

# Empfehlungen der SVVK- Automationskommission zur Darstellung logischer Datenstrukturen

Autor(en): **Sievers, B.**

Objektyp: **Article**

Zeitschrift: **Vermessung, Photogrammetrie, Kulturtechnik : VPK =  
Mensuration, photogramm trie, g nie rural**

Band (Jahr): **81 (1983)**

Heft 9

PDF erstellt am: **19.09.2024**

Persistenter Link: <https://doi.org/10.5169/seals-231652>

## **Nutzungsbedingungen**

Die ETH-Bibliothek ist Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Inhalten der Zeitschriften. Die Rechte liegen in der Regel bei den Herausgebern. Die auf der Plattform e-periodica ver ffentlichten Dokumente stehen f r nicht-kommerzielle Zwecke in Lehre und Forschung sowie f r die private Nutzung frei zur Verf gung. Einzelne Dateien oder Ausdrucke aus diesem Angebot k nnen zusammen mit diesen Nutzungsbedingungen und den korrekten Herkunftsbezeichnungen weitergegeben werden. Das Ver ffentlichen von Bildern in Print- und Online-Publikationen ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Die systematische Speicherung von Teilen des elektronischen Angebots auf anderen Servern bedarf ebenfalls des schriftlichen Einverst ndnisses der Rechteinhaber.

## **Haftungsausschluss**

Alle Angaben erfolgen ohne Gew hr f r Vollst ndigkeit oder Richtigkeit. Es wird keine Haftung  bernommen f r Sch den durch die Verwendung von Informationen aus diesem Online-Angebot oder durch das Fehlen von Informationen. Dies gilt auch f r Inhalte Dritter, die  ber dieses Angebot zug nglich sind.

entsprechende Ausbildung erworben werden und setzt somit ein entsprechendes Lehrprogramm und Lehrmethodik voraus. Diese Forderung beschränkt sich nicht auf die Lehrtätigkeit an technischen Hochschulen, sondern wird – sicherlich nach der praktischen Seite ausgerichtet – auch an den sog. Technischen Lehranstalten bzw. an Spezialschulen für photogrammetrische Techniker nötig. An die Auszubildenden wird unvermeidlich ein grösseres Wissen, vor allem in den mathematischen und physikalischen Disziplinen, verlangt werden müssen.

Die dieser Situation angepasste Forschung und Entwicklung wird einmal die Erfolgsaussichten der entsprechenden Industrie entscheidend beeinflus-

sen. Es wird dabei nicht darum gehen – um die Terminologie der Marktwirtschaft zu gebrauchen –, bereits abgefahrenen Zügen nachzutruern, sondern man muss *neue* Züge auf den Perron bringen. Für das ausführende Ingenieurbureau wird diese Entwicklung und die damit zusammenhängende Reformation der Arbeitsweisen der wohl entscheidendste Faktor in bezug auf die Wirtschaftlichkeit eines solchen Unternehmens bedeuten.

#### Literaturhinweise

Manfred Schröder: Stand der Entwicklungen von Erderkundungssatelliten und Shuttle-System. Deutsche Gesellschaft für Photogrammetrie und Fernerkundung e.V., November 1981

Hellmut Schmid: Über den Wandel der geometrischen-algebraischen Modellvorstellung in der Photogrammetrie unter dem Einfluss computergestützter Auswerteverfahren. Vermessung, Photogrammetrie, Kulturtechnik 7–8/79, Seiten 197–205

– Ein allgemeiner Ausgleichs-Algorithmus für die numerische Auswertung in der Photogrammetrie. Institut für Geodäsie und Photogrammetrie, ETH Zürich, Mitteilungen Nr. 22, Juni 1977

– Vom freien zum gelagerten Netz. Institut für Geodäsie und Photogrammetrie, ETH Zürich, Mitteilungen Nr. 29, Dezember 1980

Adresse des Verfassers:

Prof. Dr.-Ing. Dr.-Ing. E. h. Hellmut H. Schmid  
Institut für Geodäsie und Photogrammetrie  
ETH-Hönggerberg, CH-8093 Zürich

# Empfehlungen der SVK-Automationskommission zur Darstellung logischer Datenstrukturen

B. Sievers\*

Zu Beginn der Empfehlungen wird auf die Problematik hingewiesen, die bei der Analyse logischer Datenstrukturen zu bewältigen ist. Anschliessend werden wichtige Begriffe definiert und ihre graphische Darstellung erläutert. Zugleich werden Regeln aufgeführt, die einen logischen Aufbau der Datenstruktur unterstützen. Ein einfaches Beispiel veranschaulicht die Darstellung als Entitätenblockdiagramm. Abschliessend befassen sich die Empfehlungen mit den Vorteilen der vorgestellten Darstellungsform.

## 1. Einleitung, Gegenstand

Jede Anwendung der elektronischen Datenverarbeitung (EDV) beruht auf einem dem Anwendungszweck angepassten Modell der Realität. In diesem Modell wird festgelegt, in welcher Form, durch welche Art von Daten die Objekte dieses Modells beschrieben werden sollen.

Ein immer wichtiger werdendes Postulat verlangt, dass EDV-Datensysteme unabhängig von der Hardware entwickelt und dokumentiert werden sollten. Man spricht von der logischen Datenstruktur im Gegensatz zur physischen Datenorganisation im Computer.

\*B. Sievers hat diese Empfehlungen aufgrund der Aussprachen in der SVK-Automationskommission ausgearbeitet. Wertvolle Arbeitsgrundlagen waren [Zehnder 83] und [Frank 82].

Institut für Geodäsie und Photogrammetrie  
ETH-Hönggerberg, CH-8093 Zürich  
Separata Nr. 57

Um Einblick in ein EDV-Datensystem zu erhalten, ist es wichtig, seine logische Datenstruktur zu kennen. Deshalb ist eine einheitliche Darstellung dieser Datenstrukturen bedeutsam. Eine solche Darstellung kommt aber meist auch der Logik des Systems zugut. Sie zwingt dazu, sich Beziehungen zwischen den Daten bewusst zu machen. Solange die EDV als sequentielle Abläufe in einzelnen Programmen eingesetzt wurde und diese Programme einfache Strukturen zu bearbeiten hatten, war die Darstellung dieser Datenstrukturen nicht vordringend. Moderne und vor allem zukünftige Systeme werden aber immer grössere und komplexer strukturierte Datenmengen bewältigen müssen. Damit wird die klare Darstellung dieser Strukturen von erster-rangiger Bedeutung.

