

Transaktionsverwaltung SS 2007 - Übungsblatt 2

Aufgabe 1

Prüfen Sie, ob die Historien $h_1 - h_4$ in folgenden Serialisierbarkeitsklassen liegen:

- CSR
- OCSR
- COCSR

$h_1 := w_1(x) \ r_2(x) \ c_2 \ w_3(y) \ w_1(y) \ r_4(y) \ c_3 \ w_4(x) \ c_4 \ c_1$

$h_2 := w_1(y) \ w_2(x) \ c_1 \ r_3(x) \ c_3 \ w_2(y) \ r_4(y) \ w_4(x) \ c_4 \ c_2$

$h_3 := w_1(x) \ w_2(y) \ c_2 \ w_1(y) \ r_3(x) \ r_4(y) \ c_3 \ w_4(x) \ c_4 \ c_1$

$h_4 := w_1(y) \ w_2(x) \ r_3(x) \ w_2(y) \ c_1 \ r_4(y) \ c_2 \ w_4(x) \ c_3 \ c_4$

Erläutern Sie kurz, wie Sie bei der Prüfung vorgegangen sind. Angenommen, für eine Historie liege der Konfliktgraph vor. Wie sieht dann ein ganz einfaches Prüfverfahren aus?

Aufgabe 2

Gegeben sei die Historie h_5 . Erstellen Sie die Abarbeitungsreihenfolge der Transaktionen unter Verwendung folgender Sperr-Protokolle:

- 2PL
- Conservative 2PL
- Strict 2PL
- Strong 2PL

$h_5 = r_1(x) \ r_2(y) \ r_3(x) \ w_2(y) \ c_2 \ r_3(y) \ w_3(x) \ r_4(z) \ c_3 \ r_1(z) \ w_4(z) \ w_1(z) \ c_1 \ c_4$

Unterscheiden Sie außerdem bei jedem der 4 Fälle die 3 Priorisierungsregeln

- Lese-Operationen bevorzugt
- Schreib-Operationen bevorzugt
- FIFO

Aufgabe 3

Man kann bei CSR-Historien noch zwei Sonderfälle unterscheiden:

Eine Historie h ist **strikt** falls für all $t_i, t_j \in \text{trans}(h)$:

für alle $p_i(x) \in \text{op}(t_i)$, $p=r$ or $p=w$, falls $w_j(x) < p_i(x)$ dann $a_j < p_i(x)$ oder $c_j < p_i(x)$.

Eine Historie h ist **rigoros** falls sie strikt ist und es gilt für all $t_i, t_j \in \text{trans}(h)$:

falls $r_j(x) < w_i(x)$ dann $a_j < w_i(x)$ oder $c_j < w_i(x)$.

Erläutern Sie mit Ihren Worten die Eigenschaften dieser Klassen. Welche Variante(n) von 2PL erzeugen Strikte Historien, welche erzeugen Rigorose Historien?

Aufgabe 4

Phantome können zustande kommen, wenn bei Änderungen sog. Zwangsbedingungen (Constraints) zwischen verschiedenen Datenelementen zu beachten sind, die vom R/W-Modell nicht erfaßt werden.

Beispiel: In unserem Buchungsbeispiel aus Kapitel 3 besteht eine referenzielle Konsistenz von BOOKING.TicketNo nach TICKET.TicketNo. Führen Sie eine neue Transaktion t4 ein, die eine neue Reservierung hinzufügt. Konstruieren Sie einen verzahnten Ablauf der Transaktionen t1 und t4, bei der es zu einem Phantom kommt.

Aufgabe 5

Betrachten Sie erneut das Buchungsbeispiel aus Kapitel 3 mit den Transaktionen t1 (jetzt leicht abgeändert) und t2. Gehen Sie von folgendem verzahnten Ablauf aus.

t1 (Drucke Passagierliste für Flug LH400):

```
select count (distinct ticketNo)
from BOOKING
where flightNo = "LH400";
Print number of tickets;
```

t2 (Buche Plätze 19D, 19E und 19G um):

```
Update TICKET
set ticketNo = ticketNo + 100000
where ticketNo in
    (select ticketNo
     from BOOKING
     where date = 12-AUG-00 and flightNo = "LH500"
     and (seatCode = "19D"
          or seatCode = "19E,,
          or seatCode = "19G" ));

Update BOOKING
set date = 11-AUG-00, ticketNo = ticketNo + 100000
where date = 12-AUG-00 and flightNo = "LH500"
and (seatCode = "19D" or seatCode = "19E,, or seatCode = "19G");
commit;
```

```
select name
from TICKET
where ticketNo in
    (select ticketNo
     from BOOKING
     where flightNo = "LH400");
print passenger list;
commit;
```

Wie werden die beiden Transaktionen t1 und t2 unter Verwendung von Multi-Granularity-Locking abgearbeitet? Vergleichen Sie Relationen-Sperren und Tupel-Sperren wie folgt, und geben Sie dazu die jeweils entstehende Historie an.

- t1 und t2 verwenden Relationen-Sperren
- t1 verwendet Relationen-Sperren, t2 Tupel-Sperren
- t1 verwendet Tupel-Sperren, t2 Relationen -Sperren
- t1 und t2 verwenden Tupel-Sperren

Bei welchen Fällen kommt der obige Ablauf zustande?

Hinweis: Gehen Sie für das Setzen von Tupel-Sperren davon aus, dass sich qualifizierende (und damit zu sperrende) Tupel zuvor identifiziert wurden, beispielsweise über einen Index.

Aufgabe 6

Nehmen Sie jetzt an, dass die Identifikation der gewünschten Tupel nicht mehr „magisch“, sondern konkret über einen Index auf flightNo erfolgt. Nehmen Sie deshalb noch Index-Sperren hinzu. Auf welcher Körnigkeitsstufe sind diese einzuführen? Wie wirkt sich die Einführung auf das Beispiel von Aufgabe 5 aus?

Aufgabe 7

Betrachten Sie anhand eines Waits-For-Graphen die Abarbeitung des Schedules s1. Wie weit läuft er problemlos ab, bzw. an welchen Stellen können Verklemmungen auftreten?

$s1 = r1(x) \ r2(x) \ w1(y) \ r3(y) \ w1(x) \ r2(y) \ w2(y)$

Wie würde die eintretende Verklemmung mit den einzelnen aus der Vorlesung bekannten Verfahren vermieden?

Der Termin für die Besprechung dieses Übungsblattes
wird in der Vorlesung bekannt gegeben.