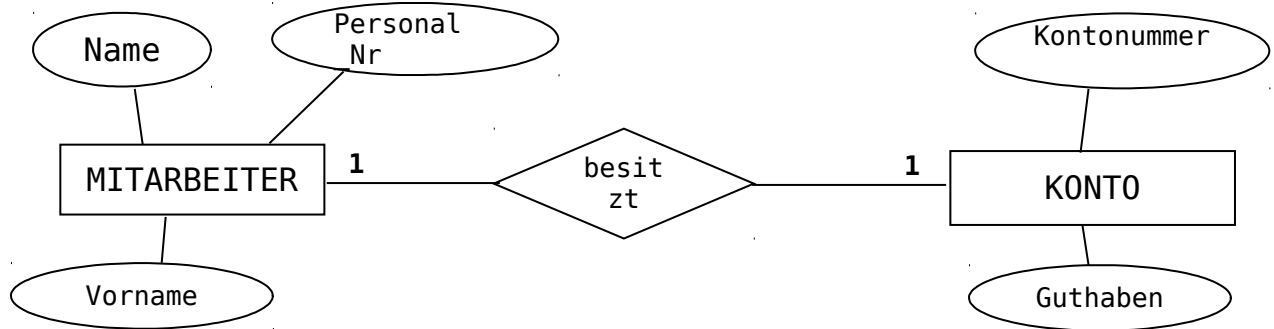


Qualifizierung und Typisierung von Beziehungen (Kardinalitäten)

Zwei Entitätstypen stehen meist in einer logischen Beziehung zueinander. Hierbei stellt sich dann die Frage, zu wie vielen Elementen der jeweils anderen Entitätsmenge ein Element bzw. eine konkrete



Entität in Beziehung stehen kann.

a) 1:1-Beziehung 1: genau 1

Jeder einzelne Mitarbeiter einer Firma hat genau ein Gehaltskonto. Jedes Gehaltskonto lässt sich genau einem Mitarbeiter zuordnen (Ausnahmen bestätigen die Regel!).

Eine der beiden Tabellen (**T_KONTOVERBINDUNGEN**) muss als Primärschlüssel den Fremdschlüssel der anderen Tabelle erhalten

T_MITARBEITER

<u>P_Personal_Nr</u>	Name	Vorname
----------------------	------	---------

T_KONTEN

<u>PF_Personal_Nr</u>	Kontonummer	Guthaben
-----------------------	-------------	----------

oder

T_MITARBEITER

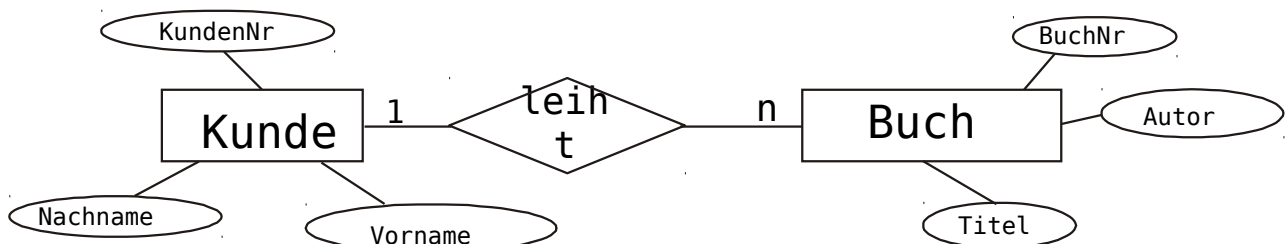
<u>PF_Kontonummer</u>	Personal_Nr	Vorname	Name
-----------------------	-------------	---------	------

T_KONTEN

<u>P_Kontonummer</u>	Guthaben
----------------------	----------

b) 1:n-Beziehung n: 1 oder viele

Ein Kunde kann mehrere Bücher ausleihen. Hinter jeder Buchausleihe steckt aber nur genau ein



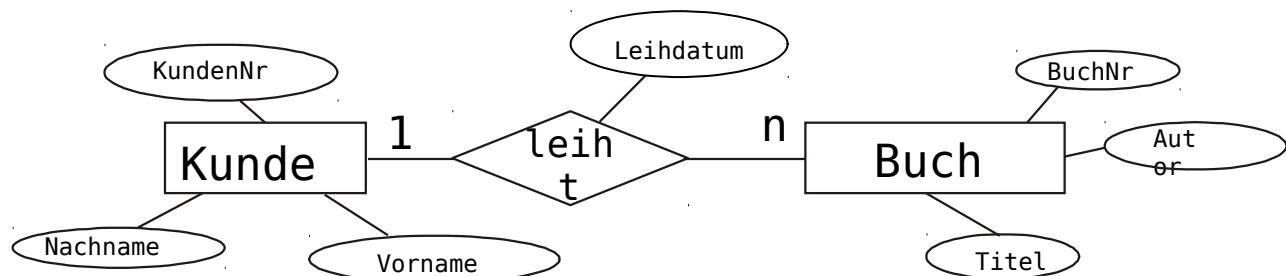
Kunde. Die Tabelle T_BÜCHER enthält neben dem Fremdschlüssel und dem Primärschlüssel als zusätzliche Attribute auch die Spalten *Titel*, und *Autor*!

