

6 Die relationale Anfragesprache SQL

6.1 Der SFW-Block im Detail

6.2 Erweiterungen des SFW-Blocks

6.3 Rekursion

Teil VI

Die relationale Anfragesprache SQL

Die relationale Anfragesprache SQL

- 1 Der SFW-Block in Detail
- 2 Erweiterungen des SFW-Blocks
- 3 Rekursion

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-1

Die relationale Anfragesprache SQL Der SFW-Block in Detail

Struktur einer SQL-Anfrage

```
-- Anfrage
select projektionsliste
from relationenliste
[ where bedingung ]
```

select

- Projektionsliste
- arithmetische Operationen und Aggregatfunktionen

from

- zu verwendende Relationen, evtl. Umbenennungen

where

- Selektions-, Verbundbedingungen
- Geschachtelte Anfragen (wieder ein SFW-Block)

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-2

Die relationale Anfragesprache SQL Der SFW-Block in Detail

Die select-Klausel

- Spezialfall der Projektionsliste: *
 - ▶ liefert alle Attribute der Relation(en) aus dem from-Teil

```
select *  
from WEINE
```

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-12

Die relationale Anfragesprache SQL Der SFW-Block in Detail

distinct eliminiert Duplikate

```
select Name from WEINE
```

- liefert die Ergebnisrelation als Multimenge:

Name
La Rose Grand Cru
Creek Shiraz
Zinfandel
Pinot Noir
Pinot Noir
Riesling Reserve
Chardonnay

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-13

Die relationale Anfragesprache SQL Der SFW-Block in Detail

distinct eliminiert Duplikate /2

```
select distinct Name from WEINE
```

- ergibt Projektion aus der Relationenalgebra:

Name
La Rose Grand Cru
Creek Shiraz
Zinfandel
Pinot Noir
Riesling Reserve
Chardonnay

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-14

Die relationale Anfragesprache SQL Der SFW-Block in Detail

Tupelvariablen und Relationennamen

- Anfrage


```
select Name from WEINE
```
- ist äquivalent zu


```
select WEINE.Name from WEINE
```
- und


```
select W.Name from WEINE W
```

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-15

Die relationale Anfragesprache SQL Der SFW-Block in Detail

Präfixe für Eindeutigkeit

```
select Name, Jahrgang, Weingut -- (falsch!)
from WEINE natural join ERZEUGER
```

- Attribut weingut existiert sowohl in der Tabelle WEINE als auch in ERZEUGER!
- richtig mit Präfix:


```
select Name, Jahrgang, ERZEUGER.Weingut
from WEINE natural join ERZEUGER
```

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-16

Die relationale Anfragesprache SQL Der SFW-Block in Detail

Tupelvariablen für Eindeutigkeit

- bei der Verwendung von Tupelvariablen, kann der Name einer Tupelvariablen zur Qualifizierung eines Attributs benutzt werden:


```
select w1.Name, w2.Weingut
from WEINE w1, WEINE w2
```

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-17

Weiteres zu SQL

- Erweiterungen des SFW-Blocks
 - ▶ innerhalb der **from**-Klausel weitere Verbundoperationen (äußerer Verbund),
 - ▶ innerhalb der **where**-Klausel weitere Arten von Bedingungen und Bedingungen mit Quantoren,
 - ▶ innerhalb der **select**-Klausel die Anwendung von skalaren Operationen und Aggregatfunktionen,
 - ▶ zusätzliche Klauseln **group by** und **having**
- rekursive Anfragen

Skalare Ausdrücke

- Umbenennung von Spalten: *ausdruck as neuer-name*
- skalare Operationen auf
 - ▶ numerischen Wertebereichen: etwa +, -, * und /,
 - ▶ Strings: Operationen wie **char_length** (aktuelle Länge eines Strings), die Konkatenation || und die Operation **substring** (Suchen einer Teilzeichenkette an bestimmten Positionen des Strings),
 - ▶ Datumsstypen und Zeitintervallen: Operationen wie **current_date** (aktuelles Datum), **current_time** (aktuelle Zeit), +, - und *
- bedingte Ausdrücke
- Typkonvertierung
- **Hinweise:**
 - ▶ skalare Ausdrücke können mehrere Attribute umfassen
 - ▶ Anwendung ist tupelweise: pro Eingabetupel entsteht ein Ergebnistupel

