grundbegriff eingeführt wird, ist *Dimension*. Die Einführung als Grundbegriff bedeutet, daß Dimension als gegeben und mit einem akzeptierten Grundverständnis belegt hingenommen wird.²⁴ Die Bedeutung von Grundbegriffen wird letztlich durch Aussagen definiert, die für alle Objekte gelten, die zu diesem Objekttyp gehören sollen.²⁵ Dimensionen ergeben sich aus der Wahrnehmung und zweckgeleiteten Konstruktion des Modellerstellers auf der Typebene, der ein Dispositives Informationssystem in einer konkreten Unternehmenssituation einführt. Sie spannen einen Raum der sachlich-zeitlichen Struktur des Untersuchungsfeldes auf, der für den konkreten Fall als zweckmäßig angesehen wird.²⁶ Die Existenz einer Dimension geht mit der eindeutigen Zuordnung von bestimmten Bezugsobjekten, den sogenannten *Dimensions-Bezugsobjekten*, einher. Diese zeichnen sich untereinander durch eine stärkere Bindung aus als zu Bezugsobjekten anderer Dimensionen. Diese starke Bindung ergibt sich ebenfalls aus dem Untersuchungszweck und der Domänensicht. In der min-max-Notation des ERM führt dies zu den (1,1):(0,n)-Kardinalitäten zwischen den Entitytypen *Dimensions-Bezugsobjekt* und *Dimension* des Metamodells und bedeutet, daß jedes Dimensions-Bezugsobjekt existentiell von einer Dimension abhängt (siehe zur Konstruktion des Metamodells Abbildung 4). Dimensionen werden zu Gruppen von Dimensionen zusammengefaßt, wenn eine Art von Dimensions-Bezugsobjekten nach verschiedenen Kriterien zu analysieren ist. Dies kann zum Beispiel für Geschäftsstellen der Fall sein, die nach den Kriterien Lage oder Konkurrenzsituation analysiert werden sollen. Da diese Kriterien aus Domänensicht orthogonal sind, sind verschiedene Dimensionen erforderlich. Mit den Objekttypen *Dimension-Gruppe* und *D-DG-Zuo* kann die Zugehörigkeit zu einer Art von Dimensions-Bezugsobjekten ausgedrückt werden.

²² Vgl. Wedekind (1981), S. 62; Lorenz (Semiotik) (1995), S. 783 f.

²³ Vgl. Holten (1999), S. 75-117. Siehe auch Becker, Holten (1998); Holten, Knackstedt (1999).

²⁴ Vgl. Holten (1999), S. 87; Wedekind (1981), S. 51 f.

²⁵ Vgl. Holten (1999), S. 23. Die entsprechende Konstruktionsoperation wird von WEDEKIND als *Subsumption* bezeichnet. Vgl. Wedekind (1981), S. 112-116.

Dimensions-Bezugsobjekte sind strikt hierarchisch angeordnet. Es kann beispielsweise als Dimensionen eine Hierarchie über Artikel, einer Verkaufsorganisationshierarchie und die hierarchische Zusammenfassung von Zeitintervallen geben. Das Modell verwendet für diese Hierarchien den Begriff *DBO-Hierarchie*, der als rekursiver Relationshiptyp mit der Kardinalität $(0,1):(0,n)$ eine strenge Hierarchie von Dimensions-Bezugsobjekten einer Dimension definiert. Jedes Paar, das Element von DBO-Hierarchie ist, drückt eine eindeutige Sohn-Vater-Beziehung aus (z. B. (Bezirk Münsterland, Gebiet Norddeutschland), (Bezirk Hannover, Gebiet Norddeutschland)). Im Rahmen der semantischen Modellierung werden zusätzlich Konstrukte zur Spezifikation von Hierarchiestufen benötigt. Im Modell wird daher der Objekttyp *Hierarchiestufe* eingeführt, der mit dem Entitytyp *Dimension* über den Relationshiptyp *D-HS-Zuo* in Beziehung steht. Um die Hierarchiestufen einer Dimension in eine feste Ordnung bringen zu können, ist der Relationshiptyp *D-HS-Ordnung* einzuführen, der den uminterpretierten Relationshiptypen *D-HS-Zuo* rekursiv mit sich selbst verbindet. Die Kardinalitäten $(0,1):(0,1)$ erzwingen die Spezifikation einer Sequenz, bei der das erste Element keinen Vorgänger und das letzte Element keinen Nachfolger hat. Außerdem muß jedes *Dimensions-Bezugsobjekt* eindeutig einem Element von *D-HS-Zuo* zugeordnet werden, sich also in der zugehörigen Dimension genau auf einer bestimmten Hierarchiestufe befinden. Der Relationshiptyp *DBO-DHS-Zuo* drückt mittels der Kardinalitäten $(1,1):(1,n)$ aus Sicht des Dimensions-Bezugsobjektes die erforderliche existentielle Abhängigkeit aus. Ein Dimensions-Bezugsobjekt ist somit durch *zwei* existentielle Abhängigkeiten geprägt: Es muß einer Dimension eindeutig zugeordnet sein und es muß sich auf genau einer Hierarchiestufe der Dimension befinden.

Zur Modellierung multidimensionaler Informationsobjekte wird auf den in der Betriebswirtschaftslehre ausführlich diskutierten Begriff *Bezugsobjekt* zurückgegriffen. Bezugsobjekte sind „alle selbständigen Maßnahmen, Vorgänge und Tatbestände, die eigenständiges Dispositionsobjekt oder Untersuchungsobjekt sein können“.²⁷ Ein Beispiel für ein *dreidimensionales* Bezugsobjekt ist *(B1):Warengruppe Kosmetik, Bezirk Münsterland, 20. Kalenderwoche des Jahres 1999*. Bezugsobjekte werden in netzwerkartigen Strukturen verknüpft. Im Modell führt dies zum Relationshiptypen *BO-Struktur*, der rekursiv mit dem Entitytypen *Bezugsobjekt* durch die Kardinalitäten $(0,n):(0,n)$ verbunden ist. Die Struktur ermöglicht die Anordnung

²⁶ Es wird hier grundsätzlich von einem konstruktivistischen Modellierungsverständnis ausgegangen. Vgl. Holten (1999), S. 9 f.; Schütte (1998), S. 59-62.

²⁷ Riebel (1979), S. 869.

mehrerer übergeordneter und untergeordneter Bezugsobjekte zu einem gegebenen Bezugsobjekt. Beispielsweise sind die dreidimensionalen Bezugsobjekte *(B2):Warengruppe Kosmetik, Gebiet Norddeutschland, 20. Kalenderwoche des Jahres 1999* (übergeordnete Komponente der Regionsdimension) und *(B3):Warengruppe Kosmetik, Bezirk Münsterland, Jahr 1999* (übergeordnete Komponente der Zeit-Wochen-Dimension) beide dem ersten Bezugsobjekt (B1) aus Domänensicht überzuordnen.

Die Identifikation von Bezugsobjekten erfolgt über die Koordinaten, die ihre Dimensionsausprägungen definieren. Bezugsobjekte sind also im Sinne von Vektoren zu verstehen und werden zunächst spezialisiert in *Kombinierte Bezugsobjekte*. Dimensions-Bezugsobjekte dienen als Koordinaten der Bezugsobjekte. Indem für jede Dimension des Kombinierten Bezugsobjektes ein konkretes Dimensions-Bezugsobjekt angegeben wird, ist ein Kombiniertes Bezugsobjekt eindeutig identifiziert. Dieser Zusammenhang wird durch den Relationstypen *K-BO-Koordinaten* abgebildet. Elemente von K-BO-Koordinaten sind beispielsweise die Paare *(B1, Warengruppe Kosmetik)*, *(B1, Bezirk Münsterland)*, *(B1, 20. Kalenderwoche des Jahres 1999)*, *(B2, Gebiet Norddeutschland)*. Da Bezugsobjekte außerdem in Dimensions-Bezugsobjekte spezialisiert werden und jedes Dimensions-Bezugsobjekt auch ein Kombiniertes Bezugsobjekt sein soll, handelt es sich um eine nicht-disjunkte und totale Spezialisierung.

Auf der Grundlage der Konstruktion der K-BO-Koordinaten kann der Raum der multidimensionalen Bezugsobjekte durch kombinatorische Berechnungen bei bekannten Dimensions-Bezugsobjekten jeder Dimension spezifiziert werden. Zur Bildung von Teilräumen, die der adressatengerechten Zusammenstellung von Bezugsobjekten dienen, werden die Begriffe *Dimensions-Ausschnitt* und *Dimensions-Ausschnitt-Kombination* eingeführt. Sie ermöglichen es, Teildimensionen zu bilden und zu Teilräumen (vieldimensionalen Teilwürfeln) zu kombinieren. Diese Begriffe werden der Forderung gerecht, abteilungs- oder aufgabenträgerspezifische Ausschnitte aus der Menge der Bezugsobjekte definieren zu können.

