

Heft 96

Peter Loos

**Die Semantik eines erweiterten
Entity-Relationship-Modells und die
Überführung in SQL-Datenbanken**

November 1992

Zusammenfassung

Für die Datenmodellierung des Fachkonzepts eines Informationssystems werden benutzerorientierte Beschreibungsmethoden wie das Entity-Relationship-Modell eingesetzt. Diese Methoden sollen einerseits anwendungsnah und einfach zu handhaben sein, andererseits müssen sie so exakt sein, daß sie die gesamte Datensemantik enthalten, um eine eindeutige Überführung in Datenbanksysteme zu ermöglichen. Im folgenden wird dargestellt, welche Erweiterungen im Entity-Relationship-Modell notwendig sind, um diesem Anspruch des Fachkonzepts gerecht zu werden. Anschließend wird gezeigt, wie die Semantik in relationale Datenbanken, die dem SQL und SQL2-Standard genügen, überführt werden können.

1. Einleitung

Datenstrukturen sind neben den Funktionen und Abläufen eine wichtige Sicht auf betriebliche Informationssysteme. Sie gehören zu den langlebigsten Elementen eines Systems und können in der Regel unabhängig von einzelnen Teilabläufen entworfen werden¹. Gesamtheitlich betrachtete Datenstrukturen und daraus abgeleitete Datenbanken sind ein wichtiges Integrationsinstrumentarium.

Die Datensicht ist Bestandteil nahezu aller Architekturkonzepte² für den Aufbau von Informationssystemen wie bsw. ARIS oder CIM-OSA. Desweiteren wird nach Ebenen unterteilt, die die Abhängigkeit der Darstellung von der Nähe zur Informationstechnik beschreibt. So unterteilt Scheer in der ARIS-Architektur die Ebenen Fachkonzept, DV-Konzept und Implementierung³. Das Fachkonzept beschreibt die Semantik des Informationssystems aus anwendungsorientierter Sicht. Dabei spielen die DV-technischen Aspekte keine Rolle. Vielmehr werden Aufbau, Abläufe und Vorgänge des betrieblichen Geschehens mit formalen Beschreibungsmitteln dargestellt. Die Beschreibung des Fachkonzepts, auch Modellierung genannt, setzt den betrieblichen Sachverstand, aber keine speziellen DV-Kenntnisse voraus und stellt den eigentlich kreativen Prozeß bei dem Entwurf eines Informationssystems dar. Das DV-Konzept setzt die Anforderungen des Fachkonzepts in Bedingungen und Möglichkeiten der Informationstechnologien um. Eine Erweiterung des fachlichen Inhalts findet nicht statt. Das Implementierungskonzept setzt das DV-Konzept in reale Informationssysteme um. Die Transformationen in das DV-Konzept sowie in das Im-

1) vgl. Scheer (1990), S. 23.

plementierungskonzept sind rein technische Vorgänge, bei denen DV-technische Überlegungen wie der Einsatz von Datenbanken, Netzwerkprotokollen oder Programmiersprachen und Performancegesichtspunkte eine Rolle spielen. Dies erfordert andererseits, daß die Informationsmodelle des Fachkonzepts auch die gesamte betriebliche Semantik abdecken und beinhalten.

Für die Darstellung der Datensicht des Fachkonzepts hat sich als Beschreibungssprache sowohl in der Literatur als auch in der Praxis das Entity-Relationship-Modell durchgesetzt.

Bei der Umsetzung in Datenbanksysteme im Rahmen des DV-Konzepts werden heute hauptsächlich relationale Datenbanken eingesetzt. Die älteren hierarchischen Datenbanken und Netzwerk-Datenbanken verlieren zunehmend an Bedeutung. Obwohl für objektorientierte Datenbanken zwischenzeitlich ein allgemein anerkanntes Paradigma⁴ vorliegt, gibt es noch keine einheitliche Beschreibungssyntax. Auch haben objektorientierte Datenbanksysteme noch nicht die Reife für größere betriebliche Anwendungen erlangt.

Im folgenden soll aufgezeigt werden, welche Erweiterungen im Entity-Relationship-Modell notwendig sind, um dem Anspruch gerecht zu werden, die Datensemantik des abzubildenden Betriebsgeschehens abzudecken. Desweiteren wird dargestellt, wie ein solches Fachkonzept in ein DV-Konzept überführt werden kann, wobei dies anhand von SQL-Datenbanken gezeigt wird.

2. Datenstrukturen im Entity-Relationship-Modell

Das Entity-Relationship-Modell (ERM) hat sich seit seiner Einführung⁵ im Jahr 1976 als Datenstrukturbeschreibungssprache durchgesetzt und ist zwischenzeitlich um zahlreiche Konstrukte erweitert worden⁶. So sind insbesondere die Klassifizierung von Beziehungstypen durch Kardinalitäten und die Generalisierung in den meisten Modellen anzutreffen. Andere Modelle, wie die NIAM-Methode⁷ oder das SAM*-Modell⁸, besitzen interessante Erweiterungen bezüglich der semantischen Ausdruckskraft, z.B. für Abhängigkeiten zwischen Beziehungen. Aufbauend auf den bestehenden Methoden sollen im folgenden die

2) eine Übersicht über Architekturkonzepte ist zu finden in: Scheer (1991), S. 24 ff.

3) vgl. Scheer (1991), S. 17 ff.

4) vgl. Atkinson et al. (1989).

5) vgl. Chen (1976).

6) z.B. ECRM in: Elmasri, Weeldreyer, Hevner (1985); SERM in: Sinz (1987); ein Übersicht ist zu finden in: Loos, Datenstrukturierung ... (1992), S. 17ff.

7) vgl. Verheijen, van Bekkum (1982).

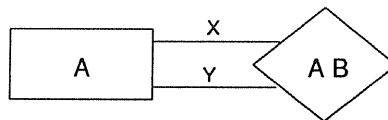
8) vgl. Su, Lo (1980).

wichtigsten und notwendigsten Erweiterungen vorgestellt werden. Sie sind Bestandteil der Modell-Erweiterung PERM⁹.

2.1 Binäre Beziehungstypen

Die Definition von Beziehungen zwischen zwei Objekten ist als Grundkonstrukt in jedem Modell vorhanden. Werden die Beziehungstypen durch die allgemein übliche (min,max)-Notation spezifiziert¹⁰, so können bei (max)-Werten von eins oder 'n' drei Beziehungstypen, bei zusätzlicher Unterscheidung der (min)-Werten von null oder eins insgesamt zehn Arten von Beziehungstypen unterschieden werden¹¹.

Bei rekursiven Beziehungen, d.h. Beziehungen zwischen zwei Entities eines gleichen Entitytyps, können insgesamt nur sieben Arten von Beziehungen unterschieden werden (vgl. Abb. 1). Rekursive Typen mit den Kardinalitätskombinationen (1,1) und (0,1), (0,1) und (1,n) sowie (1,1) und (1,n) sind bei einer endlichen Menge von Entities nicht abbildbar. Ein typisches Beispiel für rekursive Beziehungen sind Stücklisten oder Organigramme.