Skalare Ausdrücke /2

- Ausgabe der Namen aller Grand Cru-Weine

```
select substring(Name from 1 for
(char_length(Name) - position('Grand Cru' in Name)))
from WEINE where Name like '%Grand Cru'
```

- Annahme: zusätzliches Attribut **HerstDatum** in WEINE

```
alter table WEINE add column HerstDatum date
```

```
update WEINE set HerstDatum = date '2004-08-13'
where Name = 'Zinfandel'
```

- Anfrage:

```
select Name, year(current_date - HerstDatum) as Alter
from WEINE
```


Aggregatfunktionen in **where**-Klausel

- Aggregatfunktionen liefern nur einen Wert \rightsquigarrow Einsatz in Konstanten-Selektionen der **where**-Klausel möglich
- alle Weingüter, die nur einen Wein liefern:

```
select * from ERZEUGER e
where 1 = (
  select count(*) from WEINE w
  where w.Weingut = e.Weingut)
```

group by und having

- Notation

```
select ...
from ...
[where ...]
[group by attributliste ]
[having bedingung ]
```

Gruppierung: Schema

- Relation REL:

A	B	C	D
1	2	3	4
1	2	4	5
2	3	3	4
3	3	4	5
3	3	6	7
...			

- Anfrage:

```
select A, sum(D) from REL where ...
group by A, B
having A<4 and sum(D)<10 and max(C)=4
```

Die relationale Anfragesprache SQL Erweiterungen des SFW-Blocks

Attribute für Aggregation bzw. **having**

- zulässige Attribute hinter **select** bei Gruppierung auf Relation mit Schema R
 - ▶ Gruppierungsattribute G
 - ▶ Aggregationen auf Nicht-Gruppierungsattributen $R - G$
- zulässige Attribute für **having**
 - ▶ dito

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-60

Die relationale Anfragesprache SQL Erweiterungen des SFW-Blocks

Äußere Verbunde

- zusätzlich zu klassischen Verbund (**inner join**): in SQL-92 auch äußerer Verbund \rightsquigarrow Übernahme von „dangling tuples“ in das Ergebnis und Auffüllen mit Nullwerten
- **outer join** übernimmt alle Tupel beider Operanden (Langfassung: **full outer join**)
- **left outer join** bzw. **right outer join** übernimmt alle Tupel des linken bzw. des rechten Operanden
- äußerer natürlicher Verbund jeweils mit Schlüsselwort **natural**, also z.B. **natural left outer join**

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-61

Die relationale Anfragesprache SQL Erweiterungen des SFW-Blocks

Äußere Verbunde /2

LINKS

A	B
1	2
2	3

RECHTS

B	C
3	4
4	5

NATURAL JOIN

A	B	C
2	3	4

OUTER

A	B	C
1	2	⊥
2	3	4
⊥	4	5

LEFT

A	B	C
1	2	⊥
2	3	4

RIGHT

A	B	C
2	3	4
⊥	4	5

Sattler / Saake Datenbanksysteme Letzte Änderung: 11.10.2011 6-62

Behandlung von Nullwerten

- skalare Ausdrücke: Ergebnis **null**, sobald Nullwert in die Berechnung eingeht
- in allen Aggregatfunktionen bis auf **count(*)** werden Nullwerte vor Anwendung der Funktion entfernt
- fast alle Vergleiche mit Nullwert ergeben Wahrheitswert **unknown** (statt **true** oder **false**)
- Ausnahme: **is null** ergibt **true**, **is not null** ergibt **false**
- Boolesche Ausdrücke basieren dann auf dreiwertiger Logik

Behandlung von Nullwerten /2

and	true	unknown	false
true	true	unknown	false
unknown	unknown	unknown	false
false	false	false	false

or	true	unknown	false
true	true	true	true
unknown	true	unknown	unknown
false	true	unknown	false

not	
true	false
unknown	unknown
false	true

Selektionen nach Nullwerten

- *Null-Selektion* wählt Tupel aus, die bei einem bestimmten Attribut Nullwerte enthalten
- Notation

attribut is null

- Beispiel

```
select * from ERZEUGER
where Anbaugesicht is null
```


