



**Übung zur Vorlesung**  
***Einsatz und Realisierung von Datenbanksystemen im SoSe16***

Moritz Kaufmann (moritz.kaufmann@tum.de)  
<http://db.in.tum.de/teaching/ss16/impldb/>

**Blatt Nr. 09**

**Hausaufgabe 1**

Schätzen sie die Anzahl der Cache Misses die entstehen, wenn man 1000 32-bit Integer Werte (0-1000) in aufeinanderfolgender Reihenfolge in einen ART Baum einfügt. Wäre ein B+ Baum besser oder schlechter? Bei den Baumknoten müssen die Header nicht berücksichtigt werden, Pointer habe eine Größe von 64 bit.

**Hausaufgabe 2** Geben die folgenden Anfragen,

Anfrage 1: select Note from Noten where MatrNr=12345

Anfrage 2: select count(\*) from Noten where Note<1.5

Anfrage 3: insert into Noten(MatrnNr,Note) values (54321, 3.0)

Anfrage 4: update Noten set Note=1.4 where MatrNr=32154

Anfrage 5: insert into Noten(MatrnNr,Note) values (54321, 1.3)

Anfrage 6: update Noten set Note=1.6 where MatrNr=12345

analysieren sie ob die folgenden Historien unter dem MVCC Modell wie in der Vorlesung vorgestellt auftreten können. Transaktion  $X$  entspricht dabei Anfrage  $X$ , also z.B. Tx 1 ist das Ergebnis von Anfrage 1. Jede Historie steht für sich selbst und starten jeweils von einem ursprünglichen Datenzustand. Die Buchstaben innerhalb der Klammer entsprechen dabei jeweils den Tupeln auf die Zugegriffen wird. Wenn in Anfrage 2 z.B. drei Werte das Prädikat  $Note < 1.5$  erfüllen, gäbe es entsprechend drei  $r(\dots)$  Einträge auf die jeweiligen Tupel.

Historie 1:  $bot_1, r_1(A), bot_3, w_3(B), commit_1, commit_3$

Historie 2:  $bot_2, r_2(A), bot_3, w_3(B), r_2(C), commit_2, commit_3$

Historie 3:  $bot_2, r_2(A), r_2(B), bot_4, r_4(B), w_4(B), r_2(C), commit_2, commit_4$

Historie 4:  $bot_1, r_1(B), bot_6, r_6(B), w_6(B), commit_1, commit_6$

Historie 5:  $bot_2, r_2(A), bot_4, r_4(B), w_4(B), r_2(C), commit_4, commit_2$

Historie 6:  $bot_1, r_1(B), bot_6, r_6(B), w_6(B), commit_6, commit_1$

Historie 7:  $bot_2, r_2(A), bot_5, w_5(D), commit_5, r_2(D), commit_2$

Historie 8:  $bot_2, r_2(A), bot_3, w_3(B), r_2(C), commit_3, commit_2$

Historie 9:  $bot_2, r_2(A), bot_5, w_5(B), r_2(C), commit_5, commit_2$

**Hausaufgabe 3**

Definieren Sie das Prädikat  $sg(X,Y)$  das für “same generation” steht. Zwei Personen gehören zur selben Generation, wenn Sie mindestens je ein Elternteil haben, das derselben Generation angehört.

Verwenden Sie beispielsweise die folgende Ausprägung einer ElternKind Relation. Das erste Element ist hier das Kind, das Zweite ein Elternteil.

```
parent(c,a).  
parent(d,a).  
parent(d,b).  
parent(e,b).  
parent(f,c).  
parent(g,c).  
parent(h,d).  
parent(i,d).  
parent(i,e).  
parent(f,e).  
parent(j,f).  
parent(j,h).  
parent(k,g).  
parent(k,i).
```

- a) Definieren Sie das Prädikat in Datalog.
- b) Demonstrieren Sie die naive Ausführung des Prädikats.
- c) Erläutern Sie das Vorgehen bei der seminaiven Auswertung.