	X	Y
(11)	(0,1)	(0,1)
(12)	(1,1)	(1,1)
(13)	(0,1)	(0,n)
(14)	(1,1)	(0,n)
(15)	(0,n)	(0,n)
(16)	(0,n)	(1,n)
(17)	(1,n)	(1,n)

Abb. 1: Arten rekursiver, binärer Beziehungstypen

9) vgl. Loos, Datenstrukturierung ... (1992), S. 89 ff.

10) als Kardinalität oder Komplexität bezeichnet, vgl. Loos, Datenstrukturierung ... (1992), S. 25 f.

2.2 Mehrfachbeziehungstypen

Beziehungen können auch zwischen mehr als zwei Entities existieren. Sie werden als Mehrfachbeziehungstypen bezeichnet. Anhand eines Beispiels wird gezeigt, daß die übliche (min,max)-Notation zur eindeutigen Spezifizierung von Mehrfachbeziehungstypen nicht hinreichend ist.

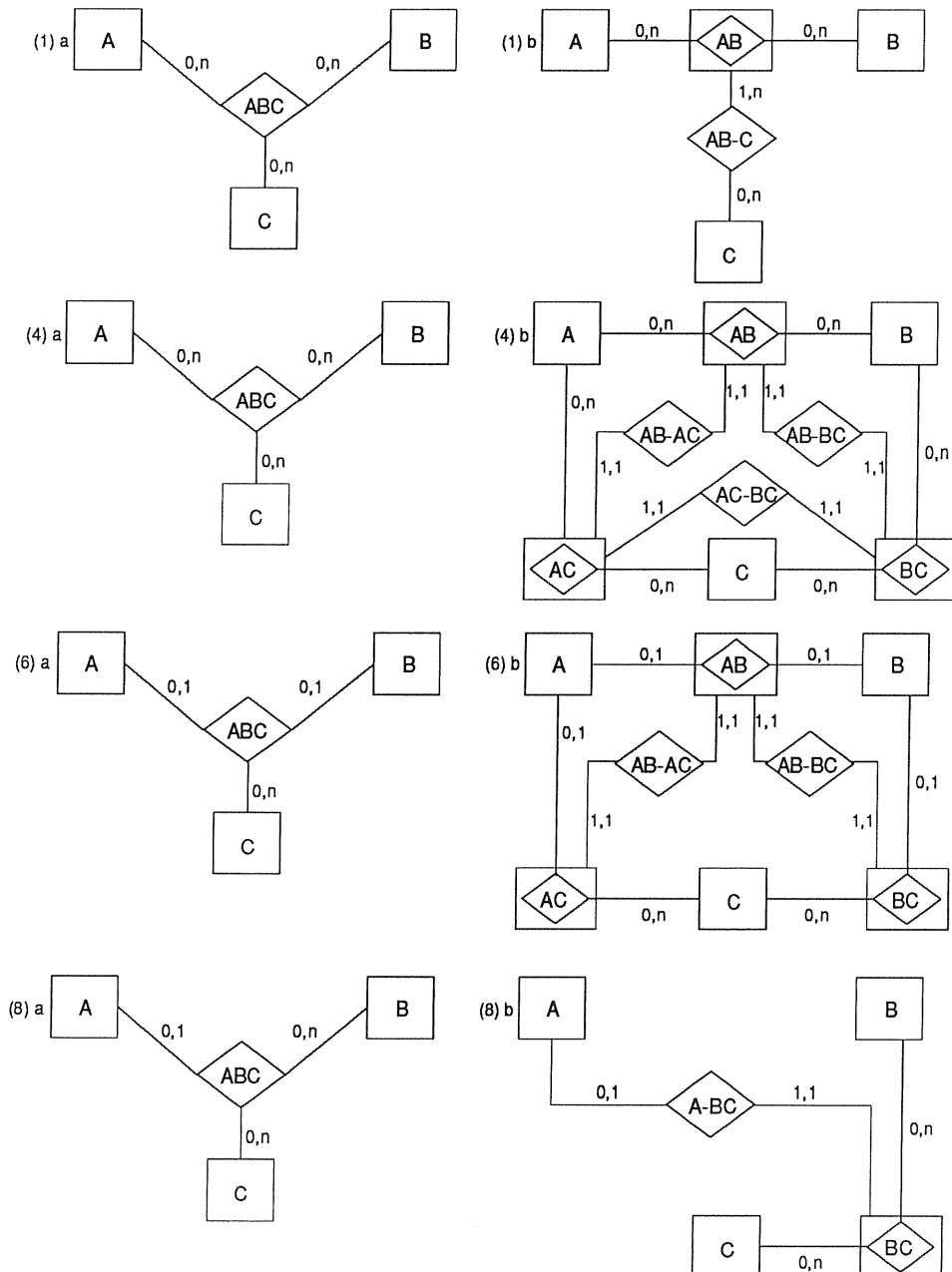


Abb. 2: Arten von Dreifachbeziehungstypen (Typ 1, 4, 6, 8)

11) die mögliche Anzahl stellt eine Kombination mit Wiederholung von n Elementen der Ordnung 2 dar, d.h. $Cw_2(2) = 3$ bzw. $Cw_4(2) = 10$.

Die Beziehung *arbeitet* verbinde die Entitytypen *Mitarbeiter*, *Abteilung* und *Projekt*. Ein Mitarbeiter kann in mehreren Abteilungen an unterschiedlichen Projekten arbeiten. Andererseits kann ein Mitarbeiter an einem Projekt in unterschiedlichen Abteilungen arbeiten. In einer Abteilung arbeiten mehrere Mitarbeiter an unterschiedlichen Projekten. Ein Projekt kann in mehreren Abteilungen von verschiedenen Mitarbeitern bearbeitet werden. Keine Entities der Typen *Mitarbeiter*, *Abteilung* und *Projekt* müssen zwingend eine Beziehung eingehen, deshalb haben die Kardinalitäten aller Typen einen (min)-Wert von null. Andererseits kann jedes einzelne Entity aller drei Typen mehrmals eine Beziehung eingehen, so daß jeweils ein 'n' als (max)-Wert anzugeben ist. Das Beispiel wird dahingehend geändert, daß ein Mitarbeiter zwar in mehreren Abteilungen, aber pro Abteilung nur in genau einem Projekt arbeiten kann. Die übrigen Aussagen bleiben bestehen. Für die Bestimmung der Kardinalitäten gilt, daß ein Mitarbeiter in unterschiedlichen Kombinationen von Abteilungen und Projekten arbeiten kann. Die Aussage trifft analog auch für die beiden anderen Entitytypen zu. Es zeigt sich, daß unterschiedliche Ausgangssituationen von Dreierbeziehungen zu gleichen Kardinalitätsangaben führen, d. h. die Angaben sind nicht eindeutig.

Die unterschiedlichen Möglichkeiten von Dreifachbeziehungen können mit Hilfe von funktionalen Abhängigkeiten¹² und einer Auflösung der Beziehung in mehrere binäre Beziehungen mit eindeutiger (min,max)-Notation dargestellt werden.

