



## Übung zur Vorlesung *Einsatz und Realisierung von Datenbanksystemen* im SoSe19

Maximilian {Bandle, Schüle} (i3erdb@in.tum.de)  
<http://db.in.tum.de/teaching/ss19/impldb/>

### Blatt Nr. 05

#### Hausaufgabe 1

Gehen Sie von folgender kombinierter Fragmentierung der in Abbildung 1 dargestellten Relation *Professoren* aus:

Professoren						
PersNr	Name	Rang	Raum	Fakultät	Gehalt	Steuerklasse
2125	Sokrates	C4	226	Philosophie	85000	1
2126	Russel	C4	232	Philosophie	80000	3
2127	Kopernikus	C3	310	Physik	65000	5
2133	Popper	C3	52	Philosophie	68000	1
2134	Augustinus	C3	309	Theologie	55000	5
2136	Curie	C4	36	Physik	95000	3
2137	Kant	C4	7	Philosophie	98000	1

Abbildung 1: Beispielausprägung der um drei Attribute erweiterten Relation *Professoren*

1. Zuerst erfolgt eine vertikale Fragmentierung in

$$\begin{aligned}\text{ProfVerw} &:= \Pi_{\text{PersNr, Name, Gehalt, Steuerklasse}}(\text{Professoren}) \\ \text{Profs} &:= \Pi_{\text{PersNr, Name, Rang, Raum, Fakultät}}(\text{Professoren})\end{aligned}$$

2. Das Fragment *Profs* wird weiter horizontal fragmentiert in

$$\begin{aligned}\text{TheolProfs} &:= \sigma_{\text{Fakultät} = \text{'Theologie'}}(\text{Profs}) \\ \text{PhysikProfs} &:= \sigma_{\text{Fakultät} = \text{'Physik'}}(\text{Profs}) \\ \text{PhiloProfs} &:= \sigma_{\text{Fakultät} = \text{'Philosophie'}}(\text{Profs})\end{aligned}$$

Übersetzen Sie aufbauend auf dieser Fragmentierung die folgende SQL-Anfrage in die kanonische Form.

```
select Name, Gehalt Rang
from Professoren
where Gehalt > 80000;
```

Optimieren Sie diesen kanonischen Auswertungsplan durch Anwendung algebraischer Transformationsregeln (Äquivalenzen).

## Hausaufgabe 2

Für die Rekonstruierbarkeit der Originalrelation  $R$  aus vertikalen Fragmenten  $R_1, \dots, R_n$  reicht es eigentlich, wenn Fragmente paarweise einen Schlüsselkandidaten enthalten. Illustrieren Sie, warum es also nicht notwendig ist, dass der Durchschnitt aller Fragmentschemata einen Schlüsselkandidaten enthält. Es muss also nicht unbedingt gelten

$$R_1 \cap \dots \cap R_n \supseteq \kappa,$$

wobei  $\kappa$  ein Schlüsselkandidat aus  $R$  ist.

Geben Sie ein anschauliches Beispiel hierfür – am besten bezogen auf unsere Beispiel-Relation *Professoren*.

## Hausaufgabe 3

Zeigen Sie, dass die *write-all/read-any* Methode zur Synchronisation replizierter Daten einen Spezialfall der *Quorum-Consensus*-Methode darstellt.

- Für welche Art von Workloads eignet sich dieses Verfahren besonders gut?
- Wie werden Stimmen zugeordnet um *write-all/read-any* zu simulieren?
- Wie müssen die Quoren  $Q_w$  und  $Q_r$  vergeben werden?

## Hausaufgabe 4

Um Ausfallsicherheit zu garantieren ist ein Datenwert 'A' auf vier Rechnern verteilt. Jeder Rechner hält dabei eine vollständige Kopie von 'A'. Um Konsistenz zu garantieren wird das Quorum-Consensus-Verfahren eingesetzt. Dabei ist jedem Rechner ein Gewicht  $w_i(A)$  wie folgt zugewiesen:

Rechner	Kopie	Gewicht
$R_1$	$A_1$	3
$R_2$	$A_2$	1
$R_3$	$A_3$	2
$R_4$	$A_4$	2

Das Lesequorum ist  $Q_r(A) = 4$  und das Schreibquorum is  $Q_w(A) = 5$ .

- Geben Sie **alle** Lesemöglichkeiten für eine Transaktion auf dem Datum 'A' nach dem Quorum-Consensus-Protokoll an.
- Geben Sie **alle** Schreibmöglichkeiten für eine Transaktion auf dem Datum 'A' nach dem Quorum-Consensus-Protokoll an.
- Zeigen Sie für dieses Beispiel, dass während eine Transaktion  $T_1$  ein Schreibquorum auf  $A$  hält es für andere Transaktionen  $T_x$  nicht möglich ist ein Lesequorum für  $A$  zu bekommen.

## Hausaufgabe 5

Überlegen Sie sich, welche Tupel bei der Anwendung des bloomfilterbasierten Joins in Abbildung 2 übertragen werden. Markieren Sie insbesondere, welche Tupel übertragen werden, obwohl sie keinen Joinpartner finden (sog. *false drops*). Wie kann die Anzahl dieser *false drops* verringert werden? Welche Eigenschaften sollte die Hashfunktion  $h(c)$  die bei dieser Joinbearbeitung verwendet wird erfüllen?