Mit den bisher eingeführten Objekttypen kann die Struktur einer gesamten vieldimensionalen Informationsbasis sowie von Auszügen daraus semantisch spezifiziert werden. Allerdings sind für Dispositive Informationssysteme noch Aspekte (oder „Measures“) zu definieren, die die benötigten Sichten des Managements auf das Unternehmensgeschehen weiter spezifizieren können. Als nächster Grundbegriff wird daher der Begriff *Kennzahl* eingeführt. Die Bedeutung von Kennzahlen für die Konzeption von Dispositiven Informationssystemen ist unbestritten. Kennzahlen erfassen quantitativ erfaßbare Sachverhalte und Zusammenhänge in konzent-

rierter Form.²⁸ Auf diese Weise sollen sie Führungskräften einen schnellen und umfassenden Überblick über komplizierte Strukturen und Prozesse ermöglichen. Jedoch erhalten Kennzahlen erst über die Kombination mit betriebswirtschaftlich relevanten Sachverhalten Informationscharakter. Dieselbe Überlegung veranlaßte RIEBEL dazu, im Rahmen der Einzelkosten- und Deckungsbeitragsrechnung Bezugsobjekten Wert- und Mengengrößen zuzuordnen.²⁹ Der Informationscharakter ist somit von der *zweckadäquaten Kombination* von Kennzahlen und Bezugsobjekten *abhängig*. Die Kennzahl *Umsatzrentabilität* beispielsweise liefert keine Information, wenn sie nicht einem Bezugsobjekt zugeordnet ist. Eine solche Zuordnung geht aber über die definitorische Eigenschaft, quantitativ erfaßbare Sachverhalte in konzentrierter Form zu erfassen, hinaus und erfordert die Einführung eines neuen Begriffs.³⁰ In Anlehnung an das OLAP- und Data-Warehouse-Konzept wird für die Kombination von Bezugsobjekt und Kennzahl der Begriff *BO-Fakt* eingeführt. Auch im Rahmen der semantischen Modellierung multidimensionaler Datenstrukturen wird die Trennung von Kennzahlen und Bezugsobjektstrukturen empfohlen.³¹ Da theoretisch jede Kennzahl mit jedem Bezugsobjekt kombiniert werden kann, hat der entsprechende Relationshiptyp die Kardinalität (0,n):(0,n).

Erst die Einführung dieses neuen Begriffes macht es möglich, sinnvoll über Verdichtung und Aggregation zu reden, da für eine Aggregation die Ordnungsbeziehungen der Bezugsobjekte erforderlich sind. Aggregationen sind somit lediglich in Verbindung mit Fakten möglich.³² Für die geeignete Wahl von Aggregationsoperatoren ist die Unterscheidung von *Bestands-* und *Bewegungszahlen* als Spezialisierung des Begriffes Kennzahl notwendig. Fakten, die aus Bestandszahlen gebildet werden, lassen sich summieren, da sich Bestandszahlen auf *Zeitpunkte* beziehen. Bewegungszahlen als Komponenten von Fakten erfordern hingegen betriebswirtschaftlich sinnvolle Aggregationsoperatoren, da sie sich auf *Zeiträume* beziehen. Weitere Kriterien zur Wahl von Aggregationsoperatoren werden aus semantischer Sicht nicht benötigt.³³

Zur Modellierung von Rechensystemen werden die Begriffe *Kennzahl-Struktur*, *Operator*, *Berechnungs-Ausdruck* und *BA-Reihenfolge* eingeführt. Diese Begriffe erlauben die Formulie-

²⁸ Vgl. Staehle (1967), S. 62; Reichmann (1997), S. 19.

²⁹ Vgl. Riebel (1994), S. 759.

³⁰ Vgl. Holten (1999), S. 93 ff.

³¹ Vgl. Gabriel, Gluchowski (1997), S. 32 ff.

³² Vgl. Holten (1999), S. 97 f.

³³ Vgl. Holten (1999), S. 97 f.

rung der benötigten Berechnungsformeln für Verhältniszahlen oder sonstige berechnete Kennzahlen auf der Typebene. Kennzahlensysteme, insbesondere Ordnungssysteme, die sich durch rein sachlogische Beziehungen zwischen den einzelnen Kennzahlen auszeichnen,³⁴ werden mit dem Entitytyp *Kennzahl-System* modelliert. Es wird gefordert, daß Kennzahlen den Kennzahlensystemen zuzuordnen sind, und daß in Ordnungssystemen eine hierarchische Ordnung der Kennzahlen existiert. Die Objekttypen *K-KS-Zuo* und *K-KS-Hierarchie* drücken dies aus.³⁵

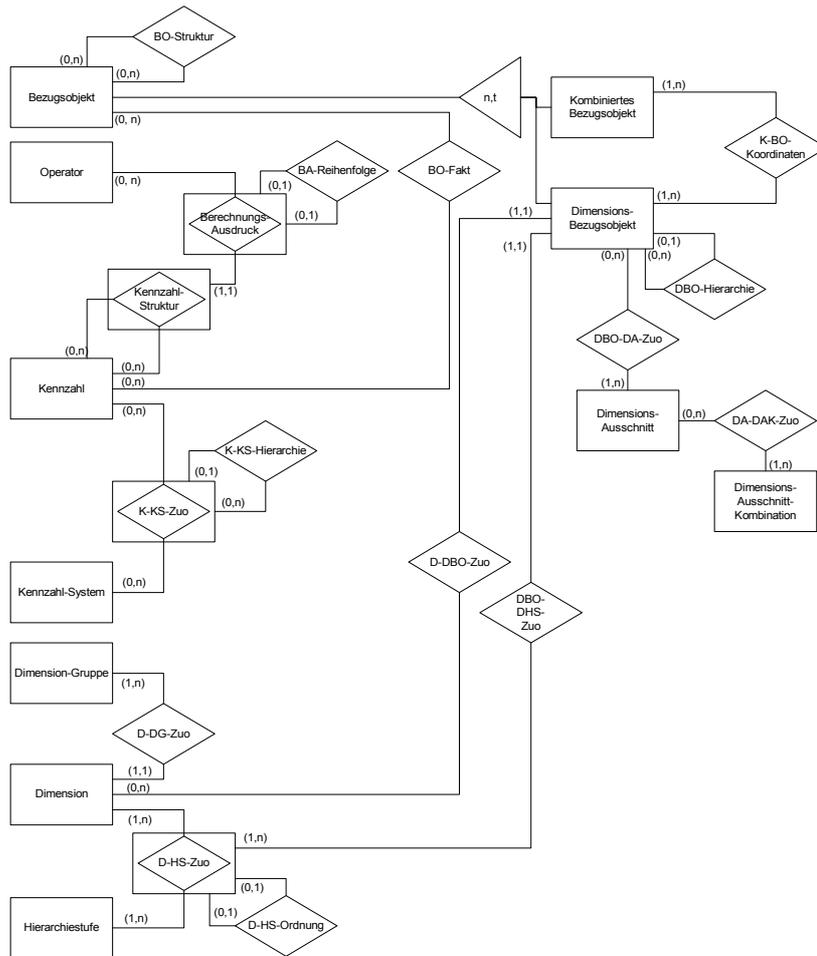


Abbildung 4: Metamodell der semantischen Modellierung von Dispositiven Informationssystemen

³⁴ Vgl. Groffmann (1992), S. 76; Reichmann (1997), S. 32 f.

³⁵ Vgl. dazu Holten (1999), S. 100 ff.; Holten, Knackstedt (1999).

3.3 Notation zur semantischen Modellierung Dispositiver Informationssysteme

Es sind in der Literatur verschiedene Notationsvorschläge für die semantische Modellierung Dispositiver Informationssysteme gemacht worden.³⁶ Zur Anwendung der im vorigen Abschnitt definierten Sprache (also zur Instantiierung des Metamodells bei der semantischen Modellierung Dispositiver Informationssysteme) ist jedoch eine eigene Notation erforderlich, da sich die vorgestellten Konzepte in den anderen Notationen nicht immer in der benötigten Form wiederfinden.³⁷ Im Rahmen von Anwendungen der vorgestellten Sprache in echten Unternehmenssituationen hat es sich als zweckmäßig erwiesen, eine auf dem ER-Modell von CHEN³⁸ basieremde Notation zu verwenden, die hier kurz vorgestellt werden soll.

Die folgende Clusterung hat sich bei der Darstellung komplexer semantischer Modelle Dispositiver Informationssysteme bewährt (siehe Abbildung 5). Dimensionen werden von oben links nach unten links angeordnet. Es wird zwischen nicht obligatorischen Dimensionen, die je nach Unternehmenssituation zu spezifizieren sind, und obligatorischen Dimensionen, die immer anzugeben sind, unterschieden. Als obligatorisch sind eine Dimension *Zeit* und eine Dimension *Wertansatz* zu spezifizieren. Zu allen Dimensionen sind Instanzenbeispiele anzugeben und unter den entsprechenden Entitytypen der Dimension in Klammern zu schreiben. Diese Notationskonvention wird im folgenden noch weiter spezifiziert. Ganz unten links ist der Entitytyp Kennzahl-System anzuordnen. Es werden als Instanzen die entsprechenden Kennzahlen des Kennzahlensystems angegeben. Auch dies ist später noch weiter zu spezifizieren.

Ein Relationshiptyp stellt die Verbindung zwischen allen beteiligten Dimensionen her. Es handelt sich um ein Kombiniertes Bezugsobjekt eines Informationsobjektes mit einem Namen. Der Relationshiptyp ist entsprechend mit *Kombiniertes Bezugsobjekt InfoObjekt „Name“* zu bezeichnen. Die beteiligten Dimensionen gehen *immer* mit den Kardinalitäten (0,m):(0,m) in den Relationshiptypen ein. Dieser Relationshiptyp wird zum Entitytyp umdefiniert und geht eine Beziehung mit dem Entitytyp *Kennzahl-System* ein. Der resultierende Relationshiptyp ist mit *InfoObjekt „Name“* zu bezeichnen und mit den Kardinalitäten (0,m):(0,m) zu versehen. Er stellt im fertigen semantischen Modell das multidimensionale Informationsobjekt dar. Die

³⁶ Vgl. zu einem Überblick Gabriel, Gluchowski (1998), S. 497-502; Schelp (1998), S. 270-274; Gabriel, Gluchowski (1997), S. S. 18-37.

