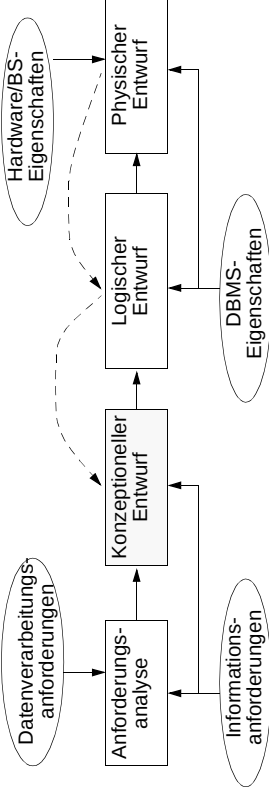


2. Konzeptioneller Datenbankentwurf

Datenorientierter Ansatz

- ☐ Welche Daten müssen im System verwaltet werden?
- ☐ Wie werden die Daten im System verändert?

Datenbankentwurfsschritte



Seite 15 von 31

Anforderungsanalyse

- ☐ basiert auf dem Wissen über Informationsstrukturanforderungen, z. B.
 - Was sind meine Objekte und deren Attribute?
 - Wie sehen die Beziehungen zwischen den Objekten aus?
 und die Datenverarbeitungsanforderungen, z. B.
 - Was sind meine typischen Operationen?
 - Laufzeit und Bedeutung der Operationen, Datenvolumen
- ☐ zentrales Problem bei der Anforderungsanalyse:
 - Entwickler einer Datenbank muß diese Informationen erst vom Benutzer der Datenbank bekommen!
 - Es gibt kein Patentrezept für eine erfolgreiche Anforderungsanalyse.
- ☐ In der Anforderungsanalyse werden im Gegensatz zu den anderen Phasen auch die Datenverarbeitungsanforderungen berücksichtigt.

Seite 16 von 31

Konzeptioneller Entwurf

- ☐ Beschreibung der Datenstrukturen in einer formalen Sprache.
- ☐ Daten in der Datenbank (Instanzen) werden nicht betrachtet.
- ☐ Transformation einer Anforderungsanalyse in einen konzeptionellen Entwurf ist schwierig:
 - Benutzer verwenden verschiedene Bezeichner für den gleichen Objekttyp
 - Benutzer verwenden den gleichen Bezeichner für verschiedene Objekttypen
 - Irrelevante Strukturen sollen weggelassen werden (Abstraktion der realen Objekte)

Seite 17 von 31

Logischer Entwurf

- ☐ Abbildung der Datenstrukturen des konzeptionellen Modells in Datenstrukturen der darunterliegenden Datenbank (z. B. Tabellen bei relationalen Datenbanken)
- ☐ Vorhandene Datenstrukturen abstrahieren von der physischen Repräsentation.
- ☐ Ziel: einmalige Speicherung von Daten (Vermeidung von Redundanz)

Physischer Entwurf

- ☐ Abbildung der Datenstrukturen des logischen Entwurfs auf Dateien.
- ☐ Welche Indexstrukturen sollen benutzt werden, um Anfragen effizient zu unterstützen?
 - zu viele Indizes: Updates der Datenbank werden zu teuer
 - zu wenige Indizes: Suchoperationen werden nicht effizient unterstützt.

Seite 18 von 31

2.1 Entity-Relationship Datenmodell

- kurz ER-Modell (Peter P. Chen: The Entity-Relationship Model - Toward a Unified View of Data, in Trans. on Database Systems 1(1): 9-36(1976))
- Das ER-Modell hat eine große Relevanz in der Praxis (nicht nur für den konzeptionellen Entwurf von Datenbanken).
- Vorgehensweise beim DB-Entwurf (siehe oben)
 - zuerst: Anforderungsanalyse und Entwurf des ER-Modells
 - dann: Umsetzung des ER Modells in ein konkreteres Modell der Datenbank
- Ziel: Modellierung eines Ausschnittes der “realen Welt” durch **Abstraktion**, so daß gewisse Fragen über die “reale Welt” mit Hilfe des Modells beantwortet werden können.
 - “*Reale Welt*”: Zunächst nur wahrnehmbar über Sinnesorgane. Menschliche *Sprache* ist bereits erster Abstraktions- und Modellierungsschritt.