## 2. Zweck dieser Empfehlung

Eine einheitliche Darstellung der logischen Datenstrukturen soll das einem

Datensystem zugrundeliegende Modell aufzeigen und verständlich machen. Ferner sollen die dabei auftretenden Begriffe klargestellt werden. Negativ ausgedrückt: der Zweck dieser Empfehlungen besteht nicht darin, die als *Beispiele* dargestellten Modelle zu empfehlen oder abzulehnen. Das ist eine Aufgabe, mit der sich voraussichtlich andere Gremien befassen werden.

## 3. Abgrenzungen

Logische Datenstrukturen werden – wie erwähnt – dargestellt, um die Logik eines Datensystems zu beschreiben. In erster Linie geht es also darum, Information über das System zu vermitteln; *die Empfehlungen beziehen sich primär auf die Form der Darstellung.*

Wenn im folgenden dennoch Regeln aufgestellt werden, so sind das Empfehlungen, die der Verbesserung des logischen Aufbaus dienen. Insbesondere ist darauf hinzuweisen, dass logische Datenstrukturen verbindlichen Formvorschriften unterliegen, sobald sie Grundlage für die Implementierung moderner Datenbanken sind.

Hält man sich nicht daran, so verzichtet man auf Konsistenzbedingungen, was dem Datenbankkonzept zuwiderläuft, welches auch in der Nachführung widerspruchsfreie Daten erzwingen muss.

#### 4. Begriffe und ihre Darstellung

##### 4.1 Entität, Entitätsmenge

«Eine Entität ist ein individuelles Exemplar von Elementen der realen oder der Vorstellungswelt. Sofern eine Beziehung zwischen Entitäten eine Bedeutung in der realen Welt hat, kann auch ein individuelles Exemplar einer solchen Beziehung als Entität aufgefasst werden» [Zehnder 83, S. 30].

Entität ist der gebräuchliche Begriff zur Beschreibung eines beliebigen, in einem Modell als abgegrenzt erscheinenden Teiles der Welt. Was eine Entität ist, hängt vom verwendeten Modell, vom Ziel der Anwendung ab.

Eine Entität ist also, vereinfacht ausgedrückt, ein abgegrenztes Objekt eines Modells der realen Welt. Sie wird im allgemeinen durch bestimmte Attribute (vgl. 4.3) beschrieben, deren Werte angegeben werden: beispielsweise das Haus, charakterisiert durch die Adresse «Bahnhofstrasse 21».

Entitäten mit gleichen oder ähnlichen Attributen, aber unterschiedlichen Attributwerten, fassen wir zu *Entitätsmengen* zusammen.

Wir bilden beispielsweise eine Entitätsmenge **Haus**, indem wir von den individuellen Attributen der verschiedenen Häuser abstrahieren.

Entitätsmengen werden durch Kästchen dargestellt, die den Namen der Entitätsmenge (in Grossbuchstaben oder fett gedruckt) enthalten (Abb. 1).

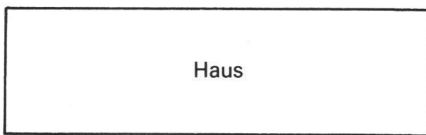


Abb. 1 Darstellung einer Entitätsmenge

Sofern notwendig, werden die charakteristischen Attribute mit Kleinbuchstaben eingetragen, also beispielsweise die administrativen Daten eines Hauses (Abb. 2).

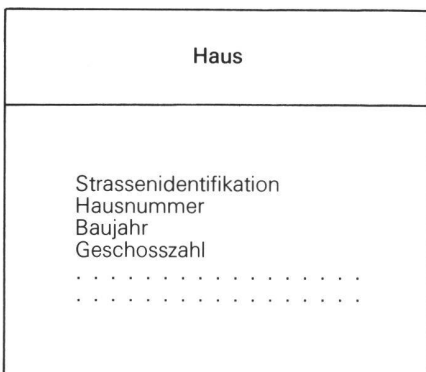


Abb. 2 Darstellung einer Entitätsmenge mit ihren Attributen

##### 4.2 Beziehungen

Die Beschreibung einer Datenstruktur besteht nicht nur aus der Aufzählung der Teile, also den Entitätsmengen, sondern auch aus der Beschreibung der Beziehungen zwischen diesen.

Bestehen Beziehungen zwischen zwei Entitätsmengen (z. B. Gemeinde-Kanton), so werden sie als Linien zwischen den entsprechenden Entitätsmengen-Kästchen eingezeichnet (Abb. 3).

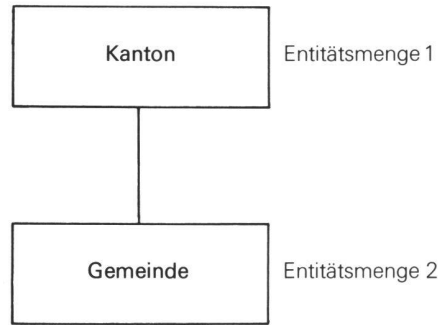


Abb. 3 Darstellung einer Beziehung zwischen zwei Entitätsmengen

Um Beziehungen zu klassieren, prüft man, wieviele Entitäten der einen Art jeweils einer Entität der andern Art zugeordnet sein können. Daraus ergibt sich der Assoziationsstyp.

Man unterscheidet vier Assoziationsstypen für die Beziehung der Entitätsmenge 1 zur Entitätsmenge 2:

- 1: einfache Assoziation einer Entität aus der Entitätsmenge 1 ist genau eine andere Entität in der Entitätsmenge 2 zugeordnet
- c: konditionelle Assoziation einer Entität aus der Entitätsmenge 1 ist eine oder keine andere Entität in der Entitätsmenge 2 zugeordnet
- m: multiple Assoziation einer Entität aus der Entitätsmenge 1 ist eine Mehrzahl (mindestens aber eine) von Entitäten in der Entitätsmenge 2 zugeordnet
- mc: multipel-konditionelle Assoziation einer Entität aus der Entitätsmenge 1 ist eine Mehrzahl, eine oder auch keine Entität in der Entitätsmenge 2 zugeordnet

Solche Assoziationen sind nun aber sowohl von Entitätsmenge 1 zu Entitätsmenge 2 als auch umgekehrt möglich. In jeder Beziehung zwischen zwei Entitätstypen bestimmt man deshalb ein Paar von Assoziationsstypen, indem man einmal von der einen und einmal von der andern Entität ausgeht.