³⁷ Ein detaillierter Überblick setzt einen Vergleich der Konzepte auf der Ebene des Metamodells voraus. Hierzu ist es zunächst erforderlich, die häufig nur implizit definierten Konstrukte der vorgeschlagenen Notationen in Form eines Metamodells zu explizieren. Ein solcher Versuch muß an dieser Stelle unterbleiben.

³⁸ Vgl. Chen (1976).

Kardinalitäten weisen für die Implementierung darauf hin, daß entsprechende Auswahlmasken für die Berichtsaufrufe zu realisieren sind, die eine weitere individuelle Einschränkung der anzuzeigenden Informationsobjekte erlauben. Solche Masken stellen heute den Standard bei entsprechenden Tools dar.

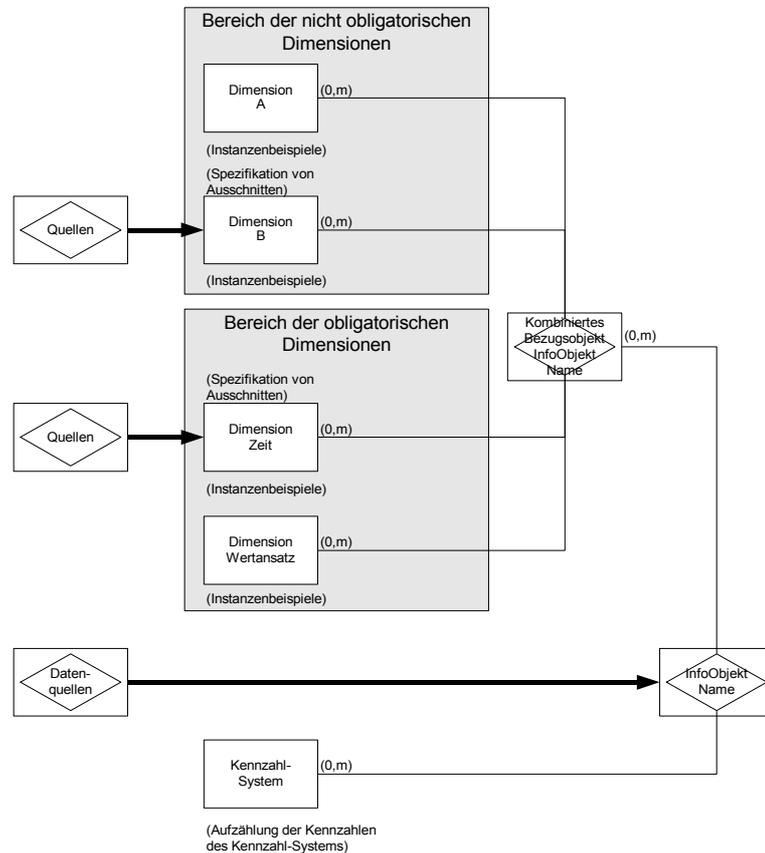


Abbildung 5: Anordnung der Modellelemente eines Informationsobjektmodells

Als Hinweise für den Übergang zum DV-Konzept (bzw. zum Design des Systems) gibt die Notation des fettgedruckten Pfeils Hinweise. In dem links anzugebenden uminterpretierten Relationshipstypen sind die Systeme zu nennen, die die zu integrierenden Daten für das Dispositive Informationssystem bereitstellen. Die Pfeilnotation ist auch zu verwenden, um Hinweise auf Systeme oder Dokumente zu geben, die den genauen Aufbau einer Dimension spezifizieren. So müssen beispielsweise die verschiedenen Artikel- oder Warengruppenhierarchien, die in Warenwirtschaftssystemen ohnehin spezifiziert sind, oder Kostenstellenhierarchien, die in Controllingssystemen spezifiziert sind, im Rahmen der semantischen Modellierung Dispositiver Informationssysteme explizit definiert sein. Weiterhin können auch Organisationshandbücher als Dokumente angegeben werden, falls dort beispielsweise eine Hierarchie von Verkaufsverantwortlichen spezifiziert ist, die sich noch nicht in Informationssystemen wiederfindet. Die

Pfeilnotation ermöglicht es somit, den Modellierungsaufwand bei der semantischen Modellierung von Dispositiven Informationssystemen zu reduzieren, da entsprechende Hierarchien selbstverständlich *vollständig* zu spezifizieren sind. Der Relationshiptyp *InfoObjekt* „Name“ ist entsprechend umzuinterpretieren. Abhängige Modellkonstrukte sind grundsätzlich rechts von den Konstrukten anzuordnen, von denen sie abhängig sind. Dies erleichtert das Lesen der Modelle und macht den Konstruktionsprozeß nachvollziehbar.

Im folgenden werden die Notationskomponenten angegeben, die die Objekttypen des Metamodells auf der Typebene instantiiieren. Anhand des Metamodells läßt sich die Vollständigkeit des semantischen Moells aus syntaktischer Sicht prüfen. Eine Vollständigkeit aus semantischer und pragmatischer Sicht ist jedoch höchstens durch Konsens der Modellierer herzustellen.³⁹

Der Meta-Objekttyp *Dimension* wird durch Entitytypen instantiiert, die gemäß der Konvention zur Clusterung den oberen linken Teil des semantischen Modells ausmachen. Der Name einer Dimension geht in die Bezeichnung des entsprechenden Entitytypen ein. Die genauen Konventionen zur Benennung werden noch weiter spezifiziert. Vorhandene Dokumente oder Datenstrukturen in Informationssystemen, die den exakten Aufbau einer Dimension spezifizieren, sind mittels der Pfeilnotation als Verweis einzufügen. Solche Dokumente wie etwa der CCG-Warengruppenhierarchie oder unternehmensspezifischer Warengruppenhierarchien, die additiv oder alternativ verwendet werden können, instantiiieren die Meta-Objekttypen *Dimensions-Bezugsobjekt* und *DBO-Hierarchie*. Die Pfeilnotation instantiiert den Meta-Objekttyp *D-DBO-Zuo*, der jedes Bezugsobjekt genau einer Dimension zuordnet.

Bezüglich der semantischen Spezifikation der Hierarchien von Dimensions-Bezugsobjekten haben sich vier Arten von Hierarchieausprägungen als besonders nützlich erwiesen. Diese vier Arten führen zu spezifischen Notationskonventionen, die erkennen lassen, um welche Art der Hierarchie es sich handelt. Abbildung 6 zeigt die Arten der Hierarchieausprägungen und die zugehörigen Notationskonventionen. Für *einstufige* Hierarchien werden seitens der Informationsempfänger keine Aggregationen von Fakten gefordert. In die Bezeichnung des Entitytypen einstufiger Dimensionen geht entsprechend der Zusatz „*nicht aggregiert*“ ein. Für den Fall der Dimension Wertansatz ist beispielsweise keine Aggregation von Fakten vorzusehen. Die Bezeichnung lautet gemäß der getroffenen Konvention folglich „*Wertansatz nicht aggregiert*“. Die Dimensions-Bezugsobjekte dieser Dimension ermöglichen die Betrachtung betrieblicher Sachverhalte gemäß der realisierten Ist-Situation oder gemäß verschiedener Planungsszenarien.

Eine Aggregation ist gegebenenfalls nach anderen Dimensionen sinnvoll. In Klammern unterhalb des Entitytypen werden seine Instanzen, die den einzelnen Dimensions-Bezugsobjekten entsprechen, aufgeführt. Ist eine vollständige Aufzählung zu umfangreich, kann mit der eingeführten Pfeilnotation auf entsprechende Quellen (Informationssysteme oder Dokumente) verwiesen werden.

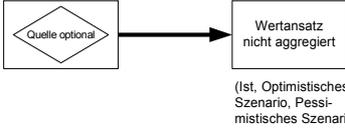
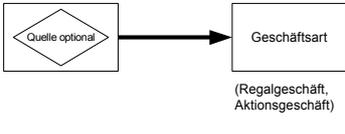
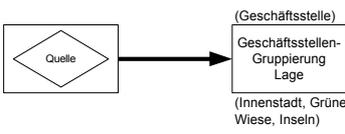
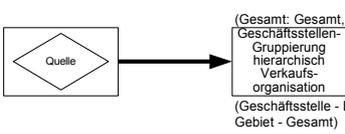
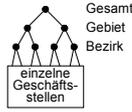
Art der Hierarchie	Notationskonvention	Beispiel
einstufig	 <p>(Ist, Optimistisches Szenario, Pessimistisches Szenario)</p>	Ist, Optimistisches Szenario, Pessimistisches Szenario
zweistufig	 <p>(Regalgeschäft, Aktionsgeschäft)</p>	Gesamt ├── Regalgeschäft └── Aktionsgeschäft
dreistufig	 <p>(Innenstadt, Grüne Wiese, Inseln)</p>	Gesamt ├── Innenstadt │ ├── einzelne Geschäftsstellen └── Grüne Wiese ├── einzelne Geschäftsstellen └── Inseln └── einzelne Geschäftsstellen
mehrstufig	 <p>(Geschäftsstelle - Bezirk - Gebiet - Gesamt)</p>	 <p>Gesamt Gebiet Bezirk einzelne Geschäftsstellen</p>

Abbildung 6: Vier Arten der Hierarchieausprägungen für Dimensions-Bezugsobjekte und Notationskonventionen

Im Falle von *zweistufigen* Hierarchien werden verschiedene Ausprägungen innerhalb einer Dimension betrachtet, die sinnvoll zu einem übergeordneten Dimensions-Bezugsobjekt zusammengefaßt werden können. Bei der Bezeichnung entfällt der Zusatz „*nicht aggregiert*“, so daß im Falle der Dimension Geschäftsart lediglich „*Geschäftsart*“ als Bezeichnung des Entitytypen aufzunehmen ist. In Klammern unterhalb des Entitytypen sind die Dimensions-Bezugsobjekte der unteren Hierarchiestufe aufzuführen. Gegebenenfalls ist die Pfeilnotation als Verweis zu verwenden. Im Falle des stationären Einzelhandels sind beispielsweise die Geschäftsarten Regalgeschäft und Aktionsgeschäft aufzuführen. Die Notation spezifiziert, daß

³⁹ Vgl. Pohl (1996), S. 36 f.; Schütte (1998), S. 121 ff.

neben den einzelnen Geschäftsarten auch das Gesamtergebnis auswertbar sein soll. Aufgrund dieser Konvention kann auf die Angabe einer Instanz „Gesamt“ verzichtet werden.

