



## Das Entity-Relationship-Modell (ERM)

Um Ausschnitte aus der realen Welt anschaulich modellieren zu können, hat sich Peter Chen bereits 1979 das Entity-Relationship-Modell (dt. Gegenstands-Beziehungs-Modell) ausgedacht.

Die Modellierung der Daten erfolgt, abhängig von der gestellten Aufgabe, in mehreren Schritten.

**Beispielaufgabe:** Lehrkräfte und ihre Fächer sollen in einer Datenbank erfasst werden.

### Entitäten und Entitätstypen

Erkenne die für dein Projekt relevanten **Entitäten** (engl. *entities*): Das sind **einzelne Datenobjekte**, wie beispielsweise Lehrer „Kannix“, Fach „Deutsch“, Lehrerin „Weissalles“, Fach „Englisch“ oder Fach „Sport“.

**Gleichartige Entitäten** werden zu einem **Entitätstyp** zusammengefasst.

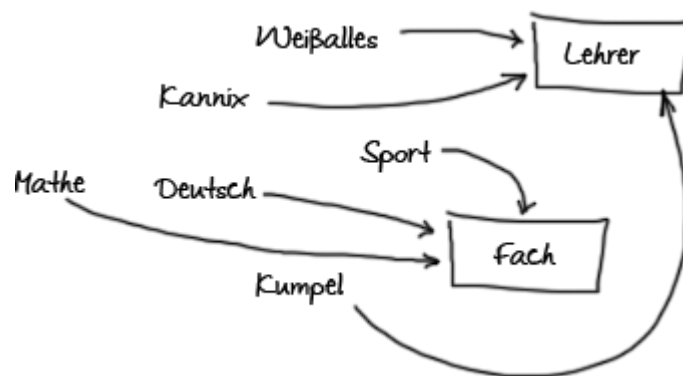


Abb. 01: Zusammenfassen von Entitäten zu Entitätstypen

Grafisch wird ein **Entitätstyp** durch ein **Rechteck** dargestellt. Bei der Bezeichnung ist es weithin üblich, die **Einzahl** zu verwenden (Fach, und nicht Fächer).

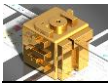
In der Praxis findet man auch Datenmodelle, in denen die Bezeichnungen der Entitätstypen in der Mehrzahl stehen. Das hat aber auf das Verständnis des ERM-Modells wenig Einfluss.

Die Verwendung der **Mehrzahl** bei **Relationen** und **Tabellenbezeichnungen** ist jedoch sinnvoll, weil es sich hierbei um **Entitätsmengen** (engl. *entity sets*) handelt. Das sind Zusammenfassungen einer bestimmten Anzahl von gleichartigen Entitäten (Datensätzen, Objekten), die sich jeweils nur durch ihre Attributwerte unterscheiden.

### Attribute

Bestimme wichtige **Attribute** (Eigenschaften) für jeden gefundenen Entitätstyp. Finde das **identifizierende Attribut** für jeden Entitätstyp. Das ist das Attribut, das jede Entität eines Typs eindeutig beschreibt. Dazu hast du drei Möglichkeiten:

- Ein vorhandenes Attribut ist bereits eindeutig, z. B. eine Personalnummer.
- Du kombinierst mehrere Attribute, wie z. B. Geburtsdatum und Nachname.
- Du nimmst ein künstliches, numerisches Attribut, etwa eine Lehrer- oder Fachnummer.



## 2.3.2 Datenbanksysteme II

### Lerninhalte 232-04 Das Entity-Relationship-Modell

Das Attribut *LehrerNR* beispielsweise unterscheidet eindeutig einen Lehrer vom anderen. Keine zwei Lehrer haben für dieses Attribut den gleichen Attributwert.

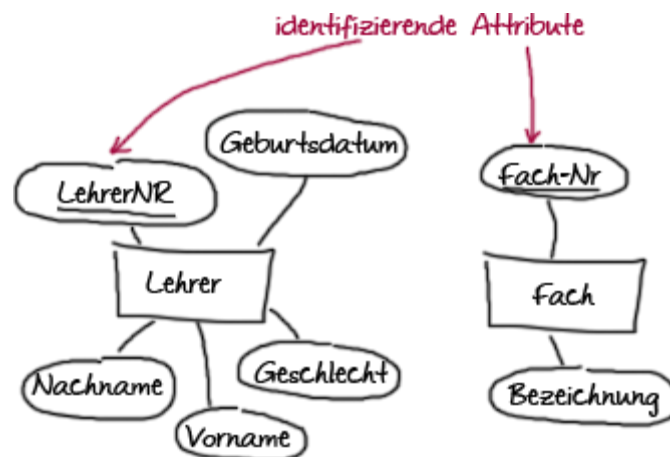


Abb. 02: Bestimmung notwendiger Attribute für die Entitätstypen

**Attribute** werden grafisch durch **Ellipsen** dargestellt. Die Bezeichnungen identifizierender Attribute unterstreicht man.

**Identifizierende Attribute** werden auch **Primärschlüssel** genannt.

### Beziehungen und Beziehungstypen

Die Beziehungen zwischen Objekten sind sehr wichtig. Wenn wir die Beziehungen kennen, können wir später die verschiedenen Tabellen unserer Datenbank verbinden.