T_KUNDEN

<u>P_Kunden_Nr</u>	Name	Vorname
--------------------	------	---------

T_BUECHER

<u>P_Buch_Nr</u>	F_Kunden_Nr	Autor	Titel
------------------	-------------	-------	-------



Auflösung bzw. Realisierung von Beziehungen mit Attributen

Bei der Auflösung einer Beziehung mit Attributen gibt es in der Regel zwei gleichwertige Möglichkeiten. Zum einen kann das Attribut mit in die Tabelle der n-Seite genommen werden. Zum anderen kann eine Zwischentabelle erstellt werden.

Variante 1:

T_KUNDEN

<u>P_Kunden_Nr</u>	Name	Vorname
--------------------	------	---------

T_BUECHER

<u>P_BuchNr</u>	F_Kunden_Nr	Autor	Titel	Leihdatum
-----------------	-------------	-------	-------	-----------

Variante 2:

T_KUNDEN

<u>P_Kunden_Nr</u>	Name	Vorname
--------------------	------	---------

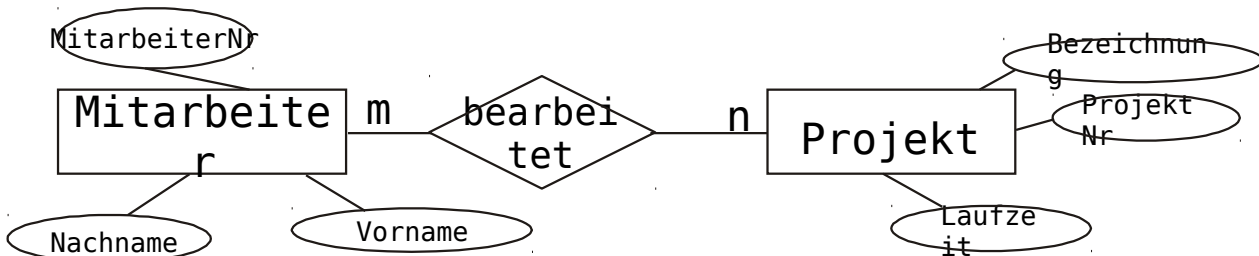
T_BUECHER

<u>P_BuchNr</u>	Autor	Titel
-----------------	-------	-------

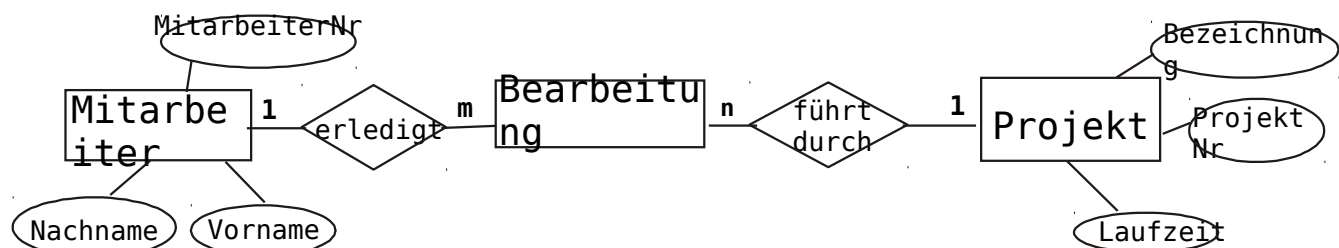
T_AUSLEIHEN

<u>PF_BuchNr</u>	F_Kunden_Nr	Leihdatum
------------------	-------------	-----------

c) m:n-Beziehung



Einzelne Mitarbeiter können mehrere Projekte bearbeiten. Jedes einzelne Projekt hat verschiedene Mitarbeiter. Relationale Datenbanksysteme (z.B. MySQL) können keine m:n-Beziehung wie diese realisieren:



Dann muss (manchmal künstlich) ein Bindeglied-Entitätstyp geschaffen werden, so dass die m:n-Beziehung durch eine 1:m- und eine n:1-Verbindung realisiert werden können.

Durch den **neuen künstlichen Entitätstyp** BEARBEITUNG kann die m:n-Verbindung auch in MySQL realisiert werden! Ein Mitarbeiter sorgt für die Bearbeitung mindestens eines Projektes mit genau einem Mitarbeiter. Eine Bearbeitung bezieht sich genau auf einen Projekt und einen Mitarbeiter. Zu jedem Projekt sollte es aber mehrere Bearbeitungen, sprich Mitarbeiter geben. Die entsprechenden Tabellen sehen wie folgt aus:

Name:

Datum:

Klasse:

Blatt Nr.: 3/3

Lfd. Nr.:

T_MITARBEITER

<u>P_MitarbeiterNr</u>	Name	Vorname
------------------------	------	---------

T_PROJEKTE

<u>P_ProjektNr</u>	Bezeichnung	Laufzeit
--------------------	-------------	----------

T_MITARBEITER_PROJEKTE

<u>PF_MitarbeiterNr</u>	<u>PF_ProjektNr</u>
-------------------------	---------------------

Der zusammengesetzte

Primärschlüssel besteht aus zwei Fremdschlüsseln, sprich den Primärschlüsseln der zu verknüpfenden Tabellen. Kommt kein weiteres Attribut hinzu, sollte man die Tabelle statt T_BEARBEITUNGEN

T_MITARBEITER_PROJEKTE nennen, damit die künstliche Entitätsmenge sofort ersichtlich ist!