Mengen von Dimensions-Bezugsobjekten einer *dreistufigen* Hierarchie werden entsprechend der mittleren Hierarchiestufe *gruppiert*. In die Bezeichnung gehen die *Art* der Dimensions-Bezugsobjekte (z. B. Geschäftsstellen), der Zusatz *Gruppierung* und das *Kriterium* der Gruppierung der Dimensions-Bezugsobjekte (z. B. Lage) ein. Zur Analyse der Geschäftsstellen nach dem Kriterium der Lage wird also der Entitytyp „*Geschäftsstellen-Gruppierung Lage*“ zur Bezeichnung der entsprechenden Dimension eingeführt. In Klammern unterhalb des Entitytypen werden die Dimensions-Bezugsobjekte der mittleren Hierarchiestufe angegeben. Beispielsweise werden Geschäftsstellen danach gruppiert, ob sie in Innenstädten, auf der Grünen Wiese oder auf Inseln (Urlaubsgebiete) liegen. Die oberste Stufe faßt dann diese Gruppierungen zur Gesamtheit zusammen. Mittels der Pfeilnotation ist auf die Quellen zu verweisen, die dreistufige Hierarchien vollständig spezifizieren.

Hierarchien, die mehr als drei Stufen haben, werden in der Bezeichnung des Entitytypen mit dem Zusatz „*Gruppierung hierarchisch*“ versehen. Außerdem gehen die *Art* der Dimensions-Bezugsobjekte (z. B. Geschäftsstellen) sowie das *Kriterium* der Gruppierung (z. B. Verkaufsorganisation) mit in die Bezeichnung ein. Zur Analyse der Geschäftsstellen nach dem Kriterium der Verkaufsorganisation wird der Entitytyp „*Geschäftsstellen-Gruppierung hierarchisch Verkaufsorganisation*“ zur Bezeichnung der entsprechenden Dimension eingeführt. In Klammern unterhalb des Entitytypen werden die Stufenbezeichnungen der entsprechenden Hierarchie aufsteigend angegeben. Beispielsweise werden die Geschäftsstellen zur Abbildung der Verkaufsorganisation über die Hierarchiestufen Geschäftsstelle, Bezirk, Gebiet, Gesamt gruppiert. Die vollständigen Hierarchien sind in den Quellen zu spezifizieren, auf die die Pfeilnotation verweist.

Zusätzlich wird vereinbart, daß bei drei- und mehrstufigen Hierarchien eine Begrenzung des Ausschnittes der Hierarchie, der für einen Aufgabenträger sinnvoll ist, in *Klammern oberhalb* des Entitytypen angegeben wird. Ziel ist es, die Informationsmenge jedes Aufgabenträgers zu begrenzen, um der Informationsüberflutung zu begegnen. Im dreistufigen Fall, ist der Name der untersten Hierarchiestufe anzugeben, wenn eine Analyse bis auf diese Stufe gewünscht wird. Es wird z. B. dem Entitytyp „*Geschäftsstellen-Gruppierung Lage*“ mit seiner dreistufigen Hierarchie „*(Geschäftsstelle)*“ übergeordnet um abzubilden, daß eine Aufspaltung der Fakten nach einzelnen Geschäftsstellen gewünscht ist. Im mehrstufigen Fall ist oberhalb des Entitytyps der Dimension ein Tupel, das die Begrenzung des gewünschten Ausschnitts in der

Form (*höchste gewünschte Stufe: Knoten-Name, niedrigste gewünschte Stufe*) vornimmt, anzugeben. Dem Entitytyp „*Geschäftsstellen-Gruppierung hierarchisch Verkaufsorganisation*“ wird also beispielsweise das Tupel „(*Gesamt: Gesamt, Bezirk*)“ übergeordnet um festzulegen, daß für den betrachteten Aufgabenträger ein tiefster Drill-down von der höchsten Stufe der Hierarchie bis zur Ebene *Bezirk* vorzusehen ist. Soll die höchste Stufe des zu definierenden Ausschnittes nicht der höchste Hierarchieknoten sein, erlaubt der Name des entsprechenden Knotens hinter dem Doppelpunkt die Identifikation des gewünschten Hierarchieausschnitts.

Mit den Notationselementen zur Einschränkung der Informationsmenge ist der Meta-Objekttyp *Dimensions-Ausschnitt* instantiiert. Außerdem sind mit den Klammer-Notationen unterhalb der Entitytypen bei den drei- und mehrstufigen Hierarchien und den Verweisen durch die Pfeilnotation auf die Quellen zur exakten Spezifizierung der Hierarchien die Meta-Objekttypen *Hierarchiestufe, D-HS-Zuo, D-HS-Ordnung* und *DBO-DHS-Zuo* instantiiert. Die entsprechenden Instantiierungen für ein- und zweistufige Hierarchien erfolgen entweder direkt durch Aufzählung oder ebenfalls durch Quellenverweise mittels der Pfeilnotation indirekt. Es wird zusammenfassend gefordert, daß *jede* benötigte Hierarchie von Dimensions-Bezugsobjekten im Rahmen der semantischen Modellierung *explizit* zu modellieren und mit benannten Hierarchiestufen zu versehen ist.

Um im semantischen Modell visualisieren zu können, daß *mehrere* Dimensionen inhaltlich *verschiedene Kriterien einer Art* von Dimensions-Bezugsobjekten betreffen, wird eine übergeordnete Gruppierung nach dieser Art von Dimensions-Bezugsobjekten vorgenommen. Die resultierende Spezialisierung (bei umgekehrter Leserichtung) ist nicht disjunkt und partiell. Verschiedene Kriterien, die bei der Analyse einer Art von Dimensions-Bezugsobjekten zu analysieren sind, führen dann zu *eigenen* Dimensionen, wenn sie sich inhaltlich sinnvoll miteinander kombinieren lassen, wenn sie also aus Domänensicht orthogonal zueinander sind. Geschäftsstellen im stationären Einzelhandel als eine Art von Dimensions-Bezugsobjekten können beispielsweise nach den Kriterien Lage, Konkurrenzsituation, geografischer Gruppierung oder dem Alter analysiert werden. Da sich diese Kriterien aus Domänensicht sinnvoll miteinander kombinieren lassen, handelt es sich für die semantische Modellierung um zu unterscheidende Dimensionen. Die Anzahl der Dimensionen kann im konkreten Fall, z. B. im Handel oder in der Versicherungsbranche, folglich recht beachtlich werden. Mit dieser Notationskomponente werden die Meta-Objekttypen *Dimension-Gruppe* und *D-DG-Zuo* instantiiert. Das semantische Modell muß den Informationsbedarf einer genau definierten Gruppe von Aufgabenträgern spezifizieren. Da verschiedene Gruppen von Aufgabenträgern folglich

verschiedene Sichten auf das betriebliche Geschehen haben, ist es erforderlich, eine integrierte Basis von Dimensions-Bezugsobjekten und Dimensionen zu pflegen, um die Sichten konsistent zu halten. Es wird daher gefordert, daß eine *zentrale Bibliothek* von Dimensionen und Dimensions-Bezugsobjekten angelegt wird, die die umfassende Struktur der Gesamtheit der benötigten Dimensionen enthält. Die Auswahlen aus dieser Bibliothek erfolgen nach den benötigten Kriterien zur Analyse von Bezugsobjekten durch den Aufgabenträger. Die Bibliothek ist außerdem der geeignete Ort, um die geforderten Quellen zur exakten Spezifikation jeder einzelnen Dimension abzulegen. Die Clusterung von auf den Aufgabenträger abgestimmten Dimensionen und Dimensions-Bezugsobjekten in Form der aus Bibliotheken ausgewählten Dimensions-Ausschnitte instantiiert die Meta-Objekttypen *Dimensions-Ausschnitt-Kombination* und *DA-DAK-Zuo*.

Eine flexible Möglichkeit zur Spezifikation von Dimensions-Hierarchien und Dimensions-Ausschnitten für die Bibliothek von Bezugsobjekten bietet die in Abbildung 7 gezeigte Darstellung. Gemäß dieser Notation wird ein Knoten der Dimensions-Bezugsobjekt-Hierarchie markiert - hier Knoten A. Die einzelnen Knoten stellen Dimensions-Bezugsobjekte dar. Der markierte Knoten ist Teil des Dimensions-Ausschnitts. Von ihm müssen alle Pfeile ausgehen. Die Pfeile markieren, welche Knoten ebenfalls Teil des Dimensions-Ausschnitts sind. Im Beispiel gehören alle dem Knoten A untergeordneten, gleichgeordneten und übergeordneten Knoten zum Dimensions-Ausschnitt.