Mit Hilfe der funktionalen Abhängigkeiten kann gezeigt werden, welche Entities aus den verbundenen Entitytypen notwendigen für die Eindeutigkeit einer Beziehung sind. Sie sind damit Indikatoren für die (max)-Werte der Kardinalitätsangaben. Ausgehend von den Entitytypen A, B und C können acht unterschiedliche Typen¹³ differenziert werden¹⁴:

- (1) $a, b, c \rightarrow \emptyset$
- (2) $a, b \rightarrow c$
- (3) $a, b \rightarrow c \wedge a, c \rightarrow b$
- (4) $a, b \rightarrow c \wedge a, c \rightarrow b \wedge b, c \rightarrow a$
- (5) $a \rightarrow b, c$
- (6) $a \rightarrow b, c \wedge b \rightarrow a, c$
- (7) $a \rightarrow b, c \wedge b \rightarrow a, c \wedge c \rightarrow a, b$
- (8) $a \rightarrow b, c \wedge b, c \rightarrow a$

Das Ausgangsbeispiel der Beziehung *arbeitet* entspricht dem Typ (1), das geänderte Beispiel dem Typ (2). Abbildung 2 zeigt vier der acht möglichen Dreifachbeziehungstypen und die jeweilig adäquate Auflösung in binäre Beziehungstypen. Die Auflösung de-

12) vgl. Date (1981).

13) Im Gegensatz dazu gibt es vier Möglichkeiten bei konventioneller Kardinalitätsangabe ($Cw_2^{(3)} = 4$).

monstriert deutlich die Notwendigkeit, in Rahmen der Modellierung Mehrfachbeziehungstypen zuzulassen. Es wird vorgeschlagen, Mehrfachbeziehungstypen im ERM durch Angabe der funktionalen Abhängigkeiten und der Kardinalitäten exakt zu formulieren.

2.3 Generalisierung

Die Generalisierung gestattet, verschiedene Entitytypen (Subtypen) zu einem neuen Entitytyp (Supertyp) zu aggregieren¹⁵. Dies ist bsw. bei gemeinsamen Attributen oder Beziehungen sinnvoll. Bezüglich des Verhältnisses der Subtypen zueinander sind disjunkte und nicht-disjunkte Submengen zu unterscheiden. Bezüglich des Verhältnisses zwischen Supertyp und Subtypen sind vollständige und nicht-vollständige Generalisierungen zu unterscheiden. Bei einer vollständigen Generalisierung ist jedes Entity des Supertyps auch Element mindestens eines Subtyps. Durch die Kombination der beiden Charakteristiken ergeben sich vier Typen. Es wird vorgeschlagen, die Typen analog zur (min,max)-Notation in der graphischen Darstellung zu kennzeichnen. Eine Kennzeichnung von (0,x) bedeutet nicht-vollständig, (1,x) bedeutet vollständig, (x,1) bedeutet disjunkt und (x,n) bedeutet nicht-disjunkt.

2.4 Semantische Integritätsbedingung

Semantik, die durch die bisherigen Konstrukte nicht abgedeckt wird, muß durch explizite Integritätsbedingungen formulierbar sein. Im folgenden sollen notwendige Integritätsbedingungen vorgestellt werden¹⁶, wobei eine Beschränkung auf statische Bedingungen erfolgt. Transitionale und dynamische Integritätsbedingungen¹⁷ werden nicht behandelt, da sie eher dem Ablaufsteuerungsmodell als dem Datenmodell zuzuordnen sind.

Die objekttypinternen Integritätsbedingungen können auf einzelne Entities sowie auf Teilmengen eines Entitytyps wirken. Integritätsbedingungen eines einzelnen Entities sind Domänen, abgeleitete Attribute (das Alter läßt sich aus dem Geburtsdatum und dem aktuellen Datum ableiten), bedingte Attribute (der Wert 'ledig' des Attributs *Familienstand* schließt die Werte '3', '4' und '5' des Attributs *Steuerklasse* aus) oder wechselseitig abhängige Attribute (das Attribut *Starttermin* soll immer einen kleineren Wert aufweisen als das Attribut *Endtermin*). Integritätsbedingungen auf Mengen von Entities können sich auf eine

14) $a \rightarrow b$ bedeutet, daß b von a funktional abhängig ist

15) vgl. Smith, Smith (1977). Aus umgekehrter Sicht wird die Generalisierung auch Spezialisierung genannt.

16) Ein genaue Beschreibung ist zu finden in: Loos, Datenstrukturierung ... (1992), S. 73 ff.

17) vgl. Lipeck (1989).

maximale oder minimale Anzahl von Entities¹⁸ oder auf Summen- oder Durchschnittswerte beziehen.

Beziehungstypabhängige Integritätsbedingungen beschreiben Bedingungen zwischen Entities unterschiedlicher Typen. Diese Bedingungen können einzelne Beziehungen oder mehrere Beziehungen betreffen. Bedingungen, die sich auf einen Beziehungstyp beziehen, sind neben der Typspezifikation mit Hilfe von Kardinalitäten und funktionalen Abhängigkeiten die beziehungsabgeleiteten Attribute (das Attribut *Mitarbeiteranzahl* im Entitytyp *Abteilung* läßt sich aus der Anzahl der Beziehungen zum Typ *Mitarbeiter* ableiten), die objektausprägungsabhängigen Beziehungen (die Möglichkeit einer Beziehung eines Entitys hängt von der Ausprägungen bestimmter Attribut ab, z.B. darf ein Entity *Teil* nur Stücklistenbeziehungen eingehen, wenn es die Ausprägung 'Baugruppe' oder 'Endprodukt' hat, eine Teileverwendungsbeziehung ist nur bei der Ausprägung 'Rohteil' oder 'Baugruppe' sinnvoll¹⁹) sowie die objektkombinationsabhängigen Beziehungen (die Möglichkeit einer Beziehung zweier oder mehrerer Entities hängt von der Kombination ihrer Ausprägungen ab, z.B. soll in einer Mitarbeiterhierarchie sichergestellt werden, daß der Vorgesetzte immer ein höheres Dienstalter aufweist als der Weisungsempfänger²⁰).

Bedingungen, die mehrere Beziehungen eines Entitytyps betreffen, können danach unterschieden werden, ob direkte Beziehungen (einfache Tiefe) oder indirekte Beziehungen betroffen sind. Bei Bedingungen einfacher Tiefe können sich Beziehungen gegenseitig ausschließen (z.B. kann ein Mitarbeiter entweder Projekten oder Fertigungsbereichen zugeordnet werden²¹) oder Beziehungen können von einander abhängig sein (z.B. wenn einem Auftrag ein Werkzeug zugeordnet ist, muß dem Auftrag auch eine Maschine zugeordnet sein²²).

Beziehungstypabhängige Bedingungen können auch auf mehrere Beziehungstypen, die einen Pfad über mehrere Entitytypen aufbauen, wirken. Bei den Beziehungspfadbefindungen werden Abhängigkeiten oder Exklusivitäten bezüglich der Verbindung zweier Entitytypen über zwei Beziehungspfade definiert. Abbildung 3 zeigt ein entsprechendes Stücklisten-Arbeitsplanbeispiel²³. Ein Arbeitsgang ist immer genau einem Arbeitsplan zugeordnet, der seinerseits mehreren Teilen zugeordnet sein kann. In einem Arbeitsgang können mehrere Teile, die als Stücklistenkomponenten für die Fertigung des Zielteils not-

18) auch als 'absolute cardinality constraint' bezeichnet, vgl. Lenzerini, Santucci (1983).