Beispiel: Zu jedem Kanton gehören eine oder mehrere Gemeinden: Assoziationsstyp m. Andererseits gehört zu jeder

Gemeinde genau ein Kanton: Assoziationsstyp 1. Die Beziehung Kanton-Gemeinde ist damit insgesamt 1 : m. Dies wird im Entitätenblockdiagramm eingetragen (Abb. 4).

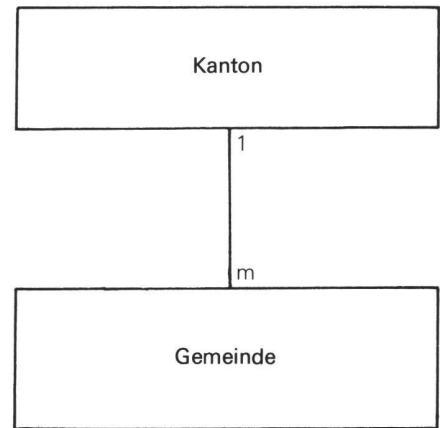


Abb. 4 Darstellung der Assoziationsstypen einer Beziehung zwischen zwei Entitätsmengen

In Ausnahmefällen wird statt der unbestimmten Anzahl m ein bestimmter Wert bzw. ein Bereich von Werten eingesetzt.

Zwischen zwei Entitäten können auch mehrere Beziehungen vorhanden sein, so z. B. bei Objekten, deren flächenhafte Ausdehnung mit einer Entitätsmenge beschrieben wird: jede Entität **Fläche** wird durch eine Entität **Linie** von einer angrenzenden Entität **Fläche** getrennt. Diese Nachbarschaftsbeziehung kann folgendermassen dargestellt werden (Abb. 5).

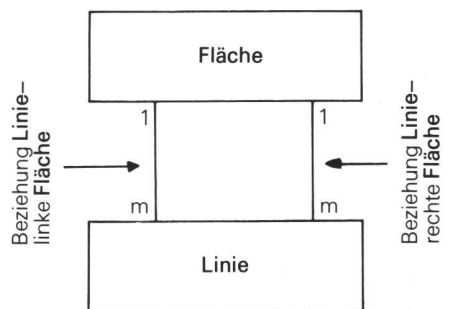


Abb. 5 Darstellung der Beziehungen zwischen der Entitätsmenge **Fläche** und der Entitätsmenge **Linie**

Das mit Abb. 5 beschriebene Modell verlangt, dass die ausserhalb des Modells liegende Restfläche gespeichert wird. Soll dies nicht geschehen, muss die Beziehung **Fläche-Linie** vom Typ 1:mc sein: am Rand des Modells begrenzen die Linien dann nur eine Fläche.

### 4.3 Attribute

Ein Attribut beschreibt eine bestimmte Eigenschaft der Entität einer Entitätsmenge näher.

Beispiel: Zu einer Entität **Haus** geben die Attribute «Anzahl Stockwerke», «Anzahl Wohnungen», «Baujahr» usw. die in das Modell einbezogenen Eigenschaften an (Abb. 6).

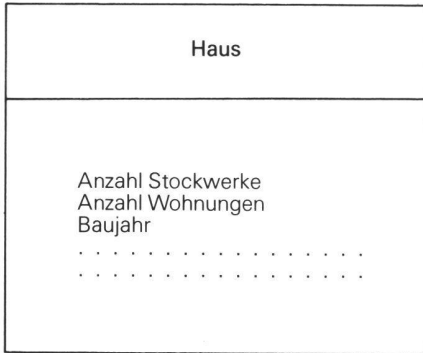


Abb. 6 Darstellung einer Entitätsmenge mit ihren Attributen

Befasst sich ein Vermessungsfachmann mit dem Entwurf der Datenstruktur für ein Programmsystem, das grosse Datenmengen verwalten und aktuell erhalten soll, helfen beim Festlegen der Attribute folgende Regeln:

4.3.1 Ein Attribut darf nur *einen* Wert annehmen. Ist dies nicht realistisch, kann das Attribut in mehrere Attribute aufgeteilt werden.

Im Beispiel von Abb. 6 kann das Attribut «Baujahr» eines Hauses, das im Jahre 1920 gebaut und 1960 umgebaut worden ist, nicht gleichzeitig die Werte «1920» und «1960» annehmen. Dies wird erst möglich, wenn «Baujahr» aufgeteilt wird in «Erstellungsbaujahr» und «Ergänzungsbaujahr».

4.3.2 Die Attributswerte werden durch Konstanten aus einem bestimmten Wertebereich dargestellt. Ist ein Attributswert einmal festgelegt, kann er später nur verändert werden, indem anstelle der alten eine neue Konstante aus dem gleichen Wertebereich tritt. Im Beispiel von Abb. 6 kann das Attribut «Anzahl Stockwerke» Werte z.B. aus dem Wertebereich «ganze positive Zahlen zwischen 1 und 100» annehmen.

4.3.3 Ein Wertebereich für Attribute soll statisch sein: bei der Definition der Daten(beschreibung) wird eine Menge von zulässigen Werten festgelegt, die sich im Verlaufe der Zeit nicht ändert. Beispiel: Der Wertebereich der X-Koordinate eines Punktes in der Schweiz sei 62000.0000 ... 302000.0000. Damit ist gleichzeitig auch die Einheit festgelegt (hier z.B. Meter mit vier Nachkommastellen).

4.3.4 Ein Attribut darf nicht Attributswerte aufweisen, die ihrerseits Mengen sind: Wenn eine Eigenschaft in unbestimmter Zahl mit einer Entität verbunden ist, muss sie als selbständige Entität behandelt werden.

Beispiel (Abb. 7):

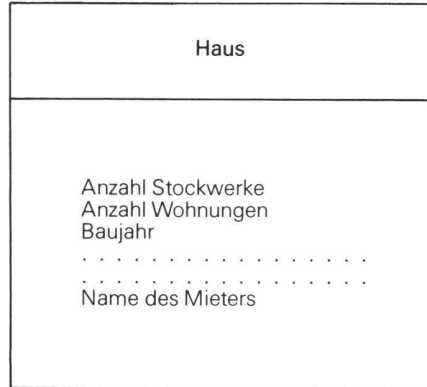


Abb. 7

Abb. 7 beschreibt ein Modell, in dem für jedes Haus nur ein einziger Mieter zulässig ist. Im Modell zu Abb. 8 ist diese Einschränkung nicht mehr enthalten (Abb. 8).