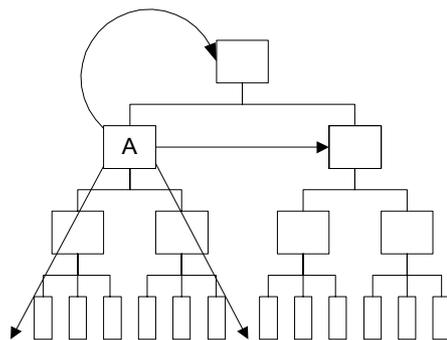


Abbildung 7: Pfeilnotation für Dimensionsausschnitte

Obligatorische und nicht obligatorische Dimensionen werden zu einem Kombinierten Bezugsobjekt des zu modellierenden Informationsobjektes zusammengeführt. Im semantischen Modell wird dazu der Relationshiptyp eingeführt, in den alle modellierten Dimensionen mit der Kardinalität (0,m) eingehen. Die Bezeichnung des Relationshiptyps setzt sich aus der Bezeichnung des zu modellierenden Informationsobjektes und dem voranzustellenden Zusatz „*Kombiniertes Bezugsobjekt InfoObjekt*“ zusammen. Für das Informationsobjekt zur monatlichen

Überwachung der Geschäftsstellen lautet die Bezeichnung entsprechend „*Kombiniertes Bezugsobjekt InfoObjekt Geschäftsstelle Monatlich*“. Diese Notationskomponente instantiiert die Meta-Objekttypen *Kombiniertes Bezugsobjekt* und *K-BO-Koordinaten*. Außerdem ist aufgrund der Spezifikation der Hierarchien der eingehenden Dimensionen das Netzwerk der Bezugsobjekte implizit definiert. Folglich ist der Meta-Objekttyp *BO-Struktur* ebenfalls instantiiert.

Kennzahlen sind genau wie Dimensionen und Dimensions-Bezugsobjekte in einer Bibliothek zentral zu verwalten. Für jede benötigte Sicht sind die definierten Kennzahlen entsprechend auszuwählen. Kennzahlen sind mit den üblichen arithmetischen Formeln zu spezifizieren oder allgemeinverständlich zu definieren. Es ist festzuhalten, ob es sich um eine Bestands- oder um eine Bewegungszahl handelt. Im Falle der Bewegungszahl ist der gewünschte Aggregationsoperator für die Fakten anzugeben. Im Falle von Bestandszahlen ist die Angabe optional. Kennzahlen sollten mittels der in Abbildung 8 gezeigten Tabelle spezifiziert werden. Diese Notationskomponenten instantiiieren Die Meta-Objekttypen zur Berechnung von Kennzahlen (*Operator*, *Berechnungs-Ausdruck*, *BA-Reihenfolge*, *Kennzahl-Struktur*) sowie *Kennzahl*, *Bestandszahl* und *Bewegungszahl*.

Kennzahl- bezeichnung	Bestandszahl / Bewegungszahl	Formel bei berech- neten Kennzahlen	Aggregations- operator	Erläuterung
Umsatz inkl. Mwst Scanner	Bestandszahl	-	Summierung	Umsatz inklusive Mehrwertsteuer auf Basis der Scannerdaten
Rohrertrag Scanner	Bestandszahl	Umsatz inkl. Scanner –Spanne nach GLD	Summierung	Rohrertrag berechnet auf Basis Scannerumsatz inkl. Mehrwertsteuer
Spanne nach GLD	Bestandszahl	(VK - EK nach GLD) / VK	Summierung	Handelsspanne nach gleitendem Durchschnittspreis

Abbildung 8: Definition von Kennzahlen

Für einen Aufgabenträger relevante Kennzahlen werden in einem Kennzahlensystem zusammengefaßt. In die Bezeichnung des entsprechenden Entitytypen gehen der Name des zu modellierenden Informationsobjektes sowie der voranzustellende Zusatz „*Kennzahl-System InfoObjekt*“ ein. Der Entitytyp für das Informationsobjekt zur monatlichen Überwachung der Geschäftsstellen ist demnach mit „*Kannzahl-System InfoObjekt Geschäftsstelle Monatlich*“ zu bezeichnen. Kennzahlen des Kennzahlensystems werden im semantischen Modell unterhalb des Entitytypen *Kennzahl-System* in Klammern aufgezählt. Die hierarchische Ordnung der Kennzahlen im Kennzahlensystem wird durch die Trennung mittels des Symbols „//“ dargestellt. Das Trennsymbol weist im Falle von Deckungsbeitragsrechnungen beispielsweise auch

auf benötigte Zwischensummen hin. Diese Notation gibt den Implementierern Hinweise zur Anordnung der benötigten Information in den zu erstellenden Berichten. Kennzahlensysteme sind möglichst mit einem durch die Pfeilnotation gekennzeichneten Verweis zu versehen. Als Lesehilfe sollte der beabsichtigte Aggregationsoperator jeder Kennzahl vorangestellt werden. Die Notationskomponenten zu Kennzahlensystemen instantiiieren die Meta-Objekttypen *Kennzahl-System*, *K-KS-Zuo* und *K-KS-Hierarchie*.

Der letzte zu instantiiierende Meta-Objekttyp, *BO-Fakt*, bewirkt die Zusammenführung der bisher erstellten Teile des semantischen Modells. Es wird ein Relationshiptyp eingeführt, dessen Bezeichnung das Präfix *InfoObjekt* erhält, gefolgt vom eigentlichen Namen des mehrdimensionalen Informationsobjektes. Der Relationshiptyp für das Informationsobjekt zur monatlichen Steuerung der Geschäftsstellen erhält beispielsweise die Bezeichnung „*InfoObjekt Geschäftsstelle Monatlich*“. Mit diesem Relationshiptyp ist die semantische Konstruktion abgeschlossen. Das Informationsobjekt hat soviele Dimensionen, wie durch die entsprechenden Entitytypen definiert wurden und umfaßt alle Kennzahlen des spezifizierten Kennzahlensystems. Durch die Zusammenführung von Bezugsobjekten und Kennzahlen ist mit dem Informationsobjekt die Menge der für einen Aufgabenträger relevanten Fakten spezifiziert. Der Relationshiptyp wird abschließend zum Entitytypen umdefiniert, um mit der Pfeilnotation Hinweise auf die wesentlichen Datenquellen für den Übergang zum DV-Konzept geben zu können. Der Name des semantischen Modells für ein Informationsobjekt setzt sich aus dem Namen des Informationsobjektes und dem Namen des Empfängers zusammen. Im Falle des Informationsobjektes zur monatlichen Steuerung von Geschäftsstellen, das für die Zentrale definiert wird, lautet die Bezeichnung „*Informationsobjekt: Geschäftsstelle Monatlich – Zentrale*“.

4 Vorgehen bei der semantischen Modellierung Dispositiver Informationssysteme und Beispiel

4.1 Ermittlung von Dispositiven Aufgaben

Zentrum und Ausgangspunkt einer semantischen Modellierung Dispositiver Informationssysteme muß die Erfassung und Systematisierung der relevanten dispositiven Aufgaben sein.⁴⁰ Dispositive Aufgaben lassen sich auf einer gewissen Abstraktionsstufe sinnvoll mittels einfa-

⁴⁰ Vgl. Holten (1999), S. 166.

cher Hierarchien systematisieren. Abbildung 9 zeigt beispielhaft einige relevante dispositive Aufgaben des Vertriebs im filialisierenden Einzelhandel. Die Steuerungs- und Regelungsaufgabe des Verkaufs läßt sich zusammenfassend beschreiben als die Beeinflussung der Filial-, Bezirks- und Gebietsleiter, die handelsbetrieblichen Leistungsfaktoren (Ware, Raum, Personal, Kapital) effizient einzusetzen.⁴¹

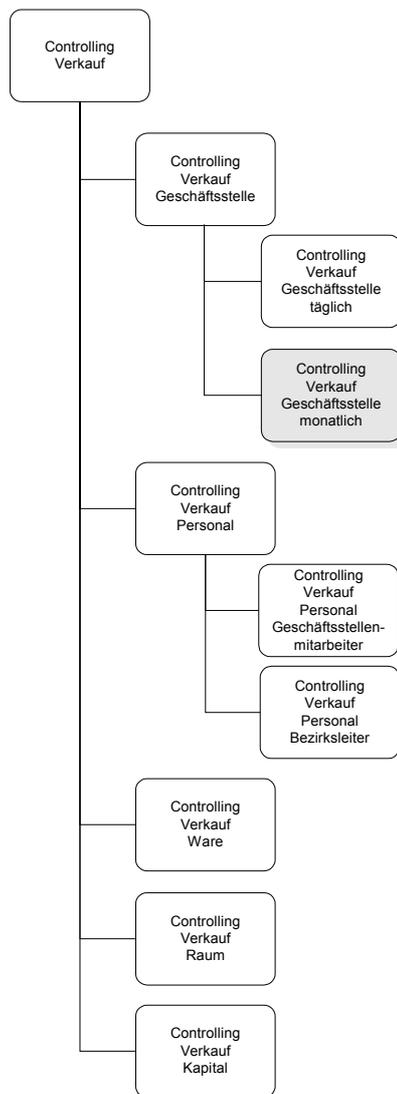


Abbildung 9: Übersicht relevanter dispositiver Aufgaben im Verkauf

4.2 Erstellung und Nutzung von Bibliotheken

Für die ermittelten dispositiven Aufgaben sind im nächsten Schritt auf der Grundlage des Begriffsapparates des Metamodells und unter Beachtung der Notationskonventionen die Bibliotheken der Bezugsobjekte und Kennzahlen und die benötigten Informationsobjekte in Form

⁴¹ Vgl. Barth (1993), S. 54.

von semantischen Modellen zu konstruieren. Dimensionen, Dimensions-Hierarchien, Dimensions-Bezugsobjekte und die diese genau spezifizierenden Quellen sind in die Bibliotheken einzufügen und erst im zweiten Schritt zu den benötigten semantischen Modellen der mehrdimensionalen Informationsobjekte zusammzusetzen. Entsprechendes gilt für Kennzahlen und Kennzahlensysteme. Dies ist in Abbildung 10 visualisiert.

