

AG Datenbanken und Informationssysteme

Wintersemester 2006 / 2007

Prof. Dr.-Ing. Dr. h. c. Theo Härder
 Fachbereich Informatik
 Technische Universität Kaiserslautern



<http://www.dvs.informatik.uni-kl.de>

11. Übungsblatt

Für die Übung am Donnerstag, 25. Januar 2007,
 von 15:30 bis 17:00 Uhr in 13/222.

Aufgabe 1: Ausschreib- und Seitenersetzungsstrategie

Demonstrieren Sie anhand eines Beispiels, dass man die Ausschreibstrategie FORCE nicht mit der Seitenersetzungsstrategie NOSTEAL kombinieren kann, wenn parallele Transaktionen gleichzeitig Änderungen an Datensätzen innerhalb einer Seite durchführen. Betrachten Sie dazu zwei Transaktionen T_1 und T_2 , die parallel ablaufen und die Datensätze $A(T_1)$ resp. $B(T_2)$ der Seite P modifizieren.

Aufgabe 2: Charakterisierung von Sicherungspunkt-Schemata

Zur Unterstützung des Entwurfs von Logging- und Recovery-Verfahren wurden verschiedene Sicherungspunkt-Schemata eingeführt:

- transaktionsorientierte Sicherungspunkte (TOC)
- transaktionskonsistente Sicherungspunkte (TCC)
- aktionskonsistente Sicherungspunkte (ACC)
- Fuzzy Checkpoints

Als Einbringstrategien wurden

- das direkte Einbringen (NON-ATOMIC) und
- das verzögerte Einbringen zu kontrollierten Zeitpunkten (ATOMIC) unterschieden.

Als Seitenersetzungsstrategien kommen

- ein Erlauben des Verdrängens schmutziger Seiten (STEAL)
- ein Verbot des Verdrängens schmutziger Seiten (NOSTEAL)

in Frage.

Die vier Möglichkeiten der Sicherungspunkterzeugung können je mit den beiden Einbringstrategien und mit den beiden Seitenersetzungsstrategien kombiniert werden, sodass für Logging- und Recovery-Strategien zunächst 16 verschiedene Ausgangssituationen entstehen, von denen einige jedoch nicht möglich sind. Der Zustand der Datenbank im Fehlerfall und die Qualität des Sicherungspunktes sind wichtige Parameter für den Entwurf von Logging- und Recovery-Verfahren.

Beschreiben Sie zunächst zur Wiederholung für jedes Sicherungspunkt-Schema die Aktivitäten beim Anlegen des Sicherungspunktes und charakterisieren Sie die jede der 16 Kombinationen nach den im folgenden genannten Kriterien. Überprüfen Sie aber zunächst die Sinnhaftigkeit der jeweiligen Kombination. Falls die Kombination nicht umsetzbar ist oder umsetzbar aber nicht sinnvoll ist, begründen Sie dies.

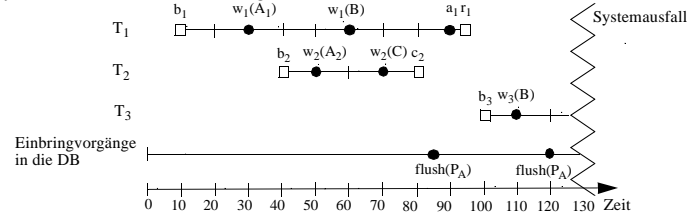
- a) Qualität der persistenten DB im Sicherungspunkt (d.h. bei dessen erfolgreichem Abschluss)
 Vorschläge für Qualitätsstufen:
- transaktionskonsistent: die DB enthält alle Änderungen erfolgreich abgeschlossener Transaktionen und keine Änderungen nicht erfolgreich abgeschlossener Transaktionen
 - aktionskonsistent: die DB enthält die Änderungen jeder Aktion (Schreiboperationen eines Transaktionsprogramms) vollständig oder gar nicht. Diese Aktionen können bereits festgeschrieben (erfolgreiches Commit der TA) oder schmutzig sein.
 - chaotisch: die DB enthält schmutzige wie festgeschriebene Änderungen. Auswirkungen einzelner Aktionen sind möglicherweise nur teilweise in der Datenbank repräsentiert. So kann beispielsweise eine Aktion, die zum Einfügen eines Satzes in eine Seite geführt hat, bereits in einer Indexseite in der DB repräsentiert sein, aber noch nicht die Änderung in der Datenseite.
- Wenn diese Qualitätsstufen nicht ausreichen, um die Qualität eines SP zu charakterisieren, führen Sie ggf. weitere Stufen ein.
- b) den Zustand der persistenten DB nach einem Systemfehler:
 Welche Qualität hat die Datenbank bei einem Systemfehler. Verwenden Sie auch hier die Qualitätsstufen von a). Wenn sich die Qualität je nach Zeitpunkt (direkt vor, direkt nach, während dem Anlegen eines Sicherungspunktes) des Systemfehlers unterscheidet, beschreiben Sie den Zustand für die ungünstigste Konstellation, da diese die Situation ist, von der bei der Recovery auszugehen ist.
- c) den Zustand des Sicherungspunktes in einem SP-Intervall:
 Wenn der Sicherungspunkt die in a) erreichte Qualität im Anschluss beibehält, nennt man ihn beständig oder stabil. Verändert sich die Qualität des Sicherungspunktes (auch innerhalb einer "Qualitätsstufe"), nennt man ihn transient.
- d) die Voraussetzungen bezüglich des Sperrrens (z.B. Einschränkungen des Sperrgranulats):
 Sind Satzsperrren möglich, oder erfordert das Verfahren zwingend ein größeres Sperrgranulat?
- e) die Grenzen der REDO- und UNDO-Recovery:
 Sicherungspunkte dienen dazu, beim Wiederanlauf nicht alle Änderungen seit Systemstart auf Redo bzw. Undo überprüfen zu müssen. Ab welcher Position muss das Log in der Redo-Phase gelesen werden? Bis zu welcher Log-Position erstreckt sich die Undo-Phase.