19) Eine Darstellung des Sachverhalt durch Spezialisierung/Generalisierung bietet keine befriedigende Lösung, da dies zu mehreren Beziehungstypen mit gleicher Bedeutung führt.

20) Hier ist eine Spezialisierung nicht möglich, da für ein Entity Mitarbeiter keine generelle Eigenschaft Vorgesetzter oder Weisungsempfänger definierbar ist.

21) Der Wirkungsbereich der Exklusivität kann weiter unterteilt werden. Nur bestimmte Arten der Exklusivität können auch durch Spezialisierung modelliert werden.

22) Je nach Art der Abhängigkeiten kann den Sachverhalt teilweise in Mehrfachbeziehungen abgebildet werden.

wendig sind, montiert werden. Dies wird durch den Beziehungstyp *AGKomp* zwischen den Typen *Arbeitsgang* und *Struktur* dargestellt, wobei die Kante zwischen Struktur und Teil mit der Rolle *UT* auf die betreffenden Komponenten verweist. Gleichzeitig muß aber auch gelten, daß die Kante mit der Rolle *OT* auf das herzustellende Teil verweist, für das der Arbeitsplan mit dem Arbeitsgang gilt. Dies bedeutet, daß der Beziehungspfad zwischen einem Entity des Typs *Arbeitsgang* und einem Entity des Typs *Teil* über die Kanten *Arbeitsgang-AGKomp-Struktur-OT-Teil* auch einen Beziehungspfad der beiden Entities über die Kanten *Arbeitsgang-Arbeitsplan-APLZuo-Teil* voraussetzt.

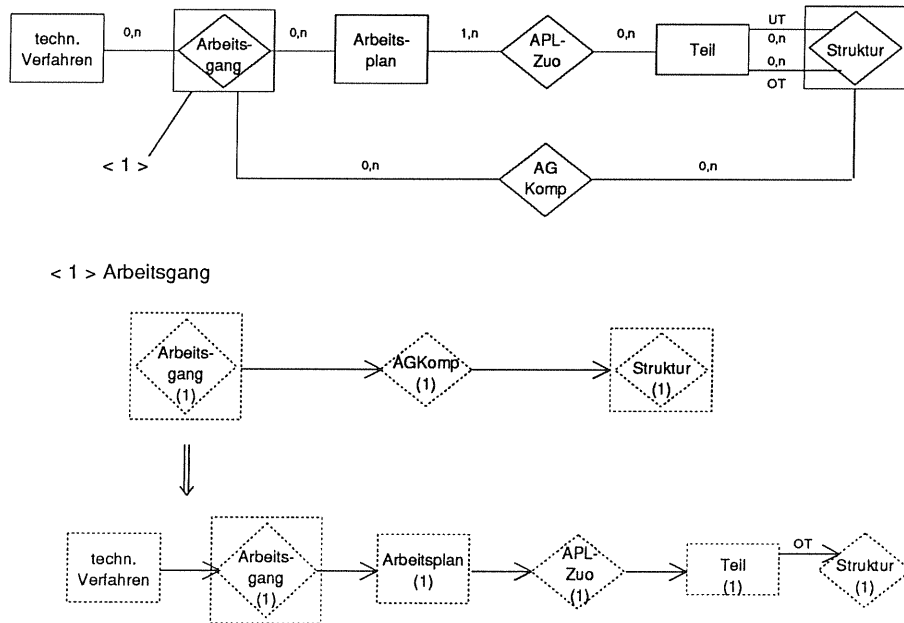


Abb. 3: Beispiel mit abhängigen Beziehungspfaden

Die beziehungstypabhängigen Integritätsbedingungen lassen sich durch graphische Darstellungen auf Entityebene formulieren. Diese Darstellungen lehnen sich an das Occurrence Structure Concept²⁴ an und werden als Relationship-Constraint-Diagramme (RC-Diagramm) bezeichnet. Dabei werden die Bedingungen an einzelnen Objekten, d.h. an Entities und deren Beziehungen verdeutlicht. Abbildung 3 zeigt neben der üblichen ERM-Darstellung ein RC-Diagramm, auf das aus dem ER-Diagramm durch die Kennzeichnung *<1>* verwiesen wird.

Ein Entitytyp kann über einen Beziehungspfad mit sich selbst verbunden sein. Bei dem kürzesten Pfad geht der Typ direkt mit sich selbst eine Beziehung ein, z. B. bei der Stückliste. Es ist häufig sinnvoll, einen Zyklus auszuschließen, so daß ein Entity weder direkt noch rekursiv über beliebig viele Stufen eine Beziehung mit sich selbst eingehen kann. Bei dem Stücklistenbeispiel bedeutet dies, daß ein Teil nicht als Komponente in sich selbst

23) Das Beispiel ist vereinfacht entnommen aus: Scheer (1990), S. 165.

24) vgl. Tabourier, Nanci (1983).

enthalten sein kann, was zumindest in der Stückgutproduktion sinnvollerweise auszuschließen ist. Diese Bedingung wird als Rekursionsbedingung bezeichnet.