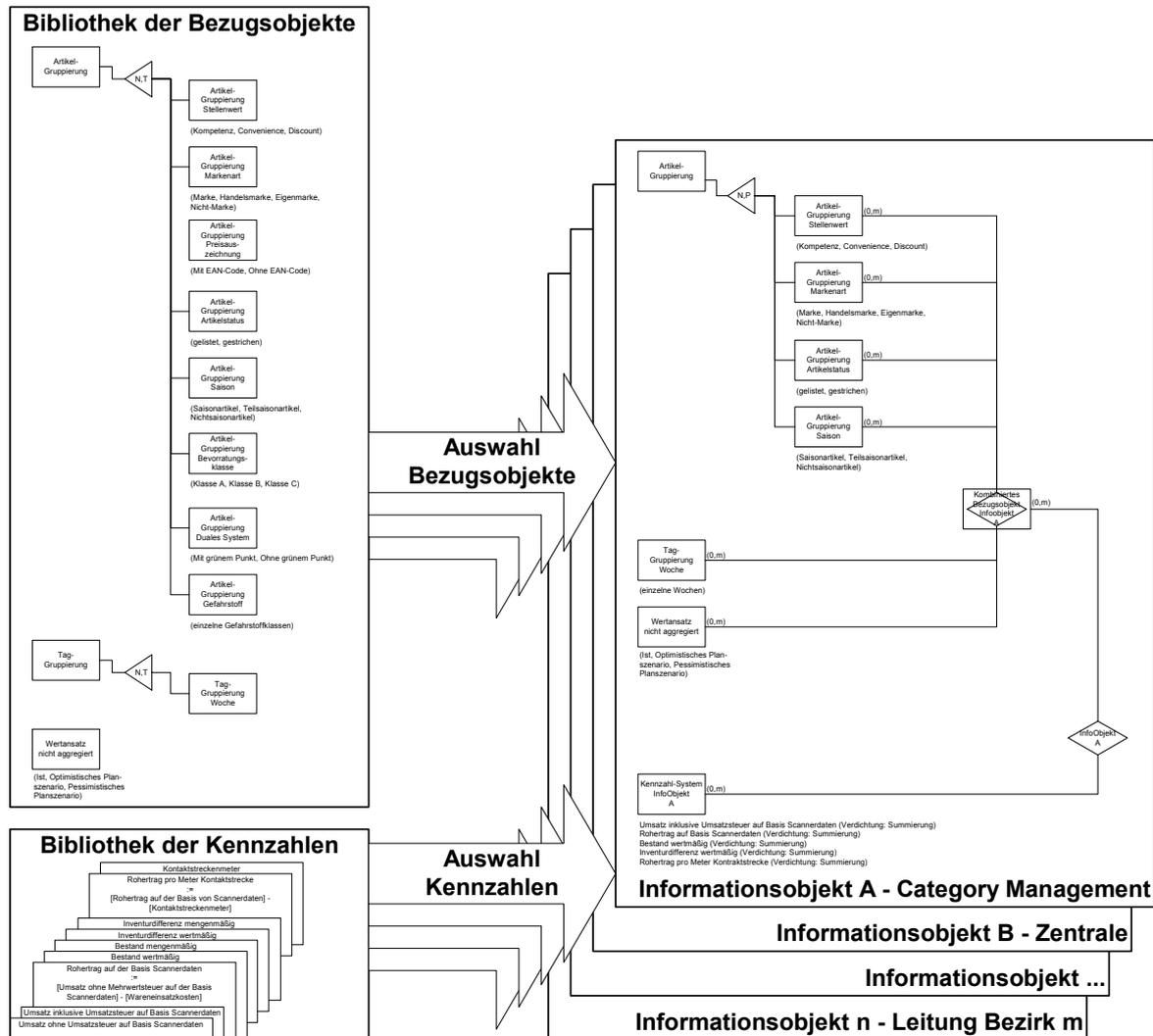


Abbildung 10: Erstellung und Nutzung der Bibliotheken

4.3 Semantische Modellierung der Informationsobjekte

Auf der Grundlage der Bibliotheken können die semantischen Modelle Dispositiver Informationssysteme zusammengesetzt werden. Abbildung 11 zeigt das semantische Modell des Informationsobjektes, das die Zentrale eines filialisierenden Einzelhändlers für die monatliche Überwachung der Geschäftsstellen verwenden kann. Das Informationsobjekt hat neun relevante Dimensionen. Fünf Dimensionen definieren spezielle Kriterien, nach denen Geschäftsstellen zu analysieren sind. Zusätzlich werden die Geschäftsstellen nach der Struktur der Verkaufsort-

ganisation und den Verantwortlichkeiten des Vertriebsmanagements analysiert. Für die beiden mehrstufigen Hierarchien in den Dimensionen „Geschäftsstellengruppierung hierarchisch Verkaufsorganisation“ und „Vertriebsmanager Gruppierung hierarchisch Verkaufsverantwortung“ sind jeweils die gesamten Hierarchien als Ausschnitte definiert. Bei den dreistufigen Geschäftsstellen-Hierarchien sind lediglich die Kriterien Konkurrenz und Umsatz-Vorjahr bis auf die unterste Stufe in das Informationsobjekt aufzunehmen. Bei allen weiteren dreistufigen Hierarchien reicht das Gruppierungskriterium ohne weitere Drill-down-Möglichkeit. Zeitlich ist das Informationsobjekt nach Monaten und Quartalen strukturiert. Ein Drill-down wird nur bis zur Ebene des Monats gewünscht. Taggenaue Daten sollen nicht bereitgestellt werden. Als relevante Kennzahlen sind neben Netto- und Bruttoumsätzen spannenwirksame Buchungen wie Bruch und Personalverkauf aufgenommen. Für alle Dimensionen und das Kennzahlensystem ist mittels der Pfeilnotation eine Quelle angegeben, die die exakten Definitionen bereithält.

Informationssystemen, wie es mit dem MetaFIS vorgestellt wurde.⁴² Solche Werkzeuge erlauben die fachkonzeptuelle Spezifikation Dispositiver Informationssysteme unabhängig von der Zielumgebung, in der das System umgesetzt werden soll. Sie unterstützen außerdem die Auswahl von geeigneten Werkzeugen zur Implementierung Dispositiver Informationssysteme, da sie eine solche Auswahl auf die Grundlage der fachlichen Anforderungen stellen. Außerdem erlauben sie es aufgrund der formalisierten Struktur der semantischen Modelle, einen Teil der Implementierung Dispositiver Informationssysteme über geeignete Schnittstellen zu automatisieren.⁴³

⁴² Vgl. Hoten (1999), S. 272 ff.; Becker, Holten (1998).

⁴³ Vgl. Holten (1999), S. 241 ff.

Literatur

- Balzert, H. (1996a): Lehrbuch der Software-Technik. Software Entwicklung. Heidelberg et al. 1996.
- Balzert, H. (1996b): From OOA to GUIs: The JANUS system. Journal of Object-Oriented Programming, (1996) 2, pp. 43-47.
- Barth, K.: Betriebswirtschaftslehre des Handels. 2. Aufl., Wiesbaden 1993.
- Becker, J.; Holten, R.: Fachkonzeptuelle Spezifikation von Führungsinformationssystemen. In: Wirtschaftsinformatik, 40 (1998) 6, S. 483-492.
- Becker, J.; Holten, R. (Wirtschaftsinformatik): Wirtschaftsinformatik. In: R. Berndt, C. Fantapié, P. Schuster (Hrsg.), Springers Handbuch der Betriebswirtschaftslehre. Berlin et al. 1998, S. 245-303.
- Becker, J.; Priemer, J.; Wild, R. G.: Modellierung und Speicherung aggregierter Daten. Wirtschaftsinformatik, 36 (1994) 5, S. 422-433.
- Bulos, D.: A New Dimension. OLAP Database Design. Database Programming & Design, 9 (1996) 6, S. 33-37.
- Chamoni, P.; Gluchowski, P.: Analytische Informationssysteme. Einordnung und Überblick. In: P. Chamoni, P. Gluchowski (Hrsg.), Analytische Informationssysteme. Berlin, Heidelberg 1998, S. 3-25.
- Chaudhuri, S.; Dayal, U.: An Overview of Data Warehousing and OLAP Technology. SIGMOD Record, 26 (1997) 1, S. 65-74.
- Codd, E. F.: The Relational Model for Database Management. Version 2. Reading, MA, Addison-Wesley, 1990.
- Codd, E. F.; Codd, S. B.; Salley, C. T.: Providing OLAP (On-Line Analytical Processing) to User Analysts: An IT Mandate. White Paper, E. F. Codd & Associates, 1993. http://www.arborsoft.com/essbase/wht_ppr/coddTOC.html. Abruf am 1998-06-05.
- Date, C. J.: An Introduction to Database Systems. Volume 1. 5th ed. Reading, MA, Addison-Wesley, 1990.

