

Arbeiten mit einer Datenbank¹

1. Datenmodelle

1.1 Das Entity-Relationship-Model (Objekt-Beziehungs-Modell)

Bevor man in einem Datenbanksystem eine Datenbank aufbaut, muss man sich die Struktur der Datenbank ausdenken (Modellierung). Das Entity-Relationship-Model ist eine systematische Methode für den Übergang von den Dingen der realen Welt in die Strukturen einer Datenbank. Das Besondere am ERM ist, dass es unabhängig vom verwendeten Datenbanksystem ist.

1.1.1 Entität (Objekttypen)

Eine **Entität (Entity)** ist ein reales oder abstraktes Objekt der realen Welt, das über **Attribute (Eigenschaften, Merkmale)** verfügt. Allgemeine Beispiele was zu Entitäten modelliert werden kann:

- Personen und Institutionen
- Sachen
- Vorgänge

Im Datenmodell wird eine Entität (Objekt) eindeutig durch die Attribute (Eigenschaften) beschrieben, d. h. zwei Entitäten müssen sich in mindestens einem Attribut unterscheiden.

Entitäten werden in ER-Diagrammen als Rechtecke dargestellt, Attribute als Ovale und die Verbindungen der Entitätstypen mit ihren Attributen als Linien.

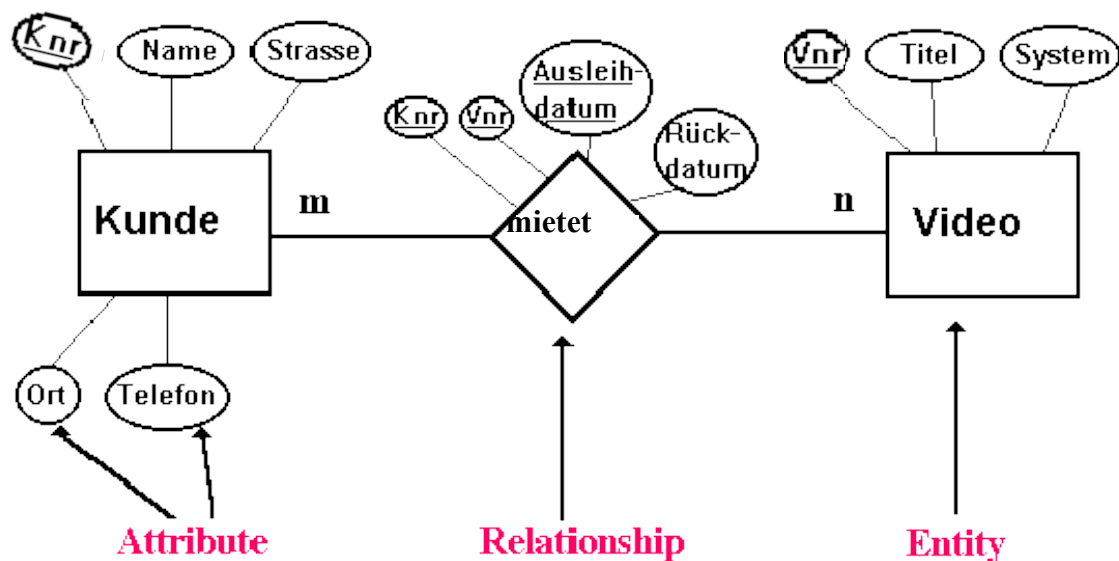
Eine Teilmenge der Attributmenge, welche die Entitäten eindeutig identifiziert heißt **Primärschlüssel**. Oft wird hierfür ein künstliches Attribut (z.B. PersonenNummer) verwendet, da eine Person nur durch Nach- und Vornamen nicht eindeutig identifiziert ist.

1.1.2 Beziehungen zwischen Entitäten (Relationen)

Die Beziehungen zwischen Entitäten werden in ER-Diagramm als Rauten dargestellt. Die Beziehungen werden durch Linien angebunden. Entitäten können immer nur über Relationen miteinander verbunden werden! Eine Relation verbindet zwei Entitäten miteinander². Zwischen zwei Entitäten steht somit immer eine Relation!

Beispiel für ein ER-Modell:

<http://www.oszhdl.be.schule.de/gymnasium/faecher/informatik/datenbanken/modell/erm.htm>



¹ Material basiert auf der Webseite des OSZ Handel:

<http://www.oszhdl.be.schule.de/gymnasium/faecher/informatik/datenbanken/index.htm>

² In den Beispielen verbindet eine Relation immer nur zwei Entitäten. Im ERM besteht für eine Relation allerdings grundsätzlich die Möglichkeit auch mehrere Entitäten verbinden!