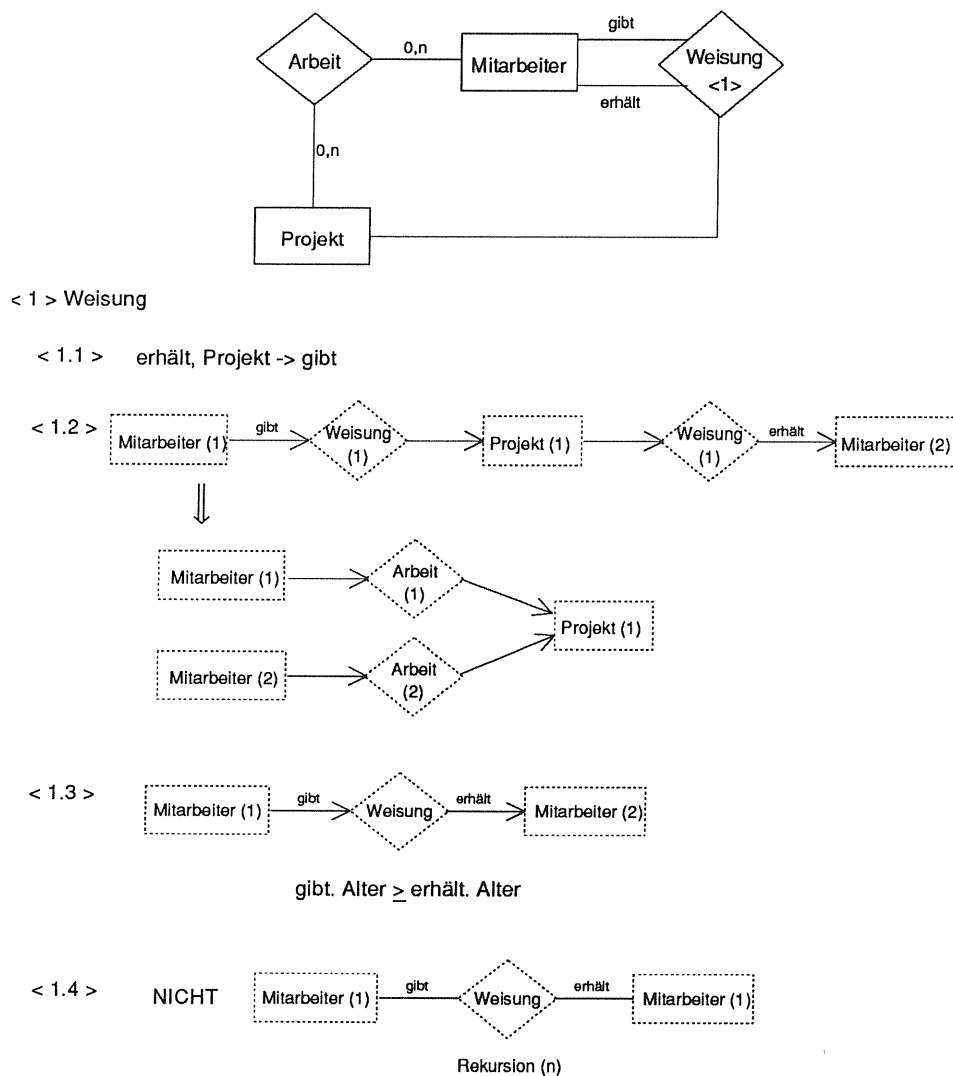


Abb. 4: Integritätsbedingungen eines Mitarbeiter-Projekt-Beispiels

Abbildung 4 enthält ein Beispiel mit RC-Diagramm, bei dem eine Dreifachbeziehung über funktionale Abhängigkeiten spezifiziert ist (Bedingung <1.1> besagt, daß ein Mitarbeiter in einem bestimmten Projekt nur von einem Vorgesetzten Weisungen erhalten kann), mit Hilfe der Beziehungspfadbedingung sichergestellt wird, daß zwei Mitarbeiter in einer Weisungshierarchie eines Projektes auch jeweils in dem Projekt arbeiten (Bedingung <1.2>), über eine objektkombinationsabhängige Bedingung die Weisungshierarchie mit den Alter korrespondiert (Bedingung <1.3>) und über eine Rekursionsbedingung Weisungen eines Mitarbeiters an sich selbst, unabhängig von der Anzahl der zwischengeschalteten Mitarbeiter, ausgeschlossen wird (Bedingung <1.4>).

Abbildung 5 zeigt einen zusammenfassenden Überblick über die semantischen Integritätsbedingungen.

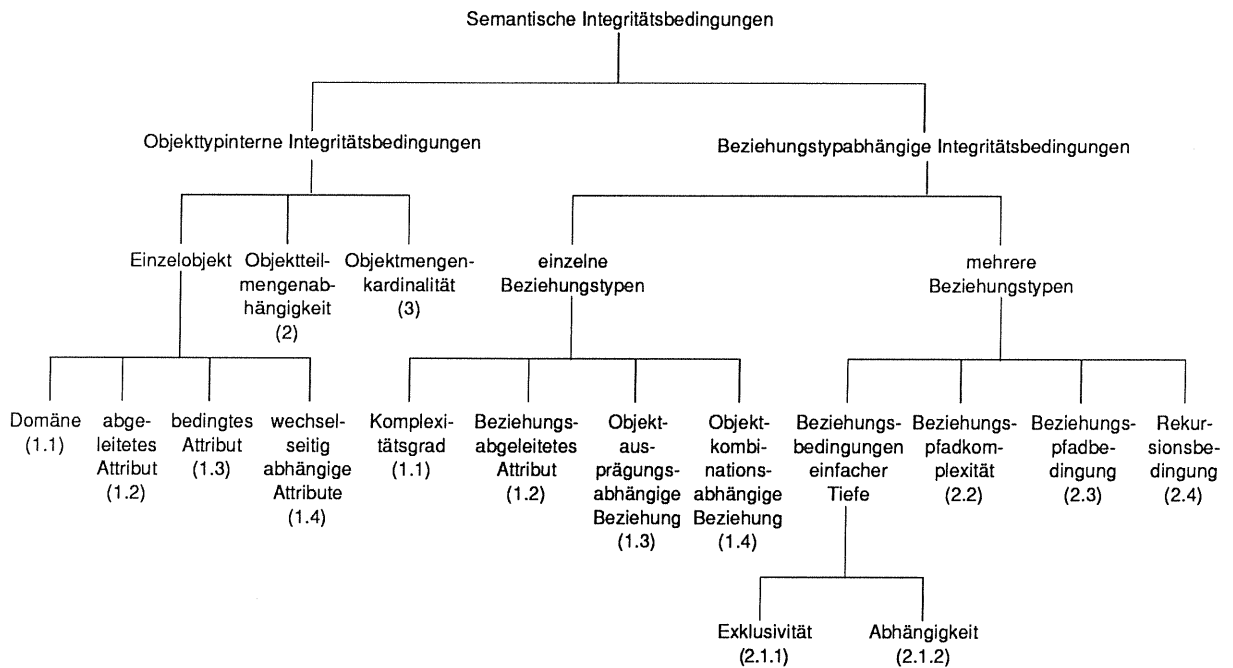


Abb. 5: Zusammenfassung der semantischen Integritätsbedingungen

3. Überführung in SQL-Datenbanken

Die Datenbanksprache SQL hat seit ihrer ersten Normung eine weite Verbreitung und Akzeptanz gefunden. Nahezu alle am Markt relevanten relationalen Datenbankmanagementsysteme bieten eine SQL-Schnittstelle an. SQL wurde von ANSI 1986 standardisiert und 1987 als ISO-Norm übernommen. Dieser Standard, der 1989 um Integritätsregeln erweitert wurde²⁵, ist die derzeit gültige Norm. Die durch die Arbeitsgruppe Datenbanken der ISO/IEC-Organisation durchgeführte Weiterentwicklung ist in der Fachliteratur als SQL2 bekannt und liegt als Normungsentwurf vor²⁶. SQL2 ist bis auf wenige Ausnahmen aufwärtskompatibel zu der Norm von 1989 und stellt unter anderem neue Konstrukte für die Formulierung von Integritätsbedingungen bereit²⁷.

25) vgl. ISO (1989).

26) vgl. ISO (1991).

27) vgl. Loos, Was ... (1992).

3.1 Repräsentationskonzepte von SQL-Datenbanken

SQL-Datenbanken nach der Norm von 1989 weisen neben Datentypen und NULL-Werten die Repräsentationskonzepte UNIQUE- bzw. PRIMARY KEY, FOREIGN KEY und CHECK auf²⁸. In dem SQL2-Standard wurden darüberhinaus u.a. ein Domänenkonzept, ein erweitertes FOREIGN KEY-Konzept und ein wesentlich mächtigeres CHECK-Konzept²⁹ aufgenommen.

