

LB-DB 2 - 28.2.2015

Dipl.-Ing. Reinhard Schlager

its
FH Salzburg

2015/ LB-Datenbanksysteme

Gliederung

- 1 Begriffe
 - Schlüssel
- 2 ERM
 - 1 zu n
 - n zu m
 - Von ERM zu den Tabellen
- 3 SQL
 - Der Join
 - Der Join - Beispiel
 - UNION ALL
 - IN - NOT IN
- 4 Übung 2 - Presse Datenbank

Schlüssel

Primärschlüssel

Primärschlüssel

Identifiziert jeden Tupel (Record) einer Relation (Tabelle) eindeutig. Mit Hilfe des Primärschlüssels selektiert man in einer relationalen Datenbank einen Datensatz eindeutig und weil dem Primärschlüssel ein Index zugrunde liegen sollte auch schnell. Das wichtigste und erstrangig verwendete Selektionskriterium, um Daten in einer Tabelle zu selektieren

Schlüssel

persnr	VorName	NachName	Adresse
1	Franz	Müller	S.Marcus Str.24
2	Fritz	Maier	G.Liebherrg. 12
3

Arten von Schlüsseln

Natürliche Schlüssel

Sozialversicherungsnummer

ISBN Nummer Vorname + NachName +Geburtsdatum ??

Künstliche Schlüssel (Surrogat)

Kundennummer

Auftragsnummer

Vorgangnummer

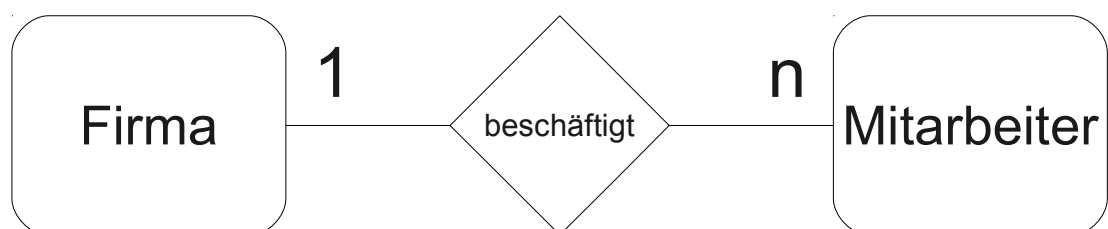
Rechnungsnummer

1 zu n Beziehung

1 zu n Beziehung

Eine Firma beschäftigt mehrere (n) Mitarbeiter

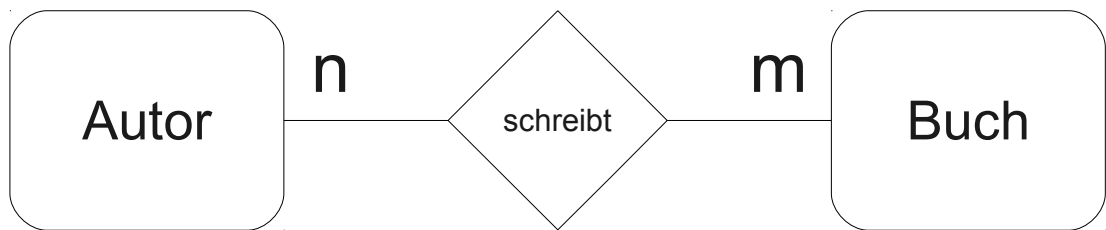
Ein Mitarbeiter ist in *einer* Firma beschäftigt.



n zu m Beziehung

n zu m Beziehung

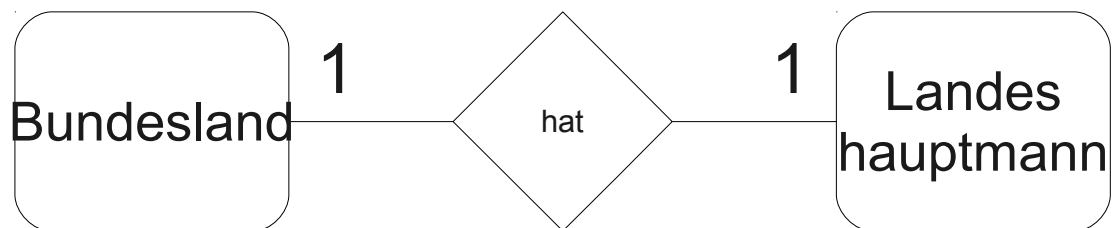
Eine Autor schreibt mehrere (m) Bücher
Ein Buch kann von n Autoren geschrieben worden sein.