ER-Modell beschreibt “reale Welt” durch

- **Objekte** (*Entities*) mit
- **Eigenschaften** (*Attributes*) und
- **Beziehungen** (*Relationships*) zueinander.

Seite 19 von 31

Entity

- existiert in der realen Welt, unterscheidet sich von anderen Entities
 - (ISBN 3-929821-31-1, Datenbanken), (Sommer, C++)
- Einzelne Entities, die “zusammengehörig” sind, werden in einer Entitymenge zusammengefaßt
 - Menge aller Bücher, Menge aller Vorlesungen, Menge aller Professoren
- zu einem Entitymenge gibt es genau einen Entitytyp
- ein Entitytyp wird durch die zugehörigen Attribute beschrieben
- Attribut eines Entitytyps beschreiben eine charakteristische Eigenschaft
 - jedes Buch besitzt eine ISBN-Nummer, einen Autor, ...
- Die Werte eines Attributes stammen aus Wertebereichen wie INTEGER, REAL, STRING
 - z. B. ist die ISBN-Nummer eines Buches ein String aus Zifferen
- Eine **minimale** Menge von Attributen, anhand deren Werte sich alle Entities einer Entitymenge unterscheiden lassen, wird als Schlüsselkandidat bezeichnet.
 - z. B. identifiziert die ISBN-Nummer das Buch

Seite 20 von 31

Relationship

- Über eine Relationship (Beziehung) lassen sich Zusammenhänge zwischen Entities herstellen.

Beispiele:

 - Student Maier hört Vorlesung DBS I
 - es gibt ein Buchexemplar zum Buch mit ISBN-Nummer 3-929821-31-1
- Eine homogene Menge von Beziehungen wird zu einer Beziehungsmenge (*Relationship-Sets*) zusammengefaßt.
 - z. B. die Beziehung Hör_Vorlesung
- zu einem Beziehungsmenge gibt es genau einen Beziehungstyp (= geordnete Liste von $n \geq 2$ Entitytypen $E_i + \text{Attribute}$)
 - n ist die Stelligkeit der Beziehung
- Bemerkungen:
 - ein Entitytyp darf in einem Beziehungstyp mehrfach vorkommen,
 - mehr als zweistellige Beziehungstypen sind möglich
 - Beziehungstypen können auch Attribute besitzen

Seite 21 von 31

Funktionalität von Beziehungstypen

- **one-to-one (1:1) - Beziehungen**
 - Falls für eine Beziehungsmenge $R(E_1, E_2)$ jedes Entity aus E_1 zu höchstens einem Entity aus E_2 in Beziehung steht und umgekehrt.
- **many-to-one (1:m) - Beziehungen**
 - Falls für eine zweistelliges Beziehungsmenge $R(E_1, E_2)$ und für alle Entities aus E_1 beliebig viele Entities aus E_2 , aber jedes Entity aus E_2 mit maximal einem Entity aus E_1 in Beziehung steht.
- **many-to-many (m:n) - Beziehungen**
 - Liegt keine Einschränkung vor, spricht man von *m:n-Beziehungsmengen*

Seite 22 von 31

Beispiele

- ☐ ABTEILUNG wird geleitet von ANGESTELLTER
Annahme: jede Abteilung hat genau einen Leiter und kein Angestellter leitet mehr als eine Abteilung.
- ☐ ANGESTELLTER arbeitet in ABTEILUNG
Annahme: jeder Angestellte arbeitet in genau einer Abteilung.
- ☐ ANGESTELLTER führt ZUG

Einige Besonderheiten:

- ☐ in Beziehungstypen, in denen Entitytypen mehrfach vorkommen, werden entsprechende Rollen vergeben.
 - Beispiel: VORGESETZTER(CHEF, A_ANG) wobei CHEF UND A_ANG Rollen für den Entitytyp ANGESTELLTER sind.
- ☐ einige Beziehungstypen sind sogenannte Spezialisierungen: IS-A-Beziehungen
 - ANGESTELLTER besitzt Attribute ANGNR, NAME und GEHALT
 - ZUGFUEHRER und TECHNIKER sind auch ANGESTELLTER, besitzen aber noch weitere Attribute

2.1.1 ER-Diagramm

- ☐ graphische Repräsentation der Entitytypen, Beziehungstypen und ihrer Attribute durch einen Graphen

folgende Vereinbarungen werden getroffen:

- ☐ Rechtecke repräsentieren Entitytypen: 
- ☐ Ellipsen repräsentieren Attribute: 
 - Sie sind über ungerichtete Kanten mit ihrem Entitytyp verbunden.
 - Schlüssel-Attribute werden unterstrichen.
- ☐ ein Beziehungstyp wird durch eine Raute  repräsentiert:
 - Beziehungstypen werden mit ihren Entitytypen durch Kanten verbunden.
 - Die Kanten sind ungerichtet, außer:
 - a) Bei one-one sind beide Kanten gerichtet.
 - b) Bei many-one ist die Kante zum hierarchisch abhängigen Entitytyp hin gerichtet.
- ☐ Eine Rolle eines Beziehungstyps wird an der entsprechenden Kante notiert:

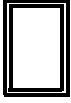


2.1.2 Erweiterungen

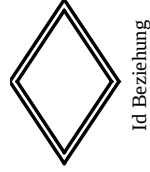
Id-Beziehungen

- ☐ Id Beziehungen sind spezielle 1:n Beziehungen, wobei die Existenz eines Entities von einem anderen Entity abhängt.
 - Man bezeichnet dann auch den existenzabhängigen Entitytyp als schwach und den anderen Entitytyp als stark.

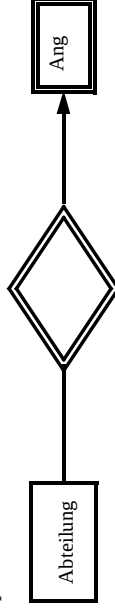
- ☐ Graphische Notation



schwacher Entity

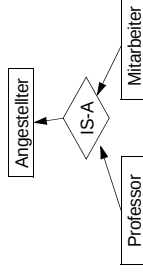


- ☐ Beispiel:



IS-A Beziehungen (Typerweiterung)

- ☐ Ein Entity vererbt **alle** seine Eigenschaften an einen anderen Entity. Die Beziehung zwischen den Entitytypen wird auch als IS-A Beziehung bezeichnet.
- ☐ IS-A Beziehung wird für die Partitionierung einer Menge in (disjunkte) Teilmengen verwendet. Beide Entities einer IS-A Beziehung besitzen den gleichen Schlüssel.
- ☐ Beispiel
 - ANGESTELLTER besitzt Attribute ANGNR, NAME und GEHALT
 - Professoren sind auch ANGESTELLTER, besitzen aber noch weitere Attribute wie z. B. FACHGEBIET



- ☐ Einige Besonderheiten
 - Es gibt höchstens nur einen IS-A oder Id Beziehungstyp zwischen zwei Entitytypen

(min,max)-Notation

- ☐ für jeden an einem Beziehungstyp beteiligten Entitytyp bezeichnet
 - *min*: jedes Entity dieses Typs steht mindestens min-mal in Beziehung
 - *max*: jedes Entity dieses Typs steht höchstens max-mal in Beziehung
- ☐ Sonderfälle
 - $\min = 0$: ein Entity braucht nicht eine Beziehung einzugehen
 - $\max = *$: ein Entity darf beliebig oft an einer Beziehung beteiligt sein
- ☐ Beispiel:
 - ein Zug besitzt mindestens einen und höchstens 10 Wagen
 - ein Wagen gehört zu mindestens einem Zug und zu höchstens einem Zug.
- ☐ !! ACHTUNG !!
 - (min,max)-Notation und Funktionalitätsbegriff entsprechen sich nicht.