3.2 Abbildung von Beziehungstypen

Beziehungstypen werden in Abhängigkeit ihrer Ausprägung in eigene Relationen überführt oder in eine Relation eines beteiligten Entitytyps eingeschlossen. Die Beziehungen lassen sich mit der FOREIGN KEY-Angabe formulieren. Da aber die referenzierten Fremdschlüssel immer eindeutig (UNIQUE) sein müssen, kann bei einer Kardinalität von (1,n) der (min)-Wert eins nicht formuliert werden. Z.B. soll in einer Auftragskopftabelle sichergestellt werden, daß mindestens eine zugehörige Auftragsposition in einer Positionentabelle existiert. Diese Bedingung kann nicht in SQL der Norm von 1989 formuliert werden. Mit der erweiterten CHECK-Bedingung von SQL2, die mit Hilfe von Subqueries auch interrelationale Formulierungen zuläßt, können auch (1,n)-Kardinalitäten abgedeckt werden, z.B.:

```
CREATE TABLE AufKopf (AID AID_Domain, ...,
  PRIMARY KEY (AID),
  CHECK ( AID IN ( SELECT AID FROM AufPos ) ) INITIALLY DEFERRED )

CREATE TABLE AufPos (AID AID_Domain, Pos Pos_Domain, ...,
  PRIMARY KEY (AID, Pos), FOREIGN KEY (AID) REFERENCES AufKopf (AID) )
```

Aufgrund der verzögerten Überprüfung der Integritätsbedingung in der Auftragskopftabelle (INITIALLY DEFERRED) wird ermöglicht, zuerst einen Auftragskopf ohne Positionen anzulegen. Anschließend können Position eingefügt werden. Spätestens am Ende der Transaktion (COMMIT) muß mindestens eine Position vorhanden sein.

Mehrfachbeziehungen lassen sich mit Hilfe der funktionalen Abhängigkeiten direkt in Relationen überführen. Dabei muß jeder determinierende Teil der funktionalen Abhängigkeiten mit Hilfe von UNIQUE eindeutig formuliert werden. Eine funktionale Abhängigkeit wird Primärschlüssel. Beziehungstyp (8) aus Abbildung 2 ergibt:

28) vgl. Loos (1991).

29) CHECK's können über die ASSERTION-Anweisung auch unabhängig von einer Table definiert werden.

```
CREATE TABLE ABC (A A_Domain, B B_Domain, C C_Domain, ...,
  PRIMARY KEY (A), UNIQUE (B, C), FOREIGN KEY (A) REFERENCES A (A),
  FOREIGN KEY (B) REFERENCES B (B), FOREIGN KEY (C) REFERENCES C (C) )
```

3.3 Generalisierung

Werden bei Generalisierungen sowohl der Supertyp als auch die Subtypen in eigene Relationen überführt, so kann mit Hilfe der FOREIGN KEY-Anweisung sichergestellt werden, daß jedes Entity eines Subtyps auch ein zugehöriges Entity im Supertyp besitzt. Vollständigkeit und Disjunktion können in SQL2 mit Hilfe von CHECK-Bedingungen formuliert werden, z.B.:

```
CREATE TABLE Supertyp (SID SID_Domain, ...,
  PRIMARY KEY (SID), CHECK ( SID IN (SELECT SID FROM Subtyp1)
  OR SID IN (SELECT SID FROM Subtyp2) )

CREATE TABLE Subtyp1 (SID SID_Domain, ...,
  PRIMARY KEY (SID), FOREIGN KEY SID REFERENCES Supertyp (SID),
  CHECK ( SID NOT IN ( SELECT SID FROM Subtyp2) ) )

CREATE TABLE Subtyp2 (SID SID_Domain, ...,
  PRIMARY KEY (SID), FOREIGN KEY SID REFERENCES Supertyp (SID),
  CHECK ( SID NOT IN ( SELECT SID FROM Subtyp1) ) )
```

3.4 Semantische Integritätsbedingungen

Da in der Norm von 1989 CHECK-Bedingungen nur intrarelativ auf das auslösende Tupel verweisen können, sind prinzipiell nur Einzelobjektbedingungen formulierbar. Bedingte Attribute können allerdings aufgrund eines fehlenden IF-Operators nicht abgebildet werden. Im Gegensatz dazu können aufgrund der erweiterten CHECK-Bedingungen in SQL2 nahezu alle Integritätsbedingungen abgebildet werden. Damit sind die Möglichkeiten der CHECK-Bedingung den produktspezifischen Triggererweiterungen einiger DBMS-Hersteller in der semantischen Ausdruckskraft ähnlich. Die Beziehungspfadbedingung aus Abbildung 3 könnte bsw. wie folgt formuliert werden³⁰:

```
CREATE ASSERTION Beziehungspfad_1 CHECK (
  ( SELECT OberTeilID, ArbeitsplanID FROM AG-Komp) IN
  ( SELECT TeilID, ArbeitsplanID FROM APL-Zuo) )
```

Schwierigkeiten gibt es bei der Abbildung der Rekursionsbedingung, da auch in SQL2 keine rekursiven Formulierungen möglich sind.

30) Es wird davon ausgegangen, daß alle Entity- und Beziehungstypen in Relationen überführt werden, die Relationen der Beziehungstypen kanonische Schlüssel besitzen und die Fremdschlüsselreferenzen sichergestellt sind.

Schlußbetrachtung

Es wurde gezeigt, welche Konzepte für die Repräsentation von Datenstrukturen im Fachkonzept notwendig sind, wobei die Konzepte als Mindestanforderung zu verstehen sind. Sie lassen sich als Erweiterung in das Entity-Relationship-Modell einbringen. Die aufgeführten Beispiele illustrieren ebenso wie die Mächtigkeit von SQL2-Datenbanken die Notwendigkeit der aufgeführten semantischen Konzepte. Wurde bisher die fehlende semantische Ausdruckskraft der Datenmodelle damit entschuldigt, daß diese Informationen doch nicht in Datenbanken abgelegt werden können, sondern im Programmcode als Funktionen beschrieben werden müssen, kann dieses Argument für SQL2-Datenbanken stark entkräftet werden. Es ist zu wünschen, daß möglichst schnell entsprechende Datenbankmanagementsysteme zur Verfügung stehen, was automatisch den Einsatz von exakteren Modellierungsmethoden nach sich zieht.

