

Datenbanksysteme

6 Integrität relationaler Datenbanken

Prof. Dr. Gregor Grambow

Hochschule Aalen
Fakultät Elektronik und Informatik

Überblick

Inhalt

- Integritätsbedingungen
 - Schlüssel

Ziele

- Integritätswahrung in relationalen Datenbanken verstehen

Referentielle Integrität

- Fremdschlüssel
 - verweisen auf Tupel einer Relation
 - z.B. gelesenVon in Vorlesungen verweist auf Tupel in Professoren
- referentielle Integrität
 - Fremdschlüssel müssen auf existierende Tupel verweisen oder einen Nullwert enthalten

Inklusionsabhängigkeiten → Fremdschlüssel

- Eine Attributmenge $F \subseteq A$ wird als *Fremdschlüssel* in $R = (A, \dots)$ bezeichnet, falls
- die Inklusionsabhängigkeit $(R|_F \subseteq R'|_{F'})$ gilt (wobei $R' \neq R$) und
- F' Schlüssel im *fremden* Schema R' ist
- $(R|_F \subseteq R'|_{F'})$ wird dann als *Fremdschlüsselabhängigkeit* bezeichnet
- Mitarbeiter = {P-Nr, N-Name, V-Name, G-Datum, Gehalt}
- Anstellung = {P-Nr, Abt-Nr}

Referentielle Integrität in SQL

- Kandidatenschlüssel: **unique**
- Primärschlüssel: **primary key**
- Fremdschlüssel: **foreign key**

- **create table** Orders (
 OrderID **integer primary key**,
 OrderNumber **integer not null**,
 PersonID **integer references** Persons(PersonID)
);

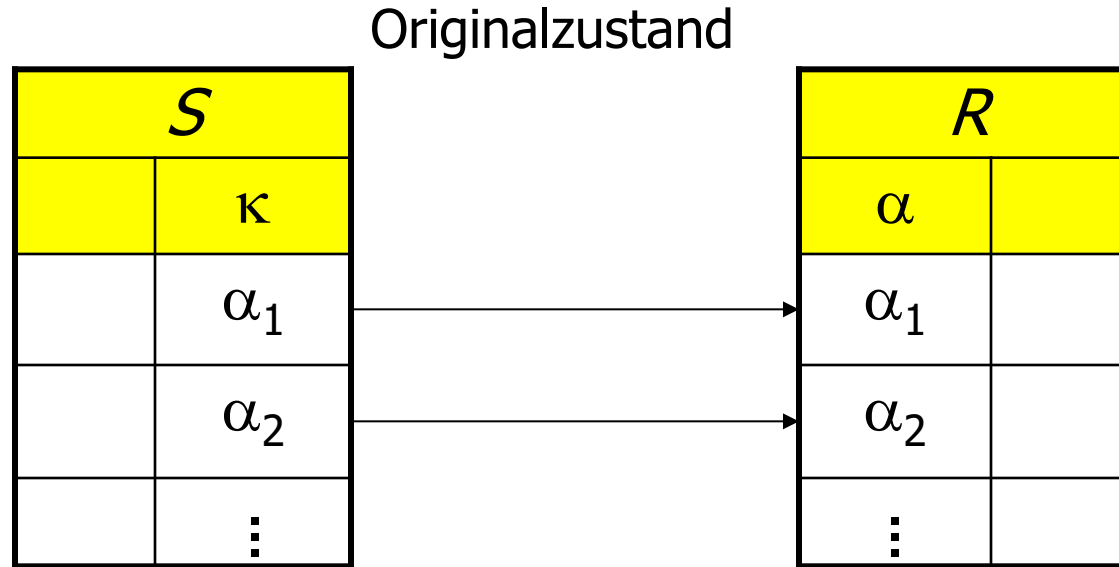
- **create table** Orders (
 OrderID **integer primary key**,
 OrderNumber **integer not null**,
 PersonID **integer**,
 foreign key(PersonID) **references** Persons(PersonID)
);