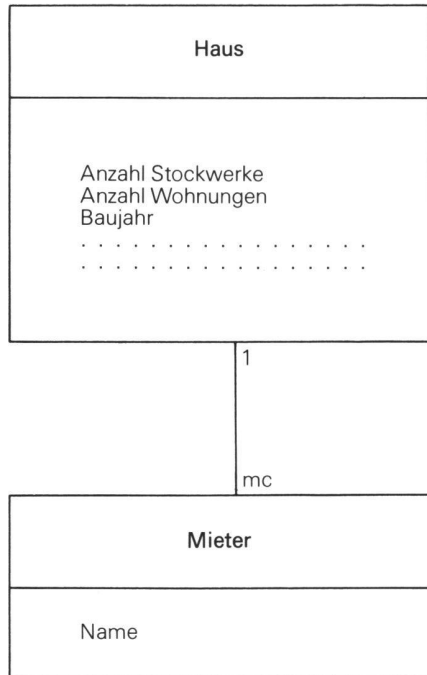


Abb. 8

4.3.5 Das Attribut muss voll von der Entität abhängig sein: Wird eine Entität fixiert, so ist damit auch der Wert des Attributes bestimmt.

Attribute oder Attributskombinationen in der gleichen oder in verschiedenen Entitätsmengen dürfen nicht funktional voneinander abhängig sein:

*Volle Abhängigkeit* liegt vor, wenn ein Attribut oder eine Attributskombination A von einem andern Attribut oder einer Attributskombination B der gleichen Entitätsmenge funktional abhängig.

Beispiel: Eine Entitätsmenge **Person** sei durch die Attribute «Name», «Vorname» und «Geburtsdatum» eindeutig identifiziert. Wird nun noch eine Attributgruppe «AHV-Nr.» eingeführt, bei der jede der Zahlengruppen einzeln aufgeführt wird (Namencode, Geburtsjahr, Geschlechts-Geburtsstags-Code usw.), besteht volle Abhängigkeit zwischen den Attributen.

*Externe Abhängigkeit* liegt vor, wenn A und B verschiedene Entitätsmengen beschreiben, die aber miteinander in Beziehung stehen.

Beispiel: in der Adresse einer **Person** sollten nicht gleichzeitig «Ortsname» und «Postleitzahl» eingeschlossen werden, weil die «Postleitzahl» offensichtlich vom «Ortsnamen» abhängig ist und z.B. in einer Entitätsmenge **Postkreis** (Quartier, Ortschaft, Fraktion...) festgelegt ist.

(Abschnitt 6.1 wird sich mit sinnvoller Redundanz befassen.)

4.3.6 Ein bestimmtes Attribut darf nur in *einer* Entitätsmenge vorkommen. Taucht es in zwei oder mehr Beschreibungen von Entitätsmengen auf, wie z.B. Strassenname in **Haus** und **Strasse** (Abb. 9),

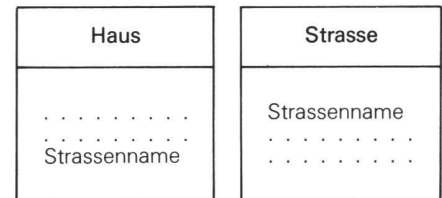
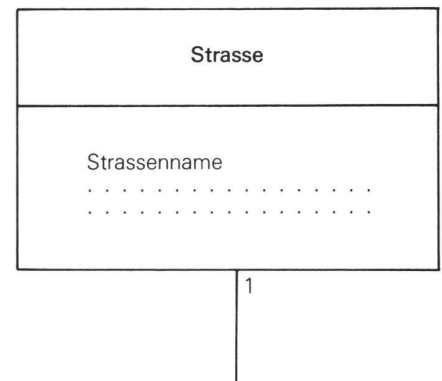


Abb. 9

so weist dies auf Redundanz und Abhängigkeiten zwischen diesen Entitätsmengen hin, die unzweckmässig sind. Wir vermeiden dies, indem wir das obige Beispiel umformen (Abb. 10).



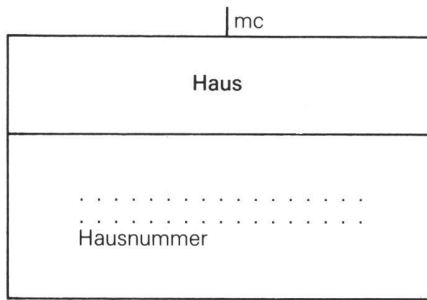


Abb.10

Durch diese Beziehung kann der zu einem **Haus** gehörige Strassenname jederzeit festgestellt werden. Er darf aber nicht als Attribut im **Haus** erscheinen. (Das Abb.10 zugrundegelegte Modell legt fest, dass jedes Haus nur an einer Strasse liegt.)

#### 4.4 Darstellung des Zugriffs zu den Daten

Im Entitätenblockdiagramm soll auch ersichtlich sein, wie zu den strukturierten Datensätzen zugegriffen werden kann.

Soll eine bestimmte Entität gesucht werden, bestehen zwei Möglichkeiten:

- man geht von der übergeordneten Entitätsmenge aus und sucht dann die untergeordnete Entitätsmenge nach der gewünschten Entität ab (Beispiel: gesucht sei die Telefonnummer von «Hans Radlinski» in Zürich-Stadt: übergeordnete Entität ist das Abonnentenverzeichnis der Schweiz, untergeordnete Entitätsmenge sind die Telefonabonnenten, die im Telefonbuch Zürich-Stadt aufgeführt sind.)

- man benützt eindeutig kennzeichnende Attribute oder Attributskombinationen zum Aufbau der Datenspeicherorganisation (z. B. Hash-Organisation, bei der die Speicheradresse einer Entität eine Funktion ihres kennzeichnenden Attributes ist). Diese Organisation unterstützt ein rasches Wiederauffinden sehr wirkungsvoll. Im Entitätenblockdiagramm lässt sich diese Zugriffsunterstützung darstellen: bei eindeutig kennzeichnendem Attribut (Abb. 11);

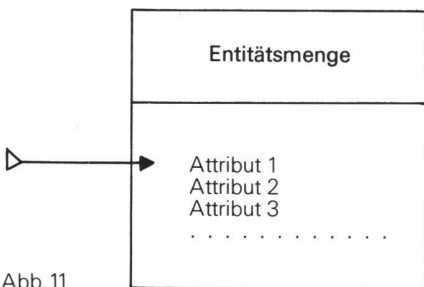


Abb.11

bei eindeutig kennzeichnender Attributskombination (Abb. 12).