Literaturverzeichnis

- Atkinson, M.; Bancilhon, F.; De Witt, D.; Dittrich, K. R.; Maier, D.; Zdonik, S.: The Object-Oriented Database System Manifesto, Rapport Technique, Altair 1989.
- Chen, P. P.: The Entity-Relationship Model - Toward a Unified View of Data, in: ACM Transactions on Database Systems, 1(1976), S. 9 - 36.
- Date, C. J.: An Introduction to Database Systems Reading, Vol. I, Massachusetts, 1981.
- Elmasri, R.; Weeldreyer, J.; Hevner, A.: The Category Concept: An Extension to the Entity-Relationship Approach, in: Data & Knowledge Engineering, 1(1985), S. 75 - 116.
- ISO/IEC 9075, JTC1/SC21/WG3, Database language SQL with integrity enhancement, 1989.
- ISO/TEC DIS 9075, JTC1/SC21/WG3, Database Language SQL2, 1991.
- Lenzerini, M.; Santucci, G.: Cardinality Constraints in the Entity-Relationship Model, in: Davis C. G. et al. (Hrsg.), Entity-Relationship Approach to Software Engineering, Amsterdam-New York-Oxford 1983, S. 529 - 549.
- Lipeck, U. W.: Dynamische Integrität von Datenbanken - Grundlagen der Spezifikation und Überwachung, Berlin-Heidelberg-New York-Tokio 1989.
- Loos, P.: Datenstrukturierung in der Fertigung, München-Wien 1992.
- Loos, P.: Repräsentation erweiterter Entity-Relationship-Strukturen im Relationenmodell, in: Himmelmann, G. W. et al. (Hrsg.), Oracle in Theorie und Praxis, Stuttgart 1991, S. 113 - 132.
- Loos, P.: Was bringt SQL2?, in: Scheer, A.-W. (Hrsg.), Datenbanken 1992 - Praxis und Tendenzen relationaler Datenbanken (Tagungsband, Saarbrücken, 15.-16. Juni 1992), S. 130 - 141.
- Scheer, A.-W.: Architektur integrierter Informationssysteme - Grundlagen der Unternehmensmodellierung, Berlin-Heidelberg-New York-Tokio 1991.

- Scheer, A.-W.: Wirtschaftsinformatik - Informationssysteme im Industriebetrieb, 3. Auflage, Berlin-Heidelberg-New York-Tokio 1990.
- Sinz, E. J.: Datenmodellierung betrieblicher Probleme und ihre Unterstützung durch ein wissenschaftliches Entwicklungssystem, Habilitationsschrift Universität Regensburg 1987.
- Smith, J. M.; Smith, D. C.: Database Abstractions: Aggregation and Generalization, in: ACM Transaction on Database Systems, 2(1977)2, S. 105 - 433.
- Su, S. Y. W.; Lo, D. H.: A Semantic Association Model for Conceptual Database Design, in: Chen, P. P. (Hrsg.), Entity-Relationship Approach to System Analysis and Design, Amsterdam-New York-Oxford 1980, S. 169 - 192.
- Tabourier, Y.; Nanci, D.: The Occurrence Structure Concept: An Approach to Structural Integrity Constraints in the Entity-Relationship Model, in: Chen, P. P. (Hrsg.), Entity-Relationship Approach to Information Modeling and Analysis, Amsterdam-New York-Oxford 1983, S. 73 - 108.
- Verheijen, G. M. A.; van Bekkum, J.: NIAM: An Information Analysis Method, in: Olle, T. W. et al. (Hrsg.), Information Systems Design Methodologies - A Comparative Review, Amsterdam-New York-Oxford 1982, S. 537 - 589.

Die Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi) im Institut für empirische Wirtschaftsforschung an der Universität des Saarlandes erscheinen in unregelmäßiger Folge.

* Die Hefte 1 - 31 werden nicht mehr verlegt.

- Heft 32: A.-W. Scheer: Einfluß neuer Informationstechnologien auf Methoden und Konzepte der Unternehmensplanung, März 1982, Vortrag anlässlich des Anwendergespräches "Unternehmensplanung und Steuerung in den 80er Jahren in Hamburg vom 24. - 25.11.1981
- Heft 33: A.-W. Scheer: Dispositio- und Bestellwesen als Baustein zu integrierten Warenwirtschaftssystemen, März 1982, Vortrag anlässlich des gdi-Seminars "Integrierte Warenwirtschafts-Systeme" in Zürich vom 10. - 12. Dezember 1981
- Heft 34: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS - Ein Ansatz zur Entwicklung prüfungsgerechter Software-Systeme, Mai 1982
- Heft 35: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Konzept einer computergestützten Prüfungsumgebung, Juli 1982
- Heft 36: A.-W. Scheer: Rationalisierungserfolge durch Einsatz der EDV - Ziel und Wirklichkeit, August 1982, Vortrag anlässlich der 3. Saarbrücker Arbeitstagung "Rationalisierung" in Saarbrücken vom 04. - 06. 10.1982
- Heft 37: A.-W. Scheer: DV-gestützte Planungs- und Informationssysteme im Produktionsbereich, September 1982
- Heft 38: A.-W. Scheer: Interaktive Methodenbanken: Benutzerfreundliche Datenanalyse in der Marktforschung, Mai 1983
- Heft 39: A.-W. Scheer: Personal Computing - EDV-Einsatz in Fachabteilungen, Juni 1983
- Heft 40: A.-W. Scheer: Strategische Entscheidungen bei der Gestaltung EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anlässlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 41: H. Krcmar: Schnittstellenprobleme EDV-gestützter Systeme des Rechnungswesens, August 1983, Vortrag anlässlich der 4. Saarbrücker Arbeitstagung "Rechnungswesen und EDV" in Saarbrücken vom 26. - 28.09.1983
- Heft 42: A.-W. Scheer: Factory of the Future, Vorträge im Fachausschuß "Informatik in Produktion und Materialwirtschaft" der Gesellschaft für Informatik e. V., Dezember 1983
- Heft 43: A.-W. Scheer: Einführungsstrategie für ein betriebliches Personal-Computer-Konzept, März 1984
- Heft 44: A.-W. Scheer: Schnittstellen zwischen betriebswirtschaftlicher und technische Datenverarbeitung in der Fabrik der Zukunft, Juli 1984
- Heft 45: J. Ahlers, W. Emmerich, H. Krcmar, A. Pocsay, A.-W. Scheer, D. Siebert: EPSOS-D, Ein Werkzeug zur Messung der Qualität von Software-Systemen, August 1984
- Heft 46: H. Krcmar: Die Gestaltung von Computer am-Arbeitsplatz-Systemen - ablauforientierte Planung durch Simulation, August 1984