1 zu 1 Beziehung

1 zu 1 Beziehung

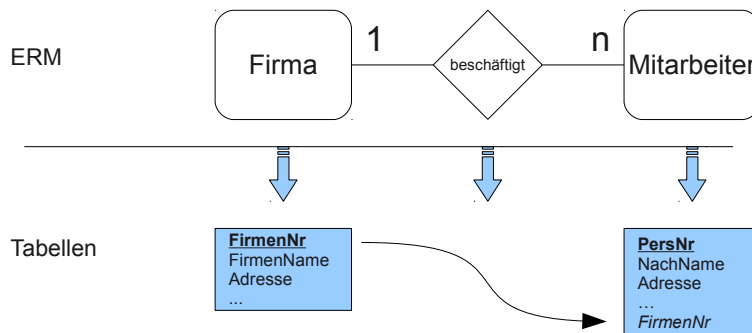
Eine Bundesland hat *einen* Landeshauptmann
Ein Landeshauptmann ist für *ein* Bundesland verantwortlich.



1:n Von ERM zu den Tabellen

1:n

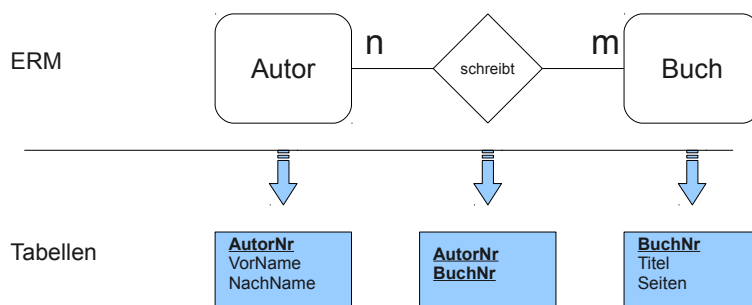
Eine 1:n Beziehung wird mit zwei Tabellen abgebildet. Der Primärschlüssel der 1 Seite wird zum Fremdschlüssel der n Seite.



n:m Von ERM zu den Tabellen

n:m

Eine n:m Beziehung wird mit drei Tabellen abgebildet. Die dritte Tabelle besteht aus den Primärschlüsseln der beteiligten Relationen.



Der Join

Join mit 2 Tabellen

Möchte man *eine* Liste aus *zwei* Tabellen, die miteinander in Beziehung stehen, erzeugen muss man diese Tabellen durch Schlüssel und Fremdschlüssel miteinander joinen.

```
SELECT  tab1.att1, tab1.att2, tab2.att1
FROM    tab1, tab2
WHERE   tab1.fk = tab2.pk
```

Lässt man die WHERE Klausel weg, enthält die Liste *alle* möglichen Kombinationen der beiden Tabellen. Wenn tab1 20 und tab2 40 Datensätze enthält, wäre das Ergebnis eine Liste von $20 \times 40 = 800$ Sätzen.

Der Join - Beispiel

Join mit 2 Tabellen - Beispiel

```
SELECT  journalist.vorname, journalist.nachname,
medium.firmenname
FROM    journalist, medium
WHERE   journalist.medium_id = medium.medium_id
```

Der Join - Beispiel mit alias

Join mit 2 Tabellen - Beispiel - mit alias

oder kürzer durch Umbenennung der Tabellen in FROM Zeile:

```
SELECT  j.vorname, j.nachname,  
        m.firmenname  
FROM    journalist j, medium m  
WHERE   j.medium_id = m.medium_id
```

UNION ALL

Syntax

```
SELECT att1,att2,att3 FROM tab1 [WHERE ...]  
UNION ALL  
SELECT att1,att2,att3 FROM tab2 [WHERE ...]
```