Zunächst identifiziert man die **Beziehungen** (engl. *relationships*) zwischen zwei oder mehreren Entitäten, wie beispielsweise:

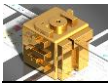
- „Lehrer Kannix unterrichtet Deutsch“
- „Lehrer Kannix unterrichtet Englisch“
- „Deutsch wird von Lehrer Kumpel unterrichtet“

Dann fasst man **gleichartige Beziehungen** zu einem **Beziehungstyp** zusammen:



Abb. 03: Festlegen des passenden Beziehungstyps

**Beziehungstypen** werden grafisch durch **Rauten** abgebildet.



#### Kardinalität

Die Kardinalität eines Beziehungstyps gibt an, mit wie vielen anderen Entitäten eine Entität eines bestimmten Entitätstyps in einer konkreten Beziehung stehen kann.

Zur **vereinfachten** Darstellung der **Kardinalitäten** im ERM werden die Ziffer 1, im Sinne von 0 oder 1, und die Buchstaben m und n, im Sinne von 0 bis unendlich verwendet.

#### 1:1

Bei dieser Kardinalität ist jeweils genau eine Entität höchstens einer anderen zugeordnet.



Abb. 04: Beispiel 1:1-Beziehung

#### 1:n (n:1)

Einer Entität sind keine, eine oder mehrere Entitäten auf der anderen Seite der Beziehung zugeordnet. (n:1 ist einfach ein von rechts nach links gelesenes 1:n und wird daher sehr selten angegeben)



Abb. 05: Beispiel 1:n-Beziehung

#### m:n

Auf beiden Seiten können beliebig viele Entitäten miteinander in Beziehung stehen.



Abb. 06: Beispiel m:n-Beziehung

Bei der Modellierung besteht der Zusammenhang zwischen zwei Entitätstypen in beiden Richtungen. Man muss also eine Beziehung immer von beiden Seiten her betrachten.

#### Beispiel 1:n

- Ein Schüler kann (noch) kein, ein oder mehrere Schulbücher haben ( $\rightarrow n$ ).
- Ein Schulbuch kann an keinen oder höchstens einen Schüler verliehen werden ( $1 \leftarrow$ ).

Wie sieht nun die **Kardinalität** in unserem Beispiel mit den Lehrern und ihren Fächern aus?

#### Leserichtung $\rightarrow$

Als erstes müssen wir die Frage zu beantworten, ob ein Lehrer nur ein oder auch mehrere Fächer unterrichten kann. In der Regel ist dies eindeutig zu beantworten. Jeder Lehrer kann mehr als ein Fach unterrichten ( $\rightarrow n$ ).

#### Leserichtung $\leftarrow$

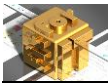
Umgekehrt kann man sagen, dass ein Fach von einem oder mehreren Lehrern an der Schule unterrichtet werden kann ( $n \leftarrow$ ).

#### Ergebnis:

Die Beziehung zwischen den Entitätstypen **Lehrer** und **Fach** besitzt also die Kardinalität **m:n** (sprich: m zu n).



Abb. 07: Kardinalität des Beziehungstyps 'unterrichtet'



#### Fertiges Datenmodell

Das grafische Entity-Relationship-Modell für unsere Beispielaufgabe „Lehrer unterrichten Fächer“ sieht nun so aus:



Abb. 08: Fertig modelliertes Entity-Relationship-Modell (ERM)

Im Falle eines **n:m-Beziehungstyps** kann aus dem **Beziehungstyp** ein **eigenständiger Entitätstyp** mit Beziehungstypen (jeweils 1:n) zu den ursprünglich beteiligten Entitätstypen geschaffen werden. Dem neuen Entitätstyp können dann auch Attribute zugeordnet werden.

1:n-Beziehungstypen haben hingegen üblicherweise keine Attribute, da diese immer einem der beteiligten Entitätstypen zugeordnet werden können.

#### Überführung in ein relationales Modell

Die Überführung des ERM in das relationale Modell erfolgt vereinfacht nach diesen Transformationsregeln:

##### 1) Entitätstypen

Für jeden Entitätstyp wird eine Relation (**Tabelle!**) mit einem Primärschlüssel und den anderen Attributen der Entität erstellt.

##### 2) m:n-Beziehungstypen

Für jeden m:n-Beziehungstyp wird eine neue Relation mit den Attributen der Beziehung, sowie für die Primärschlüssel der beteiligten Relationen erstellt.

##### 3) 1:n-Beziehungstypen

Für den 1:n-Beziehungstyp wird die mit der Kardinalität n eingehende Relation um den Fremdschlüssel<sup>1</sup> der anderen Relation (Kardinalität 1) erweitert.

##### 4) 1:1-Beziehungstypen

Für einen 1:1-Beziehungstyp wird eine der beiden Relationen um den Fremdschlüssel für die jeweils andere Relation erweitert.

<sup>1</sup> Ein Fremdschlüssel verweist auf einen Primärschlüssel einer anderen Relation. Beide Relationen benötigen ein gemeinsames Attribut vom selben Datentyp. Man bezeichnet die Tabelle, auf deren Primärschlüssel verwiesen wird, als **Mastertabelle**. Die Tabelle, die den Fremdschlüssel enthält, bezeichnet man als **Detailtabelle**.