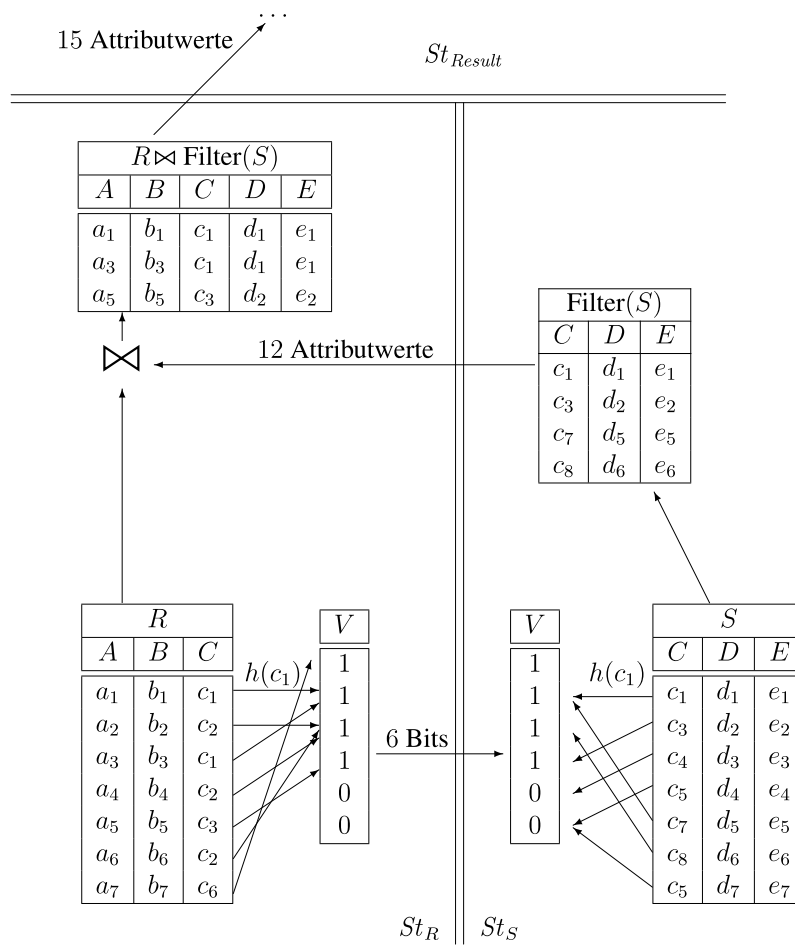


Abbildung 2: Beispiel einer verteilten Joinbearbeitung mit Bloomfilter.

## Hausaufgabe 6

Zeigen Sie, dass die Suche in einem Chord-Overlaynetzwerk durch die Nutzung der FingerTabellen in maximal logarithmisch vielen Schritten zur Größe des Zahlenrings (bzw. der Anzahl der Stationen) durchgeführt werden kann. Verwenden Sie die Suche nach K57 beginnend an Station P11 (siehe Abbildung 3) zur Illustration.

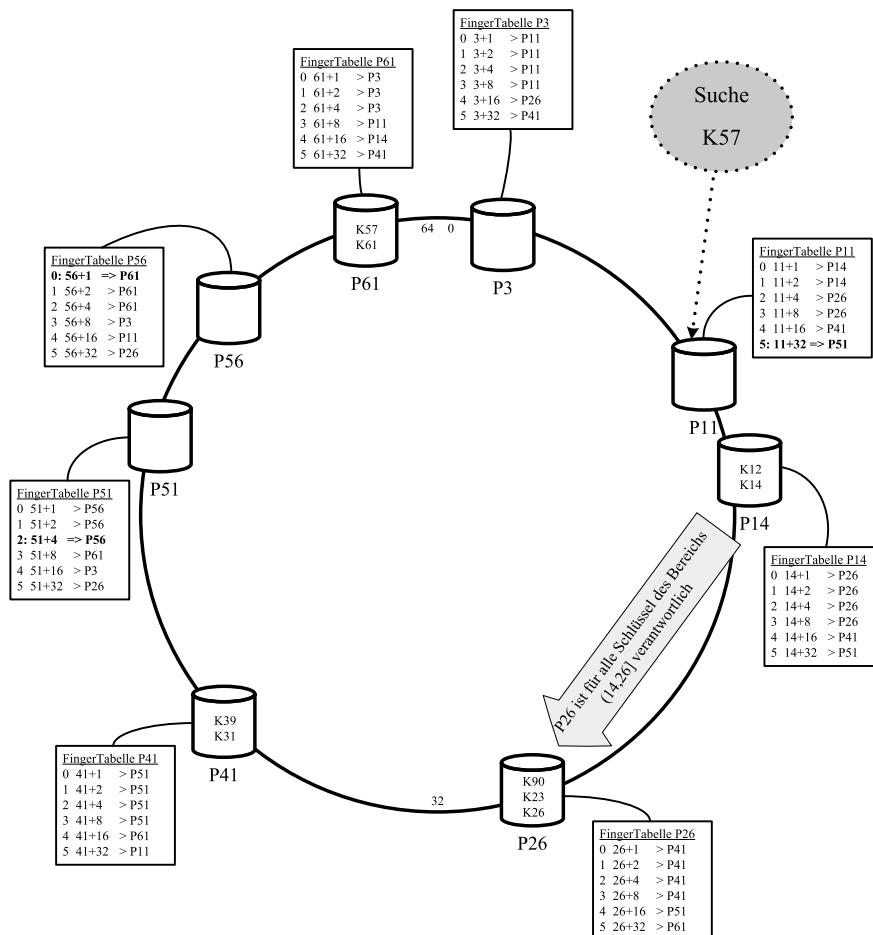


Abbildung 3: Beispiel eines Chord-Overlaynetzwerks.

## Hausaufgabe 7

Skizzieren Sie die Vorgehensweise beim Hinzufügen eines neuen Peers im Chord Netzwerk. Als Beispiel nehmen Sie die Hinzunahme eines Peers P33 in dem Beispiel-Netzwerk aus Abbildung 3.