- Davis, A. M.: The Analysis and Specification of Systems and Software Requirements. In: R. H. Thayer, M. Dorfman (Eds.): Systems and Software Requirements Engineering, IEEE Computer Society Press-Tutorial 1990, S.119-134.
- Ehrenberg, D.; Heine, P.: Konzept zur Datenintegration für Management Support Systeme auf der Basis uniformer Datenstrukturen. Wirtschaftsinformatik 40 (1998) 6, S. 503-512.
- Embley, D. W.: Object Database Development. Concepts and Principles. Reading, MA. Addison-Wesley, 1998.
- Gabriel, R.; Gluchowski, P.: Semantische Modellierungstechniken für multidimensionale Datenstrukturen. HMD Theorie und Praxis der Wirtschaftsinformatik, 34 (1997) 195, S. 18-37.
- Gabriel, R.; Gluchowski, P.: Grafische Notationen für die semantische Modellierung multidimensionaler Datenstrukturen in Management Support Systemen. Wirtschaftsinformatik 40 (1998) 6, S. 493-502.
- Golfarelli, M.; Maio, D.; Rizzi, S.: Conceptual Design of Data Warehouses from E/R Schemes. Proceedings of the Hawaii International Conference on System Science. IEEE 1998.
- Groffmann, H.-D.: Kooperatives Führungsinformationssystem. Wiesbaden 1992.
- Gupta, A.; Horowitz, E.: Object-Oriented Databases with Applications to CASE, Networks, and VLSI CAD. Englewood-Cliffs, NJ. Prentice-Hall 1991.
- Holten, R.: Entwicklung von Führungsinformationssystemen. Ein methodenorientierter Ansatz. Wiesbaden 1999.
- Holten, R. (Framework): A Framework for Information Warehouse Development Processes. In: J. Becker; H. L. Grob; U. Müller-Funk; Klein, S.; Kuchen, H.; G. Vossen: Arbeitsberichte des Instituts für Wirtschaftsinformatik Nr. 67, Münster 1999.
- Holten, R.; Knackstedt, R.: Fachkonzeptuelle Modellierung von Führungsinformationssystemen am Beispiel eines filialisierenden Einzelhandelsunternehmens. In: E. J. Sinz et al. (Hrsg.), Tagungsband MobIS99. Erscheint im Oktober 1999.
- Holthuis, J.: Multidimensionale Datenstrukturen. Modellierung, Strukturkomponenten, Implementierungsaspekte. In: H. Mucksch, W. Behme (Hrsg.), Das Data Warehouse-Konzept. 3. Aufl. Wiesbaden 1998, S. 143-193.
- IEEE-830: Guide to Software Requirements Specification. ANSI/IEEE Std. 830, 1984.

- ISO/IEC: Information Technology. Information Resource Dictionary Systems (IRDS)-
Framework. ISO/IEC Intl. Standard, 1990.
- Jacobs, S.; Holten, R.: Goal Driven Business Modelling. Supporting Decision Making within
Information Systems Development. In: N. Comstock, C. Ellis, R. Kling, J. Mylopoulos, S.
Kaplan (Hrsg.): Conference on Organizational Computing Systems, August 13-16, 1995,
Milipatis, California, U.S.A. acm Press 1995, S. 96-105.
- Kambartel, F.: Stufe. In: Mittelstraß (Hrsg.): Enzyklopädie Philosophie und Wissenschaftsthe-
orie, Band 4. Stuttgart, Weimar 1996, S. 116.
- Lehmann, P.; Ellerau, P.: Implementierung eines Data Warehouse für die Versicherungsindust-
rie. HMD Theorie und Praxis der Wirtschaftsinformatik, 34 (1997) 195, S. 76-93.
- Lorenz, K. (Metasprache): Metasprache. In: J. Mittelstraß (Hrsg.): Enzyklopädie Philosophie
und Wissenschaftstheorie, Band 2. Stuttgart, Weimar 1995, S. 875.
- Lorenz, K. (Objektsprache): Objektsprache. In: J. Mittelstraß (Hrsg.): Enzyklopädie Philoso-
phie und Wissenschaftstheorie, Band 2. Stuttgart, Weimar 1995, S. 1054 f.
- Lorenz, K. (Semiotik): Semiotik. In: J. Mittelstraß (Hrsg.): Enzyklopädie Philosophie und
Wissenschaftstheorie, Band 3. Stuttgart, Weimar 1995, S. 781-786.
- McMenamin, S. M.; Palmer, J. F.: Essential System Analysis. Yourdon Press, Prentice Hall.
Englewood Cliffs, NY, 1984.
- Mucksch, H.; Behme, W.: Das Data Warehouse-Konzept als Basis einer unternehmensweiten
Informationslogistik. In: H. Mucksch, W. Behme (Hrsg.), Das Data Warehouse-Konzept.
3. Aufl. Wiesbaden 1998, S. 33-100.
- Nagl, M.: Software Technik. Methodisches Programmieren im Großen. Berlin, Heidelberg
1990.
- Nissen, H. W.; Jeusfeld, M.; Jarke, M.; Zemanek, G. V.; Huber, H.: Managing Multiple Requi-
rements Perspectives with Metamodels. IEEE Software, 13 (1996) 3, S. 37-48.
- Pohl, K.: Process-Centered Requirements Engineering. Taunton, Somerset 1996.
- Reichmann, T.: Controlling mit Kennzahlen und Managementberichten. 5. Aufl. München
1997.

- Schelp, J.: Konzeptionelle Modellierung mehrdimensionaler Datenstrukturen. In: P. Chameni, P. Gluchowski (Hrsg.), *Analytische Informationssysteme*. Berlin, Heidelberg 1998, S. 263-276.
- Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Wiesbaden 1998.
- Staehe, W. H.: Kennzahlen und Kennzahlensysteme. Ein Beitrag zur modernen Organisations- theorie. Dissertation, München 1967.
- Stöhr, T.; Müller, R.; Rahm, E.: An Integrative and Uniform Model for Metadata Management in Data Warehousing Environments. In.: S. Gatztiu, M. Jeusfeld, M. Staudt, Y-Vassiliou (Hrsg.): *Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'99)*. <http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-19/>. Abruf am 1999-07-12.
- Vossen, G.: *Datenbankmodelle, Datenbanksprachen und Datenbankmanagementsysteme*. 3. Aufl. München, Wien 1999.
- Wedekind, H.: Die Objekttypen-Methode beim Datenbankentwurf - dargestellt am Beispiel von Buchungs- und Abrechnungssystemen. *ZfB (Zeitschrift für Betriebswirtschaft)* 49 (1979) 5, S. 367-387.
- Wedekind, H.: Strukturveränderung im Rechnungswesen unter dem Einfluß der der Daten- banktechnologie. *ZfB* 50 (1980) 6, S. 662-677.
- Wedekind, H.: *Datenbanksysteme I. Eine konstruktive Einführung in die Datenverarbeitung in Wirtschaft und Verwaltung*. 2. Aufl., Mannheim et al. 1981.
- Wedekind, H.; Ortner, E.: Der Aufbau einer Datenbank für die Kostenrechnung. *Die Betriebs- wirtschaft* 37 (1977) 4, S. 533-542.
- Wietek, F.: Modelling Multidimensional Data in a Dataflow-Based Visual Data Analysis Environment. In: M. Jarke, A. Oberweis (Hrsg.): *CaiSE'99, LNCS 1626*. Berlin Heidel- berg 1999, S. 149-163.

Arbeitsberichte des Instituts für Wirtschaftsinformatik

- Nr. 1 Bolte, Ch., Kurbel, K., Moazzami, M., Pietsch, W.: Erfahrungen bei der Entwicklung eines Informationssystems auf RDBMS- und 4GL-Basis; Februar 1991.
- Nr. 2 Kurbel, K.: Das technologische Umfeld der Informationsverarbeitung - Ein subjektiver 'State of the Art'-Report über Hardware, Software und Paradigmen; März 1991.
- Nr. 3 Kurbel, K.: CA-Techniken und CIM; Mai 1991.
- Nr. 4 Nietsch, M., Nietsch, T., Rautenstrauch, C., Rinschede, M., Siedentopf, J.: Anforderungen mittelständischer Industriebetriebe an einen elektronischen Leitstand - Ergebnisse einer Untersuchung bei zwölf Unternehmen; Juli 1991.
- Nr. 5 Becker, J., Prischmann, M.: Konnektionistische Modelle - Grundlagen und Konzepte; September 1991.
- Nr. 6 Grob, H. L.: Ein produktivitätsorientierter Ansatz zur Evaluierung von Beratungserfolgen; September 1991.
- Nr. 7 Becker, J.: CIM und Logistik; Oktober 1991.
- Nr. 8 Burgholz, M., Kurbel, K., Nietsch, Th., Rautenstrauch, C.: Erfahrungen bei der Entwicklung und Portierung eines elektronischen Leitstands; Januar 1992.
- Nr. 9 Becker, J., Prischmann, M.: Anwendung konnektionistischer Systeme; Februar 1992.
- Nr. 10 Becker, J.: Computer Integrated Manufacturing aus Sicht der Betriebswirtschaftslehre und der Wirtschaftsinformatik; April 1992.
- Nr. 11 Kurbel, K., Dornhoff, P.: A System for Case-Based Effort Estimation for Software-Development Projects; Juli 1992.
- Nr. 12 Dornhoff, P.: Aufwandsplanung zur Unterstützung des Managements von Softwareentwicklungsprojekten; August 1992.
- Nr. 13 Eicker, S., Schnieder, T.: Reengineering; August 1992.
- Nr. 14 Erkelenz, F.: KVD2 - Ein integriertes wissensbasiertes Modul zur Bemessung von Krankenhausverweildauern - Problemstellung, Konzeption und Realisierung; Dezember 1992.
- Nr. 15 Horster, B., Schneider, B., Siedentopf, J.: Kriterien zur Auswahl konnektionistischer Verfahren für betriebliche Probleme; März 1993.
- Nr. 16 Jung, R.: Wirtschaftlichkeitsfaktoren beim integrationsorientierten Reengineering: Verteilungsarchitektur und Integrationsschritte aus ökonomischer Sicht; Juli 1993.
- Nr. 17 Miller, C., Weiland, R.: Der Übergang von proprietären zu offenen Systemen aus Sicht der Transaktionskostentheorie; Juli 1993.
- Nr. 18 Becker, J., Rosemann, M.: Design for Logistics - Ein Beispiel für die logistikgerechte Gestaltung des Computer Integrated Manufacturing; Juli 1993.
- Nr. 19 Becker, J., Rosemann, M.: Informationswirtschaftliche Integrationsschwerpunkte innerhalb der logistischen Subsysteme - Ein Beitrag zu einem produktionsübergreifenden Verständnis von CIM; Juli 1993.