IN - NOT IN

Syntax

```
SELECT att1,att2,att3 FROM tabl  
WHERE att1 [NOT] IN  
(wert1{,wert2})
```

IN - NOT IN

Beispiel

```
SELECT persnr,nachname FROM person  
WHERE persnr IN  
(1,2,3)
```


IN - NOT IN

Beispiel 2

```
SELECT persnr,nachname FROM person  
WHERE persnr NOT IN  
(SELECT persnr FROM othertable WHERE ...)
```

Übung 2 LB-DB 28.2.2015

Anforderungen 1

Presse Datenbank

Sie sind in der IT Abteilung eines Messeveranstalters und mit dem Entwurf einer Presse Datenbank beauftragt. Ihre Firma veranstaltet verschiedene Messen, zu der sich die Journalisten registrieren müssen.

Journalisten können sich für mehr als eine Messe registrieren. Journalisten arbeiten für Medien oder sind als freie Journalisten tätig. Es genügt, wenn Ihre Datenbank den Sachverhalt darstellen kann:

"Ein Journalist arbeitet für EIN Medium oder ist freier Journalist"

Übung 2 LB-DB 28.2.2015

Anforderungen 2

Auch wenn kein Journalist namentlich bekannt ist, soll ein Medium mit Informationen versorgt werden können.

Mit Ihrem System soll es möglich sein, alle Pressekontakte zu einer bestimmten Messe zu informieren.

Berücksichtigen Sie dabei mindestens 3 Informationskanäle (email, Post und Telefon) und vergessen Sie nicht die Zustimmungserklärung nach dem Telekommunikationsgesetz §107 zu protokollieren.
(erteilt/nicht erteilt mit Zeitpunkt ist ausreichend)

Übung 2 LB-DB 28.2.2015

Erstellen der Tabellen und erstellen der Testdaten

- Entwerfen Sie ein entsprechendes ERM Modell, legen Sie die Primärschlüssel und Fremdschlüssel fest. Unterstreichen Sie alle Primärschlüssel.
- Erzeugen Sie die Tabellen.
- Legen Sie mindestens zwei Messen an
- Legen Sie mindestens drei Medien an
- Legen Sie mindestens vier Journalisten an, die in den angelegten Medien arbeiten
- Legen Sie mindestens einen freien Journalisten an

Übung 2 LB-DB 28.2.2015

Registrierungen eintragen

- Tragen Sie in Ihre Datenbank ein:
- Alle Journalisten registrieren sich für mindestens eine der angelegten Messen
- Legen Sie jetzt noch mindestens einen Journalisten an, der sich für keine Messe registriert.
- Legen Sie noch ein Medium an und registrieren Sie es für irgendeine Messe.

Übung 2 LB-DB 28.2.2015

Erstellen der Abfragen

- Erzeugen Sie eine Liste aller Kontakte, die einer beliebigen Messe zugordnet wurden.
Diese Liste besteht aus freien Journalisten, Journalisten die in Medien beschäftigt sind und Medien ohne namentlich registrierte Journalisten.
- Legen Sie diese Abfrage als view in der Datenbank ab.
- Erzeugen Sie eine Liste der Journalisten, die keiner Messe zugeordnet sind.

Übung 2 LB-DB 28.2.2015

...für die Profis

- Erzeugen sie Indizes auf alle Schlüssel und Fremdschlüssel (UNIQUE?)
- Skizzieren Sie eine einfache Zugriffsteuerung Benutzer, read only, read write
- Protokollieren Sie jede Änderung eines Journalisten in einer Tabelle log
wer hat wann geändert, wie sah der Kontakt vorher aus.

links

- 1 <http://de.wikipedia.org/wiki/Entity-Relationship-Modell>
- 2 http://de.wikipedia.org/wiki/Relationale_Datenbank
- 3 <http://de.wikipedia.org/wiki/SQL>
- 4 <http://www.sql-und-xml.de/xml/sql-tutorial/index>
- 5 http://de.wikipedia.org/wiki/Liste_der_Datenbankmanagementsysteme