Aufgabe 3: Logging und Recovery mit Rollback und Satzsperrn

Gegeben sei ein DBMS, das folgende parallel laufende Transaktionen T_1, T_2 und T_3 mit Hilfe von Satzsperrn verwaltet. Dabei ändert T_1 die Datenelemente A_1 und B , T_2 die Datenelemente A_2 und C , und T_3 das Datenelement B . Die beiden Datenelemente A_1 und A_2 befinden sich in der Seite P_A , B in P_B und C in P_C .

T_2 wird zum Zeitpunkt 80 erfolgreich beendet (c_2), während T_1 zum Zeitpunkt 90 mit einer Abort-Operation beginnt (a_1). Die Seite P_A wird jeweils zu den Zeitpunkten 85 und 120 aus dem DB-Puffer verdrängt. Alle Rücksetzmaßnahmen von T_1 , d. h. die Ausführung ihrer Kompensationsoperationen, sind zum Zeitpunkt 95 (r_1) vor b_3 vollständig ausgeführt worden.

Während der Ausführung der Transaktionen werden keinerlei Sicherungspunkte gesetzt. Bei der Recovery wendet das DBMS vollständiges REDO an.



- a) Wie behandelt man die Rollback-Operation?
- b) Führen Sie anschließend die in der Abbildung gezeigten Aktionen der Transaktionen nacheinander durch und vervollständigen dabei Sie die folgende Tabelle:

Zeit	Aktion	Änderung im DB-Puffer (Seite, LSN)	Änderung in der DB (Seite, LSN)	Log-Eintrag im Log-Puffer (LSN, TAID, Log-Info, PrevLSN, UndoNxtLSN)	Zur Log-Datei hinzugefügte Log-Einträge (LSNs)
...

- c) Führen Sie mit Hilfe der Log-Datei aus b) die in der Vorlesung vorgestellte Restart-Prozedur unter Berücksichtigung der Idempotenz der REDO-/UNDO-Operationen durch.
- d) Zeichnen Sie die in der Vorlesung benutzte schematische Zustandsdarstellung der Log-Datei nach dem erfolgreichen Restart.

Aufgabe 4: Logging und Recovery mit Sicherungspunkten

In der folgenden Tabelle sei eine Historie gegeben, in der ein Sicherungspunkt (Checkpoint cp) erzeugt wurde. Das DBMS stürzt nach der Ausführung der letzten Operation ab und setzt vollständiges Redo bei der Recovery ein.

Zeit	Aktion	Änderung im DB-Puffer (Seite, LSN)	Änderung in der DB (Seite, LSN)	Log-Eintrag im Log-Puffer (LSN, TAID, Log-Info, PrevLSN, UndoNxtLSN)	Zur Log-Datei hinzugefügte Log-Einträge (LSNs)
z1	b1				
z2	w1(A)				
z3	w1(B)				
z4	c1				
z5	flush(PA)				
z6	b2				
z7	w2(A)				
z8	w2(C)				
z9	cp				
z10	c2				
z11	b3				
z12	flush(PA)				
z13	w3(A)				
z14	w3(B)				
z15	flush(PB)				
z16	w3(C)				

Vervollständigen Sie die angegebene Tabelle und führen Sie anschließend den Restart durch, wenn jeweils folgende Sicherungspunktschemata angewendet werden:

- a) aktionskonsistente Sicherungspunkte (ACC)
- b) Fuzzy Checkpoints