- Heft 47: A.-W. Scheer: Integration des Personal Computers in EDV-Systeme zur Kostenrechnung, August 1984
- Heft 48: A.-W. Scheer: Kriterien für die Aufgabenverteilung in Mikro-Mainframe Anwendungssystemen, April 1985
- Heft 49: A.-W. Scheer: Wirtschaftlichkeitsfaktoren EDV-orientierter betriebswirtschaftlicher Problemlösungen, Juni 1985
- Heft 50: A.-W. Scheer: Konstruktionsbegleitende Kalkulation in CIM-Systemen, August 1985
- Heft 51: A.-W. Scheer: Strategie zur Entwicklung eines CIM-Konzeptes - Organisatorische Entscheidungen bei der CIM-Implementierung, Mai 1986
- Heft 52: P. Loos, T. Ruffing: Verteilte Produktionsplanung und -steuerung unter Einsatz von Mikrocomputern, Juni 1986
- Heft 53: A.-W. Scheer: Neue Architektur für EDV-Systeme zur Produktionsplanung und -steuerung, Juli 1986
- Heft 54: U. Leismann, E. Sick: Konzeption eines Bildschirmtext-gestützten Warenwirtschaftssystems zur Kommunikation in verzweigten Handelsunternehmen, August 1986
- Heft 55: D. Steinmann: Expertensysteme (ES) in der Produktionsplanung und -steuerung (PPS) unter CIM-Aspekten, November 1987, Vortrag anlässlich der Fachtagung "Expertensysteme in der Produktion" am 16. und 17.11.1987 in München
- Heft 56: A.-W. Scheer: Enterprise wide Data Model (EDM) as a Basis for Integrated Information Systems, Juli 1988
- Heft 57: A.-W. Scheer: Present Trends of the CIM Implementation (A qualitative Survey) Juli 1988
- Heft 58: A.-W. Scheer: CIM in den USA - Stand der Forschung, Entwicklung und Anwendung, November 1988
- Heft 59: R. Herterich, M. Zell: Interaktive Fertigungssteuerung teilautonomer Bereiche, November 1988
- Heft 60: A.-W. Scheer, W. Kraemer: Konzeption und Realisierung eines Expertenunterstützungssystems im Controlling, Januar 1989
- Heft 61: A.-W. Scheer, G. Keller, R. Bartels: Organisatorische Konsequenzen des Einsatzes von Computer Aided Design (CAD) im Rahmen von CIM, Januar 1989
- Heft 62: M. Zell, A.-W. Scheer: Simulation als Entscheidungsunterstützungsinstrument in CIM, September 1989
- Heft 63: A.-W. Scheer: Unternehmens-Datenbanken - Der Weg zu bereichsübergreifenden Datenstrukturen, September 1989
- Heft 64: C. Berkau, W. Kraemer, A.-W. Scheer: Strategische CIM-Konzeption durch Eigenentwicklung von CIM-Modulen und Einsatz von Standardsoftware, Dezember 1989
- Heft 65: A. Hars, A.-W. Scheer: Entwicklungsstand von Leitständen^[1], Dezember 1989

- Heft 66: W. Jost, G. Keller, A.-W. Scheer: CIMAN - Konzeption eines DV-Tools zur Gestaltung einer CIM-orientierten Unternehmensarchitektur, März 1990
- Heft 67: A.-W. Scheer: Modellierung betriebswirtschaftlicher Informationssysteme (Teil 1: Logisches Informationsmodell), März 1990
- Heft 68: W. Kraemer: Einsatzmöglichkeiten von Expertensystemen in betriebswirtschaftlichen Anwendungsgebieten, März 1990
- Heft 69: A.-W. Scheer, R. Bartels, G. Keller: Konzeption zur personalorientierten CIM-Einführung, April 1990
- Heft 70: St. Spang, K. Ibach: Zum Entwicklungsstand von Marketing-Informationssystemen in der Bundesrepublik Deutschland, September 1990
- Heft 71: D. Aue, M. Baresch, G. Keller: URMEL, Ein UnternehmensModELLierungsansatz, Oktober 1990
- Heft 72: M. Zell: Datenmanagement simulationsgestützter Entscheidungsprozesse am Beispiel der Fertigungssteuerung, November 1990
- Heft 73: A.-W. Scheer, M. Bock, R. Bock: Expertensystem zur konstruktionsbegleitenden Kalkulation, November 1990
- Heft 74: R. Bartels, A.-W. Scheer: Ein Gruppenkonzept zur CIM-Einführung, Januar 1991
- Heft 75: M. Nüttgens, St. Eichacker, A.-W. Scheer: CIM-Qualifizierungskonzept für Klein- und Mittelunternehmen (KMU), Januar 1991
- Heft 76: Ch. Houy, J. Klein: Die Vernetzungsstrategie des Instituts für Wirtschaftsinformatik - Migration vom PC-Netzwerk zum Wide Area Network (noch nicht veröffentlicht)
- Heft 77: W. Kraemer: Ausgewählte Aspekte zum Stand der EDV-Unterstützung für das Kostenmanagement: Modellierung benutzerindividueller Auswertungssichten in einem wissensbasierten Controlling-Leitstand, Mai 1991
- Heft 78: H. Heß: Vergleich von Methoden zum objektorientierten Design von Softwaresystemen, August 1991
- Heft 79: A.-W. Scheer: Konsequenzen für die Betriebswirtschaftslehre aus der Entwicklung der Informations- und Kommunikationstechnologien, Mai 1991
- Heft 80: G. Keller, J. Kirsch, M. Nüttgens, A.-W. Scheer: Informationsmodellierung in der Fertigungssteuerung, August 1991
- Heft 81: A.-W. Scheer: Papierlose Beratung - Werkzeugunterstützung bei der DV-Beratung, August 1991
- Heft 82: C. Berkau: VOKAL (System zur Vorgangskettendarstellung und -analyse) - Struktur der Modellierungsmethode - Juni 1991 (wird nicht verlegt)
- Heft 83: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - Theoretical Foundations - 1991
- Heft 84: A. Hars, R. Heib, Ch. Kruse, J. Michely, A.-W. Scheer: Concepts of Current Data Modelling Methodologies - A Survey - 1991

- Heft 85: W. Hoffmann, M. Nüttgens, A.-W. Scheer, St. Scholz: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 1: Produktionsplanung), Oktober 1991
- Heft 86: A.-W. Scheer: Koordinierte Planungsinselfn: Ein neuer Lösungsansatz für die Produktionsplanung, November 1991
- Heft 87: M. Nüttgens, G. Keller, A.-W. Scheer, S. Stehle: Konzeption hyperbasierter Informationssysteme, Dezember 1991
- Heft 88: W. Hoffmann, B. Maldener, M. Nüttgens, A.-W. Scheer: Das Integrationskonzept am CIM-TTZ Saarbrücken (Teil 2: Produktionssteuerung), Januar 1992
- Heft 89: G. Keller, M. Nüttgens, A.-W. Scheer: Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßkosten (EPK)", Januar 1992
- Heft 90: C. Berkau, A.-W. Scheer: VOKAL (System zur Vorgangskettendarstellung und -Analyse), Teil 2: VKD-Modellierung mit VOKAL
- Heft 91: C. Berkau: Konzept eines controllingbasierten Prozeßmanagers als intelligentes Multi-Agent-System, Januar 1992
- Heft 92: A. Hars, R. Heib, Chr. Kruse, J. Michely, A.-W. Scheer: Approach to Classification for Information Engineering - Methodology and Tool Specification, August 1992
- Heft 93: M. Nüttgens, A.-W. Scheer, M. Schwab: Integrierte Entsorgungssicherung als Bestandteil des betrieblichen Informationsmanagements, August 1992
- Heft 94: Chr. Kruse, A.-W. Scheer: Modellierung und Analyse dynamischen Systemverhaltens, Oktober 1992
- Heft 95: R. Backes, W. Hoffmann, A.-W. Scheer: Konzeption eines Ereignisklassifikationssystems in Prozeßketten, November 1992
- Heft 96: P. Loos: Die SEMantik eines erweiterten Entity-Relationship-Modells und die Überführung in SQL-Datenbanken, November 1992