Unterschiedlich in
verschiedenen
DBMS

Einhaltung referentieller Integrität

- Änderung von referenzierten Daten
 - Default: Zurückweisen der Änderungsoperation
 - Propagieren der Änderungen: cascade
 - Verweise auf Nullwert setzen: set null

Einhaltung referentieller Integrität



Änderungsoperationen

update R

set $\alpha = \alpha'_1$

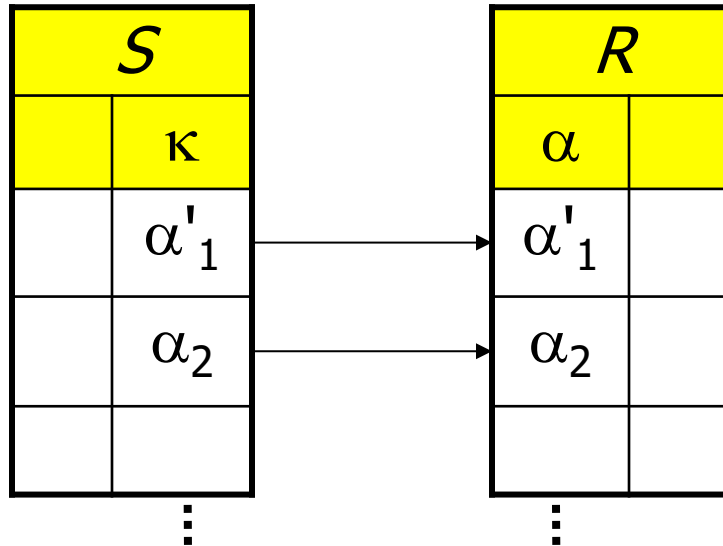
where $\alpha = \alpha_1;$

delete from R

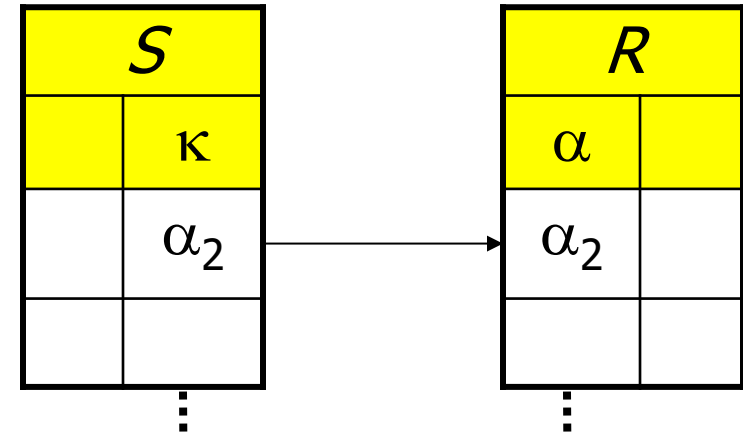
where $\alpha = \alpha_1;$

Einhaltung referentieller Integrität

Kaskadieren



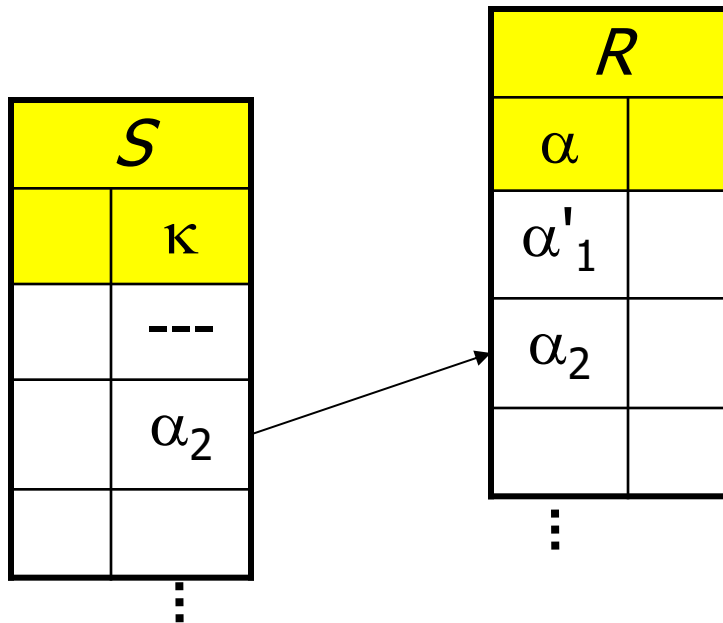
```
create table S
( ...,
  K integer references R
  on update cascade );
```



```
create table S
( ...,
  K integer references R
  on delete cascade );
```