2.2 Entwurf eines ER-Diagramms

an Hand des Beispiels eines Auskunfts- und Buchungssystem für DB-ICE-Netz

Mögliche Anfragen (Datenverarbeitungsanforderungen)

- ☐ wann fährt ein Zug von München nach Bremen?
- ☐ reserviere ein Platz von München nach Frankfurt im Zug ICE7345 am 6.5.93.
- ☐ gib Liste der reservierten Plätze in Zug ICE7345 am 6.5.93.
- ☐ gibt es eine Verbindung von München nach Essen mit Abfahrt zwischen 8.00 und 10.00 (ohne umzusteigen)?

Informationsanforderungen:

Entitytypen:

- ☐ Zug
- ☐ Wagen
- ☐ Platz
- ☐ Bahnhof

Attribute:

- ☐ Zugnr, Name, Verkehrstage
- ☐ Wagennr., Klasse, Platzanzahl
- ☐ Platznr., R/NR
- ☐ Name, Umsteigebahnhof

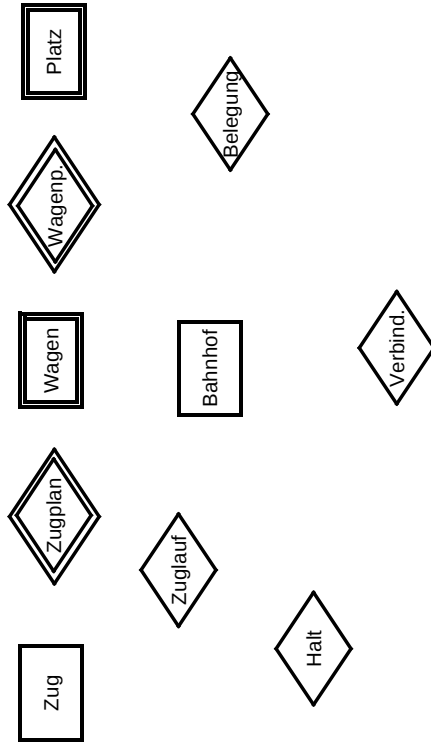
Beziehungstypen:

- ☐ Zugplan: Zug x Wagen
- ☐ Wagenplan: Wagen x Platz
- ☐ Belegung: Platz x Bahnhof x Bahnhof
- ☐ Halt: Zug x Bahnhof
- ☐ Zuglauf: Zug x Bahnhof x Bahnhof x Bahnhof
- ☐ Verbindung: Bahnhof x Bahnhof x Bahnhof

Einschränkungen:

- ☐ ein Platz kann nicht auf überlappenden Teilstrecken reserviert werden
- ☐ die Zeiten in den Beziehungstypen Halt und Verbindung müssen übereinstimmen

ER-Diagramm



Zusammenfassung

- ☐ Datenbankentwurf ist ein äußerst komplexer Vorgang
- ☐ ER-Modellierung ist ein auch in der Industrie anerkanntes und weit verbreitetes Verfahren zur Datenmodellierung
 - Voraussetzung ist aber bereits eine Anforderungsanalyse
 - Beachte, daß das ER-Modell von dem DBMS *unabhängig* ist.
- ☐ wichtige Komponenten:
 - Entity, Entitymenge, Entitytyp
 - Beziehung, Beziehungsmenge, Beziehungstyp.
- ☐ Charakterisierung von Beziehungstypen
 - n:m, 1:m, 1:1-Beziehungen
 - is-a Beziehung, has-a Beziehung
 - “starke” Beziehungen
- ☐ Frage:
Wie kann das ER-Modell in eine Datenbank abgebildet werden?