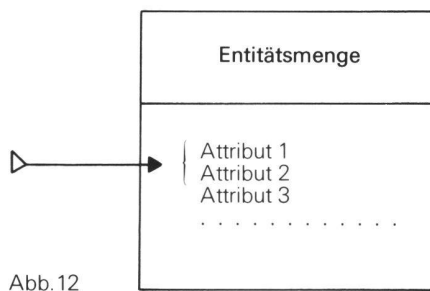


Abb.12

Sind hingegen die kennzeichnenden Attribute oder Attributskombinationen nicht innerhalb des ganzen Modelles, sondern nur innerhalb eines bestimmten Bereiches eindeutig, kann dies auch dargestellt werden (Abb. 13).

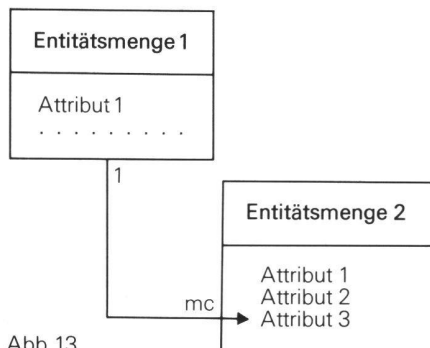


Abb.13

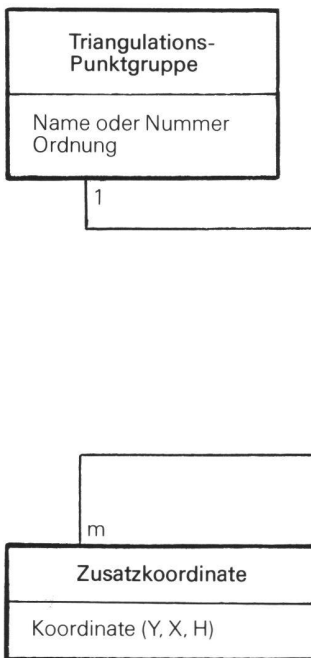


Abb.15 Entitätenblockdiagramm mit Generalisierung

Beispiel: Strassenamen sind meist nur innerhalb einer Gemeinde eindeutig (Abb. 14).

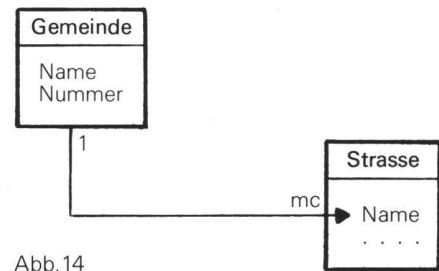
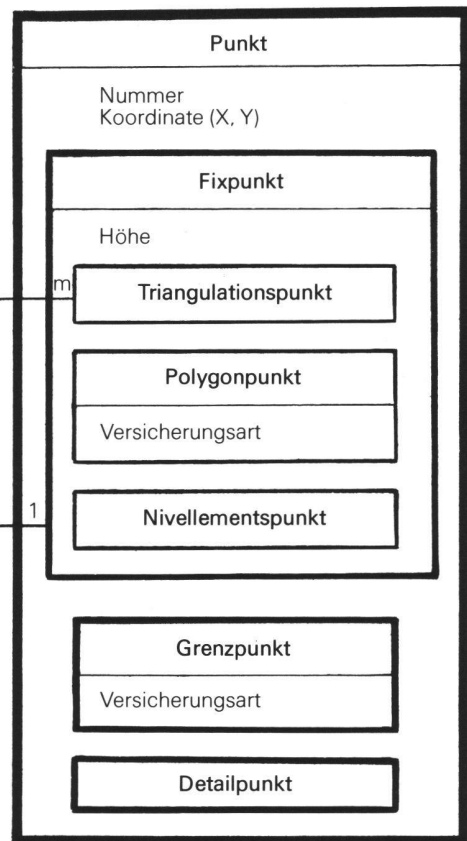


Abb.14

#### 4.5 Generalisierung

Die Darstellungsform komplizierter Datenstrukturen kann nach [Frank 82] durch die Generalisierung übersichtlicher gestaltet werden: die Generalisierung wird graphisch durch ineinander geschachtelte Entitätskästchen dargestellt, wobei alle geschachtelten Entitätsmengen die Attribute erben, die in der Oberbegriffs-Entitätsmenge festgelegt worden sind (Abb. 15).

**Polygonpunkt** hat also die Attribute «Nummer», «Koordinate (Y, X)», «Höhe» und «Versicherungsart» und eventuell eine untergeordnete Entitätsmenge **Zusatzkoordinate**.



#### 4.6 Entitätenblockdiagramme

Die Darstellung einer logischen Datenstruktur mit Entitätsmengen-Kästchen, Verbindungslinien für die Beziehungen und Bezeichnung der Assoziationstypen wird *Entitätenblockdiagramm* genannt. Im *erweiterten Entitätenblockdiagramm* werden zusätzlich noch der Datenzugriff und die Generalisierung dargestellt.

#### 4.7 Redundanz

«Redundanz ist in einem Datenbestand genau dann vorhanden, wenn ein Teil davon ohne Informationsverlust weggelassen werden kann» [Zehnder 83, S. 38]. Redundant sind, somit Daten, die mehrfach gespeichert sind oder gespeicherte Daten, die auch aus anderen, bereits gespeicherten Daten abgeleitet werden könnten. Redundanzen treten aber auch auf, wenn die Regeln 4.3.5 und 4.3.6 nicht beachtet werden (Abschnitt 6.1 befasst sich mit sinnvoller Redundanz).

#### 4.8 Konsistenz

«Wir verstehen darunter die Übereinstimmung des Inhalts des Datenbestandes mit der Datenbeschreibung und die innere Widerspruchsfreiheit der Daten» [Zehnder 83, S.132] (vgl. dazu 3. Absatz in Abschnitt 3).

### 5. Anwendung auf ein einfaches Beispiel

Nehmen wir an, unser Modell der realen Welt könne mit einem Plan beschrieben werden. Es enthalte im Plangebiet Strassenparzellen, Grundstücke und Häuser mit ihren Umrisslinien. Als Punkte seien im Modell nur Grenz- und Detailpunkte enthalten (Abb. 16).