- Nr. 20 Becker, J.: Neue Verfahren der entwurfs- und konstruktionsbegleitenden Kalkulation und ihre Grenzen in der praktischen Anwendung; Juli 1993.
- Nr. 21 Becker, K., Prischmann, M.: VESKONN - Prototypische Umsetzung eines modularen Konzepts zur Konstruktionsunterstützung mit konnektionistischen Methoden; November 1993
- Nr. 22 Schneider, B.: Neuronale Netze für betriebliche Anwendungen: Anwendungspotentiale und existierende Systeme; November 1993.
- Nr. 23 Nietsch, T., Rautenstrauch, C., Rehfeldt, M., Rosemann, M., Turowski, K.: Ansätze für die Verbesserung von PPS-Systemen durch Fuzzy-Logik; Dezember 1993.
- Nr. 24 Nietsch, M., Rinschede, M., Rautenstrauch, C.: Werkzeuggestützte Individualisierung des objektorientierten Leitstands ooL; Dezember 1993.
- Nr. 25 Meckenstock, A., Unland, R., Zimmer, D.: Flexible Unterstützung kooperativer Entwurfsumgebungen durch einen Transaktions-Baukasten; Dezember 1993.
- Nr. 26 Grob, H. L.: Computer Assisted Learning (CAL) durch Berechnungsexperimente; Januar 1994.
- Nr. 27 Kirn, St., Unland, R. (Hrsg.): Tagungsband zum Workshop "Unterstützung Organisatorischer Prozesse durch CSCW". In Kooperation mit GI-Fachausschuß 5.5 "Betriebliche Kommunikations- und Informationssysteme" und Arbeitskreis 5.5.1 "Computer Supported Cooperative Work", Westfälische Wilhelms-Universität Münster, 4.-5. November 1993
- Nr. 28 Kirn, St., Unland, R.: Zur Verbundintelligenz integrierter Mensch-Computer-Teams: Ein organisationstheoretischer Ansatz; März 1994.
- Nr. 29 Kirn, St., Unland, R.: Workflow Management mit kooperativen Softwaresystemen: State of the Art und Problemabriß; März 1994.
- Nr. 30 Unland, R.: Optimistic Concurrency Control Revisited; März 1994.
- Nr. 31 Unland, R.: Semantics-Based Locking: From Isolation to Cooperation; März 1994.
- Nr. 32 Meckenstock, A., Unland, R., Zimmer, D.: Controlling Cooperation and Recovery in Nested Transactions; März 1994.
- Nr. 33 Kurbel, K., Schnieder, T.: Integration Issues of Information Engineering Based I-CASE Tools; September 1994.
- Nr. 34 Unland, R.: TOPAZ: A Tool Kit for the Construction of Application Specific Transaction; November 1994.
- Nr. 35 Unland, R.: Organizational Intelligence and Negotiation Based DAI Systems - Theoretical Foundations and Experimental Results; November 1994.
- Nr. 36 Unland, R., Kirn, St., Wanka, U., O'Hare, G.M.P., Abbas, S.: AEGIS: AGENT ORIENTED ORGANISATIONS; Februar 1995.
- Nr. 37 Jung, R., Rimpler, A., Schnieder, T., Teubner, A.: Eine empirische Untersuchung von Kosteneinflußfaktoren bei integrationsorientierten Reengineering-Projekten; März 1995.
- Nr. 38 Kirn, St.: Organisatorische Flexibilität durch Workflow-Management-Systeme?; Juli 1995.
- Nr. 39 Kirn, St.: Cooperative Knowledge Processing: The Key Technology for Future Organizations; Juli 1995.

- Nr. 40 Kirn, St.: Organisational Intelligence and Distributed AI; Juli 1995.
- Nr. 41 Fischer, K., Kirn, St., Weinhard, Ch. (Hrsg.): Organisationsaspekte in Multiagentensystemen; September 1995.
- Nr. 42 Grob, H. L., Lange, W.: Zum Wandel des Berufsbildes bei Wirtschaftsinformatikern, Eine empirische Analyse auf der Basis von Stellenanzeigen, Oktober 1995.
- Nr. 43 Abu-Alwan, I., Schlagheck, B., Unland, R.: Evaluierung des objektorientierten Datenbankmanagementsystems ObjectStore, Dezember 1995.
- Nr. 44 Winter, R., Using Formalized Invariant Properties of an Extended Conceptual Model to Generate Reusable Consistency Control for Information Systems; Dezember 1995.
- Nr. 45 Winter, R., Design and Implementation of Derivation Rules in Information Systems; Februar 1996.
- Nr. 46 Becker, J.: Eine Architektur für Handelsinformationssysteme; März 1996.
- Nr. 47 Becker, J., Rosemann, M. (Hrsg.): Workflowmanagement - State-of-the-Art aus Sicht von Theorie und Praxis, Proceedings zum Workshop vom 10. April 1996; April 1996.
- Nr. 48 Rosemann, M., zur Mühlen, M.: Der Lösungsbeitrag von Metadatenmodellen beim Vergleich von Workflowmanagementsystemen; Juni 1996.
- Nr. 49 Rosemann, M., Denecke, Th., Püttmann, M.: Konzeption und prototypische Realisierung eines Informationssystems für das Prozeßmonitoring und -controlling; September 1996.
- Nr. 50 v. Uthmann, C., Turowski, K. unter Mitarbeit von Rehfeldt, M., Skall, M.: Workflow-basierte Geschäftsprozeßregelung als Konzept für das Management von Produktentwicklungsprozessen; November 1996.
- Nr. 51 Eicker, S., Jung, R., Nietsch, M., Winter, R.: Entwicklung eines Data Warehouse für das Produktionscontrolling: Konzepte und Erfahrungen; November 1996.
- Nr. 52 Becker, J., Rosemann, M., Schütte, R. (Hrsg.): Entwicklungsstand und Entwicklungsperspektiven der Referenzmodellierung, Proceedings zur Veranstaltung vom 10. März 1997; März 1997.
- Nr. 53 Loos, P.: Capture More Data Semantic Through The Expanded Entity-Relationship Model (PERM); Februar 1997.
- Nr. 54 Becker, J., Rosemann, M. (Hrsg.): Organisatorische und technische Aspekte beim Einsatz von Workflowmanagementsystemen. Proceedings zur Veranstaltung vom 10. April 1997; April 1997.
- Nr. 55 Holten, R., Knackstedt, R.: Führungsinformationssysteme - Historische Entwicklung und Konzeption; April 1997.
- Nr. 56 Holten, R.: Die drei Dimensionen des Inhaltsaspektes von Führungsinformationssystemen; April 1997.
- Nr. 57 Holten, R., Striemer, R., Weske, M.: Ansätze zur Entwicklung von Workflow-basierten Anwendungssystemen - Eine vergleichende Darstellung -, April 1997.
- Nr. 58 Kuchen, H.: Arbeitstagung Programmiersprachen, Tagungsband, Juli 1997.

- Nr. 59 Vering, O.: Berücksichtigung von Unschärfe in betrieblichen Informationssystemen - Einsatzfelder und Nutzenpotentiale am Beispiel der PPS, September 1997.
- Nr. 60 Schwegmann, A., Schlagheck, B.: Integration der Prozeßorientierung in das objektorientierte Paradigma: Klassenzuordnungsansatz vs. Prozeßklassenansatz, Dezember 1997.
- Nr. 61 Speck, M.: In Vorbereitung.
- Nr. 62 Wiese, J.: Ein Entscheidungsmodell für die Auswahl von Standardanwendungssoftware am Beispiel von Warenwirtschaftssystemen, März 1998.
- Nr. 63 Kuchen, H.: Workshop on Functional and Logic Programming, Proceedings, Juni 1998.
- Nr. 64 v. Uthmann, C.; Becker, J.; Brödner, P.; Maucher, I.; Rosemann, M. (Hrsg.): PPS meets Workflow. Proceedings zum Workshop vom 9. Juni 1998, Juni 1998.
- Nr. 65 Scheer, A.-W.; Rosemann, M.; Schütte, R. (Hrsg.): Integrationsmanagement, Januar 1999.
- Nr. 66 zur Mühlen, M.; Ehlers, L.: Internet – Technologie und Historie. Juni 1999.
- Nr. 67 Holten, R.: A Framework for Information Warehouse Development Processes. Mai 1999.
- Nr. 68 Holten, R.; Knackstedt, R.: Fachkonzeption von Führungsinformationssystemen – Instanziierung eines FIS-Metamodells am Beispiel eines Einzelhandelsunternehmens. Mai 1999.
- Nr. 69 Holten, R.: Semantische Spezifikation Dispositiver Informationssysteme. Juli 1999.