Einhaltung referentieller Integrität

Auf Null setzen

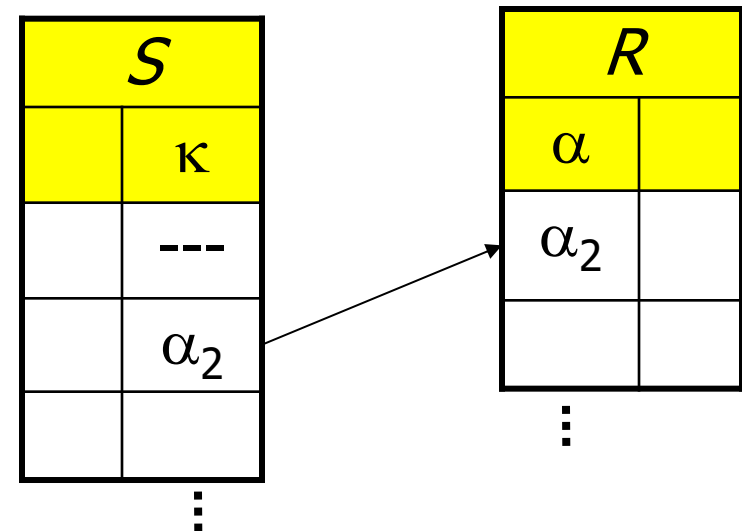


`create table S`

`(...,`

`K integer references R`

`on update set null);`



`create table S`

`(...,`

`K integer references R`

`on delete set null);`

Einfache statische Integritätsbedingungen

- Wertebereichseinschränkungen
 - ... check Semester between 1 and 13
- Aufzählungstypen
 - ... check Rang in ('C2', 'C3', 'C4') ...

Das Universitätsschema mit Integritätsbedingungen

- **create table** Studenten

```
( MatrNr      integer primary key,  
  Name        varchar(30) not null,  
  Semester    integer check Semester between 1 and 13);
```

- **create table** Professoren

```
( PersNr      integer primary key,  
  Name        varchar(30) not null,  
  Rang        character(2) check (Rang in  
    (`C2`, `C3`, `C4`)),  
  Raum        integer unique );
```

Das Universitätsschema mit Integritätsbedingungen

- **create table** Assistenten
 (PersNr **integer primary key**,
 Name **varchar(30) not null**,
 Fachgebiet **varchar(30)** ,
 Boss **integer**,
 foreign key (Boss) **references** Professoren(PersNr)
 on delete set null);
- **create table** Vorlesungen
 (VorlNr **integer primary key**,
 Titel **varchar(30)** ,
 SWS **integer**,
 gelesenVon **integer references** Professoren(PersNr)
 on delete set null);

Das Universitätsschema mit Integritätsbedingungen

- **create table** hören

```
( MatrNr          integer references Studenten on delete cascade,  
  VorlNr          integer references Vorlesungen on delete cascade,  
  primary key     (MatrNr, VorlNr));
```

- **create table** voraussetzen

```
( Vorgänger       integer references Vorlesungen(VorlNr)  
                        on delete cascade,  
  Nachfolger       integer references Vorlesungen(VorlNr)  
                        on delete cascade,  
  primary key      (Vorgänger, Nachfolger));
```

- **create table** prüfen

```
( MatrNr integer references Studenten on delete cascade,  
  VorlNr          integer references Vorlesungen,  
  PersNr          integer references Professoren on delete set null,  
  Note            numeric (2,1)  check (Note between 0.7 and 5.0),  
  primary key     (MatrNr, VorlNr));
```

Zusammenfassung

- Integritätsbedingungen
 - Schlüssel