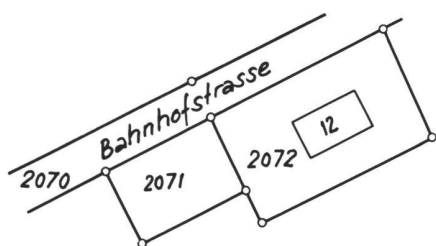


Abb. 16 Beispiel für ein Modell der realen Welt

Dazu ein erster Entwurf eines Entitätenblockdiagramms (Abb. 17).

Versuchen wir nun, die Möglichkeiten des Beispiels in Abb. 17 auszuloten: Wenn ein Haus oder eine Strasse zugleich Parzelle ist, müssen die geometrischen Informationen (Punktkoordinaten, Liniendefinitionen) für jede der drei Bedeutungen gesondert verfügbar sein.

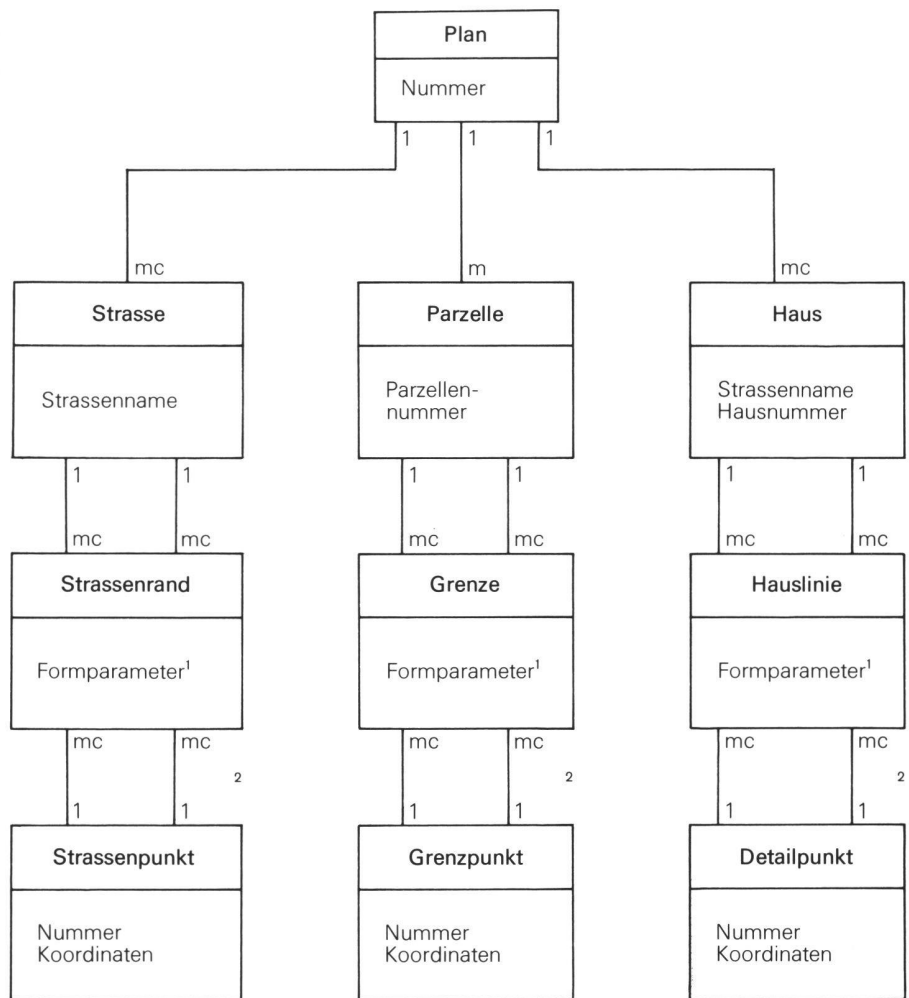


Abb. 17 Beispiel für ein Entitätenblockdiagramm zu Abb. 16

<sup>1</sup> mit «Formparameter» wird die Krümmung der Strecke beschrieben.  
<sup>2</sup> (Abb. 18)

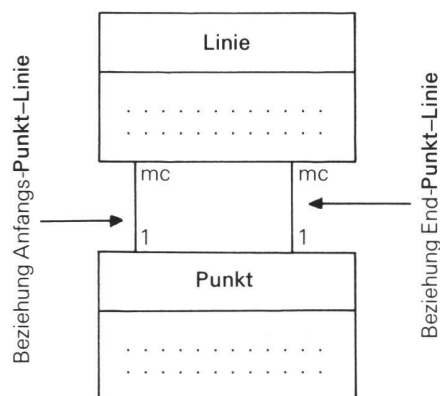


Abb. 18

Gewisse Attribute («Strassenname», «Formparameter», «Punktnummer» und «Punktkoordinaten») kommen in mehreren Entitätsmengen vor. Bei Mutationen muss in diesem Falle immer der ganze Datenbestand auf seine Konsistenz überprüft werden.

Nachbarschaftsbeziehungen (z. B. Haus 12 liegt in Parzelle 2072, Parzelle 2071 grenzt an eine Strasse an) sind nicht eingeführt.

Häuser müssen eine eindeutige Adresse haben, jedes Haus muss an einer Strasse liegen. (Diese Bedingung ist im vorliegenden Modell bewusst angenommen worden. Andere Möglichkeiten bieten sich mit der Einführung der Assekuranzznummern oder einer allfälligen doppelten Adresse.)

In einem verbesserten Entwurf soll nun versucht werden, Identitäten zu berücksichtigen und Mehrfachspeicherungen möglichst zu verhindern (Abb. 19).

Mit der Einführung einer Entitätsmenge **Nutzung** haben wir ermöglicht, dass sich eine Parzelle aus verschiedenen Nutzungsflächen (z. B. Wald, Wiese, Gebäude, Weg...) zusammensetzen kann. Dieses Modell unterstellt, dass die Nutzungsflächen den Parzellen untergeordnet sind: Eine Fläche gleicher Nutzung kann maximal so gross sein wie eine Parzellenfläche, andererseits kann eine Parzellenfläche aus mehreren Flächen verschiedener Nutzung zusammengesetzt sein.