1.1.3 Beziehungstypen

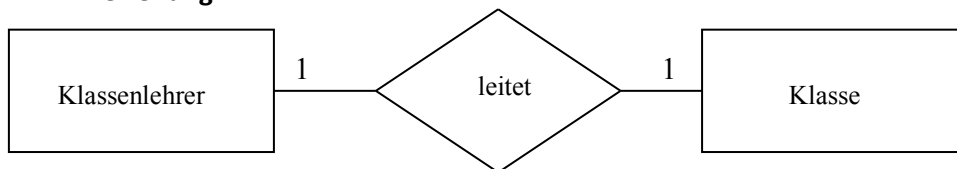
Der Beziehungstyp (**Kardinalität**) beschreibt die numerischen Zusammenhänge zwischen den einzelnen Entitäten der verschiedenen Entitätstypen.

Dabei werden drei Grundtypen unterschieden:

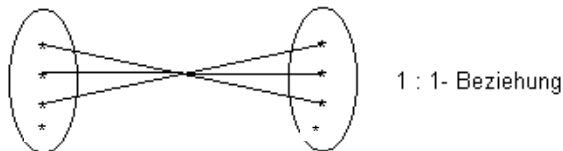
- 1 : 1
- 1 : n
- n : m

Beispiele:

1 : 1 – Beziehung:



Ein Klassenlehrer leitet genau eine Klasse **und** eine Klasse hat genau einen Klassenlehrer.

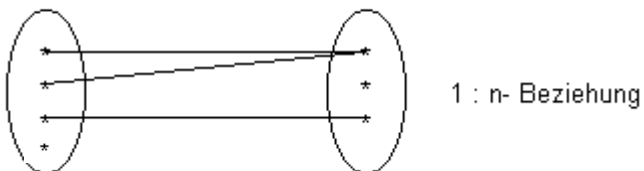


3

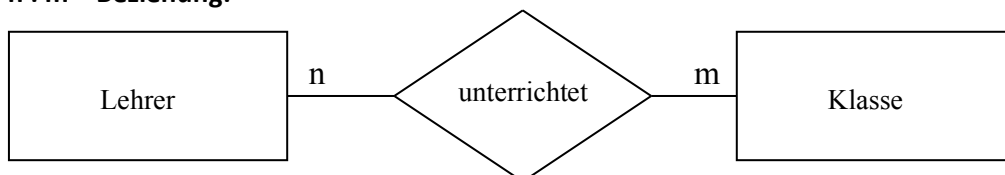
1 : n – Beziehung:



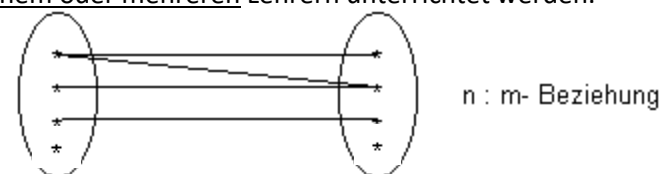
Eine Klasse besteht aus einem oder mehreren Schülern **und** ein Schüler gehört zu genau einer Klasse.



n : m – Beziehung:



Ein Lehrer kann eine oder mehrere Klassen unterrichten **und** eine Klasse kann von einem oder mehreren Lehrern unterrichtet werden.



³ Quelle der Mengenabbildungen: http://erp.wu-wien.ac.at/download/erm_d/gr/grundlagen.html

1.2 Das relationale Datenmodell

Das relationale Datenmodell ist die theoretische Grundlage für relationale Datenbanken wie z.B. MySQL, BASE (LibreOffice) oder ACCESS (Microsoft).

Das ER-Modell lässt sich ganz einfach in das relationale Tabellenmodell überführen, das dann fast direkt in eine relationalen Datenbank implementiert (hier: überführt) werden kann.

1.2.1 Tabellen (Relationen)

- In **relationalen Datenbanken** werden alle Informationen in Form von **Tabellen (Relationen)** abgelegt.
- Die **Spaltenbezeichnungen** enthalten die **Attribute** (Merkmale).
- Die unter den Spaltenbezeichnungen aufgeführten Werte sind die **Attributwerte** (Merkmalswerte).
- In einer **Zeile** der Tabelle steht dann zu **jedem Attribut (Merkmal)** jeweils **ein** Wert.
- So eine Tabellenzeile stellt **einen Datensatz** dar, d. h. Merkmale werden zu Datensätze zusammengefasst.