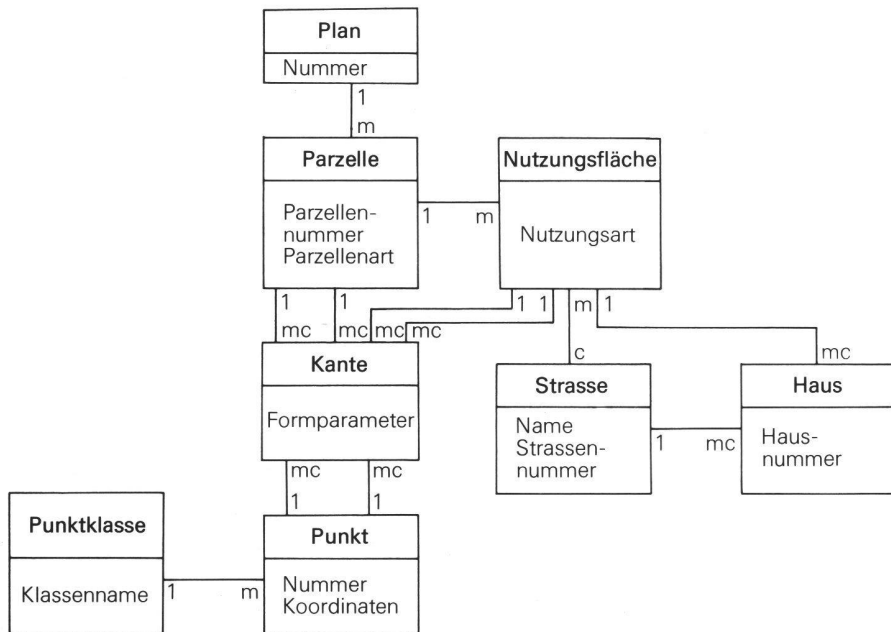


Abb. 19 Verbessertes Beispiel für ein Entitätenblockdiagramm zu Abb. 16

Die **Kante** ist wie der **Punkt** zur Obermenge der im ersten Entwurf dargestellten verschiedenen Punkt- bzw. Linien-Entitätsmengen geworden; dies ermöglicht, dass zusammenfallende Grenz- und Nutzungslinien (mit gleichen Anfangs- und Endpunkten) nur einmal vorkommen.

Den verschiedenen möglichen Bedeutungen der Entitätsmenge **Punkt** (Grenz-, Detailpunkt...) dient die eingeführte Entitätsmenge **Punktklasse**. Fallen hingegen in einem physischen Punkt Punkte verschiedener Punktklassen zusammen, so muss gemäss Diagramm von Abb. 19 je Punktklasse ein Punkt gespeichert werden.

**Beziehung Strasse–Nutzung:** Eine Strasse setzt sich aus mehreren Nutzungsflächen zusammen; eine Nutzungsfläche darf aber höchstens zu einer Strasse gehören, was bedeutet, dass z. B. Kreuzungen aufzuteilen sind.

**Beziehung Strasse–Haus:** Die Bemerkung zu Abb. 10 gilt auch hier. Für Modelle, in denen nicht jedes Haus an einer Strasse oder ein Haus an mehreren Strassen liegt, könnte folgendes Entitätenblockdiagramm verwendet werden (Abb. 20).

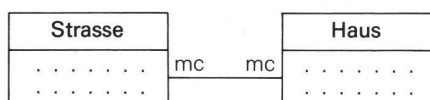


Abb. 20

\*Diese in der Praxis nicht realistische Einschränkung liesse sich einfach beseitigen: Zwischen die Entitätsmengen **Plan** und **Parzelle** wird eine «Zwischen»-Entitätsmenge eingefügt.

Bei der Implementierung von m/mc-m/mc- oder c/mc-c/mc-Beziehungen zwischen Entitätsmengen ergäben sich Schwierigkeiten. Der EDV-Spezialist löste diese Beziehungen daher durch Einfügen einer «Zwischen»-Entitätsmenge auf (Abb. 21).

In einem letzten Beispiel sollen bisher erkannte Modelleinschränkungen eliminiert werden.

Modellumschreibung:

In einer Gemeinde, deren Grundriss in Pläne unterteilt sei, sollen Parzellen und Nutzungsflächen gleichberechtigt enthalten sein. Besonders hervorzuheben sind Punkte, Strassen und Häuser. Zwischen Strassen und Häusern kann eine Beziehung bestehen, zwischen Parzellen und Häusern muss sie eingeführt werden (über die Entitätsmenge **Nutzung**). Die Nutzung muss parzellenweise ausgewiesen, soll aber auch parzellenunabhängig eingeführt werden können.

Als bleibende Einschränkungen dieses Modelles seien unter anderen erwähnt:

- Eine Parzelle darf nicht auf mehreren Plänen liegen\*.
- Unterirdische Bauten können nicht ausgeschieden werden. Gleiches gilt für Gebäudeunterteilungen.
- Punktnummern müssen eindeutig sein.



Abb. 21

- Plan- und Parzellennummern sowie Strassennamen müssen innerhalb einer Gemeinde eindeutig sein.

In Abb. 22 ist die Datenstruktur dieses Modells dargestellt.

Erläuterungen: Die Entitätsmenge **Bedeutung** trägt die Information über die «Nutzungsart» der **Kante** und ihrer Endpunkte (z. B. «Strasse», «Parzelle», «Haus»). Die Entitätsmenge **Knoten** ermöglicht es, in einem physischen Punkt zusammenfallende Punkte verschiedener Punktklassen (z. B. Polygonpunkt = Grenzpunkt = Detailpunkt) sofort aufzufinden.

Die Darstellung **Punkt** steht stellvertretend für die vollständige Darstellung in Abb. 15, wobei das Attribut «Koordinate (Y, X)» aus der Entitätsmenge **Punkt** in die Entitätsmenge **Knoten** versetzt ist.

## 6. Vorteile des Entitätenblockdiagramms

### 6.1 Sinnvolle Redundanz

Eine formal eindeutige Darstellung der Datenstruktur gestattet es, auf der konzeptionellen Ebene noch vorhandene, unnötige Redundanzen zu erkennen und zu eliminieren.

Man unterscheidet bei der Beurteilung sinnvoller Redundanz drei Ebenen:

- Konzeptionelle Ebene
- Benutzerebene (Externe Ebene)
- Physische Ebene der Datenbasis (Interne Ebene)

#### 6.1.1 Konzeptionelle Ebene

Die Analyse und Darstellung der logischen Datenstruktur gehört auf diese Ebene. Ziel ist es, ein konsistentes Datensystem zu konstruieren. Die Gefahr von «Mutationsanomalien» (redundant gespeicherte Daten werden nicht mitmutiert oder können wegen Redundanz nicht mutiert werden) muss gegen die Wünsche nach Datensicherheit (redundanzfreie Daten lassen sich nach einer Beschädigung kaum mehr wiederherstellen) oder schnellerem Zugriff abgewogen werden. Resultat dieser Beurteilung ist eine sinnvolle Redundanz.

#### 6.1.2 Benutzerebene

Der Benutzer wünscht vom Datensystem, dass es seine Anfragen mit umfassenden, kombinierten Darstellungen mit genügend Redundanz beantwortet.

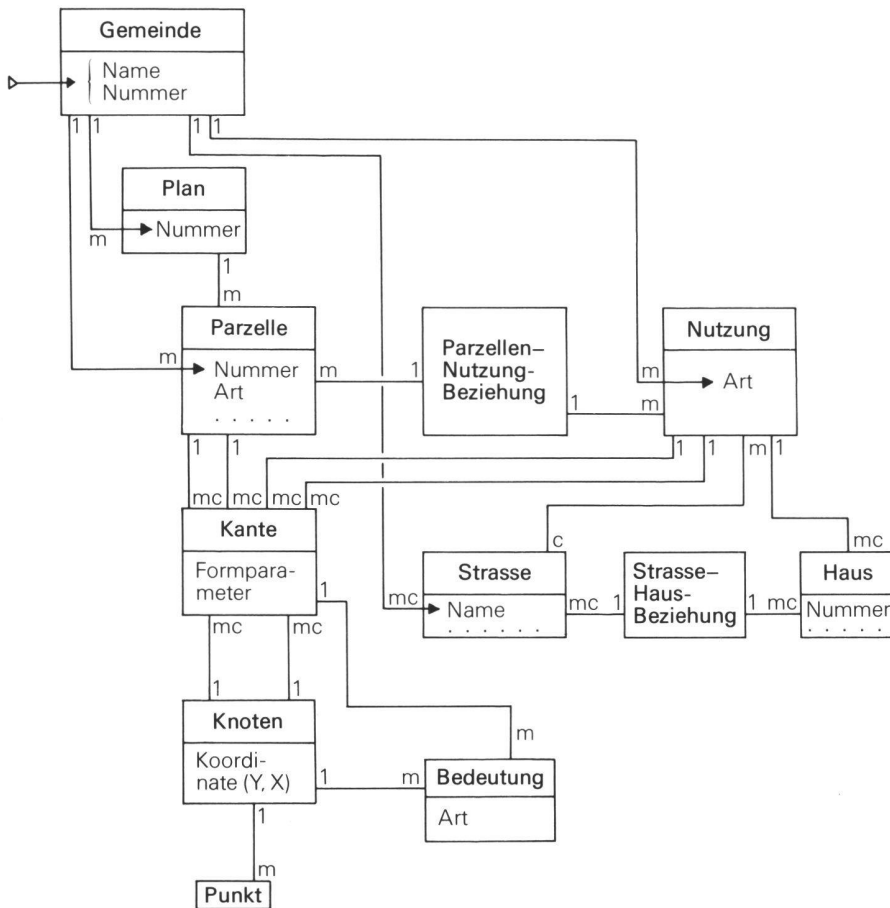


Abb. 22 Erweitertes Entitätenblockdiagramm

Beispiel: Der Benutzer der Datenstruktur in Abb. 22 möchte die (gespeicherten) Daten seines Hauses sehen. Das System liefert ihm: (Parzellennummer), (Strassenname) und (Hausnummer) sowie (Haus-Grundfläche).

### 6.1.3 Physische Ebene

Werden Datensätze auf peripheren Hardwareeinheiten abgespeichert, muss der Computer durch Einführen

von Redundanz sicherstellen, dass er später fehlerhaft gespeicherte Informationen erkennen und korrigieren kann. Schnelleres Auffinden der Datensätze erfordert auf dieser Ebene auch mehr Redundanz und damit Speicherplatz.

### 6.2 Darstellung des Zugriffs zu den Daten

Die Darstellung des Zugriffspfades (wie und in welcher Reihenfolge werden die

einzelnen Datensätze angesprochen) zeigt, auf welche Datensätze (physische Implementierung der Entitäten) der Zugriff optimal ausgerichtet ist und wie Datensätze verknüpft werden können.

### 6.3 Logik

Die gewählte Darstellungsform beschreibt die Daten und ihre Beziehungen einfach und klar. Werden die in 4.3 formulierten Vorschriften eingehalten, so sind bereits wichtige Konsistenzbedingungen erfüllt, die an die Daten gestellt werden müssen.

### 7. Resolution

Die Automationskommission vertritt die Auffassung, dass die vorgestellten Empfehlungen zur Darstellung logischer Datenstrukturen bei EDV-Programmen, die allenfalls bewilligungspflichtig würden, vorzuschreiben seien. Sie unterbreitet diese Darstellungsvorschriften der Projektleitung RAV und ihren Experten sowie der Interessengemeinschaft EDV mit der Bitte, sich intensiv damit auseinanderzusetzen und dazu Stellung zu nehmen.

### Literatur

[Bauknecht 83]: Bauknecht, K./Zehnder, C.A.: Grundzüge der Datenverarbeitung, B.G. Teubner, Stuttgart 1983

[Zehnder 83]: Zehnder, C.A.: Informationssysteme und Datenbanken, Verlag der Fachvereine, Zürich 1983

[Frank 82]: Frank, A.U.: Datenstrukturen für Landinformationssysteme – Semantische, topologische und räumliche Beziehungen in Daten der Geo-Wissenschaften, Diss. ETH Nr. 7197, 1982.

Adresse des Verfassers:

Dipl. Ing. B. Sievers

Institut für Geodäsie und Photogrammetrie  
ETH-Hönggerberg, CH-8093 Zürich

# Recommandations de la commission d'automatisation de la SSMAF à propos de la représentation des structures logiques de données

B. Sievers\*

Au début de ces recommandations, la problématique, qui est à maîtriser lors de l'analyse de la structure logique des données, est mentionnée. Ensuite d'importantes notions sont définies et leur représentation graphique expliquée. En même temps les règles, qui appuient la construction logique de la structure des données, sont énumérés. Un exemple illustre la représentation du diagramme structuré des entités. En conclusion les recommandations traitent les avantages des formes de représentations présentées.

\*B. Sievers a rédigé ces recommandations en se basant sur les délibérations de la CA SSMAF. Les bases de travail étaient données par [Zehnder 83] et [Frank 82]. La traduction française est dû à l'initiative de Messieurs A. Bercher, R. Durussel et B. Studemann. Qu'ils en soient remerciés.