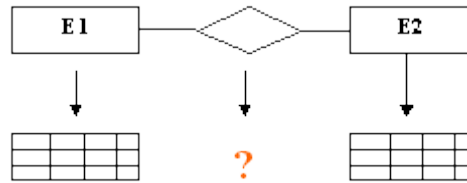
Für eine Tabelle einer relationalen Datenbank müssen folgende Bedingungen gelten:

1. Jede Zeile der Tabelle tritt nur einmal auf, d. h. zwei Zeilen **unterscheiden** sich also **mindestens** in einem Attributwert (Merkmalswert).
Jeder Datensatz muss eindeutig identifiziert werden können. Dazu dient ein **Primärschlüssel**.
Am besten ist ein künstlicher Primärschlüssel wie eine Kundennummer, die nur einmal vergeben wird.
2. Alle Attributwerte (Merkmalswerte) sind nicht weiter zerlegbar, sie sind **atomar**.
In der Tabelle treten keine **Attributwerte** wie etwa Adressen (nicht atomar) auf, da diese selbst wieder aus Straße (atomar), Hausnummer (atomar), Postleitzahl (atomar) und Ort (atomar) bestehen würden.

1.3 Transformation von ERM in das relationale Modell

Quelle: <http://www.oszhdl.be.schule.de/gymnasium/faecher/informatik/datenbanken/modell/abbild.htm>

Die Transformation des ER-Modells in Relationen ist ein mehr technischer Prozess, der nach festen Regeln abläuft und der heutzutage auch bereits von Werkzeugen übernommen werden kann. Bei den Entitäten ist es klar - sie werden in Relationen (Tabellen) überführt. Bei den Beziehungsmengen (Relationship) sind einige Besonderheiten in Abhängigkeit von der Konnektivität zu beachten.



Transformationsregeln:

1. Regel:



Jede Entität wird als Tabelle dargestellt.

2. Regel



Jede $n : m$ - Beziehung wird durch eine eigene Tabelle dargestellt.

Die Beziehung wird dadurch hergestellt, dass die **Primärschlüssel** von E1 und E2 als **Fremdschlüsselattribute** in der Beziehungsrelation aufgenommen werden.

Beispiel: Einem Projekt gehören mehrere Mitarbeiter an, die auch in mehreren verschiedenen Projekten mitarbeiten.

PERSONAL (**P-Nr**, Name)

PROJEKT (**Pro-Nr**, Bezeichnung, Termin)

ZUGEHÖRIGKEIT (**P-Nr**, **Pro-Nr**)

3. Regel



Jede Beziehung, die eigene Attribute hat, wird durch eine eigene Tabelle dargestellt.

4. Regel



Ist eine Entität E2 **zwingendes** Mitglied (d. h. genau 1) einer $1 : n$ - Beziehung mit einer Entität E1, dann erhält E2 den Primärschlüssel von E1 als Attribut. (Zwei Tabellen)

Sofern eine Entität E2 **freies** Mitglied (d. h. 0,1 - konditionell) der $1 : n$ - Beziehung mit einer Entität E1 ist, wird diese Beziehung gewöhnlich in einer eigenen Relation dargestellt. (Drei Tabellen)

Diese Bedingungen gelten für $1:1$ und $1:n$ - Beziehungen.

Beispiele für Regel 4:

Zwingendes Mitglied am Beispiel Abteilungszuordnung:

Eine Mitarbeiterin gehört zwingend genau einer Abteilung an und eine Abteilung hat mehrere Mitarbeiterinnen.



Da hier eine Entität (Mitarbeiterin) von E2 zwingendes Mitglied einer Entität aus E1 (Abteilung ist), wird die Beziehung in den Tabellen dadurch hergestellt, in dem der Primärschlüssel von E1 (A-Nr) in E2 als Fremdschlüsselattribut aufgenommen wird.

ABTEILUNG (A-Nr, A-Name)
PERSONAL (P-Nr, Name, A-Nr)

Freigestelltes Mitglied am Beispiel Betriebssportgemeinschaft:

Eine Mitarbeiterin kann muss aber nicht Mitglied genau einer Betriebssportgemeinschaft sein.



Da hier eine Entität aus E2 (Mitarbeiterin) freies Mitglied der Entität E1 (BSG) ist, wird die Beziehung in einer eigenen Tabelle dargestellt und die Primärschlüssel von E1 und E2 werden als Fremdschlüssel eingetragen.

BSG (BSG-Nr, BSG-Name)
PERSONAL (P-Nr, Name)
MITGLIED (BSG-Nr, P-Nr)

(nach Material von J. Penon)