

(*) Aufgabe 22 (2-Phasen-Sperrprotokolle):

a) Sehen Sie sich noch einmal alle möglichen Varianten des 2-Phasen-Sperrprotokolls (2-PL) aus der Vorlesung an und betrachten Sie die Unterschiede. Warum wird in der Praxis oft das strikte bzw. das rigorose 2-PL bevorzugt?

Weil so gewährleistet ist, daß keine Transaktion den modifizierten Wert einer anderen Transaktion lesen oder weiter ändern kann, bevor das commit durchgeführt wurde.

b) Ergänzen Sie die Schedules S₂ und S₃ aus Aufgabe 20 sowie S₁ und S₂ aus Aufgabe 21 um die Operationen

zum Sperren/Entsperren eines Datenbankobjekts. Dabei sollen Sie das klassische 2-PL anwenden. Welche der Schedules können mit dem 2-PL ausgeführt werden, welche nicht? Vergleichen Sie dieses Ergebnis mit der Serialisierbarkeit des jeweiligen Schedules.

c) Wiederholen Sie die Aufgabe b) für das strikte 2-PL.

d) Wiederholen Sie die Aufgabe b) für das rigorose 2-PL.

e) Wiederholen Sie die Aufgabe b) für Preclaiming.

	S1			S2			S3	
Sperren	T1	T2	Sperren	T1	T2	Sperren	T1	T2
T1: L(a)	R(a)		T1: L(a)	R(a)		T1: L(a)	R(a)	
	a:=a-10		T2: L(b)		R(b)		a:=a-10	
T1: U(a)	W(a)			a:=a-10		T2: L(b)		R(b)
T1: L(b)	R(b)				b:=b-20	T1: U(a)	W(a)	
	b:=b+10		T1: U(a)	W(a)				b:=b-20
T1: U(b)	W(b)		T2: U(b)		W(b)	T1: L(b)	R(b)	
T2: L(b)		R(b)	T1: L(b)	R(b)		T2: U(b)		W(b)
		b:=b-20	T2: L(c)		R(c)		b:=b+10	
T2: U(b)		W(b)		b:=b+10		T2: L(c)		R(c)
T2: L(c)		R(c)			c:=c+20	T1: U(b)	W(b)	
		c:=c+20	T1: U(b)	W(b)				c:=c+20
T2: U(c)		W(c)	T2: U(c)		W(c)	T2: U(c)		W(c)

c) Man sieht im S3 deutlich, daß sich die Locks für b überschneiden. T1 will das von T2 gesperrte b lesen, danach will T2 es nach seinen Änderungen, die T1 nicht mitbekommen hat, schreiben. T1 müsste also beim R(b) blockiert werden, bis T2 das b freigibt.

d) Beim strikten PL würde S1 normal funktionieren, bei S2 und S3 gibt es eine Blockierung beim Zugriff auf b. Erst nach der vollständigen Abarbeitung von T2 kann T1 weiterarbeiten. Die Ergebnisse bleiben die gleichen. Berücksichtigt wurde in dieser Antwort nicht der Unterschied zwischen Read- und Write-Lock.

e) Auch mit Preclaiming ändert sich der Ablauf von S1 nicht. S2 und S3 werden im gleichen Schema manipuliert: Da T1 als erstes arbeitet, wird T2 blockiert. Erst nach vollständigem Durchlauf von T1 beginnt T2. Bei nicht kommutativen Operationen kann das zu nicht beabsichtigten Endergebnissen führen.

	S1				S2		
Sperren	T1	T2	T3	Sperren	T1	T2	T3
T2: L(Z)		R(Z)		T3: L(Y)			R(Y)
T2: L(Y)		R(Y)		T3: L(Z)			R(Z)
T2: U(Y)		W(Y)		T1: L(X)	R(X)		
T3: L(Y)			R(Y)	T1: U(X)	W(X)		
T3: L(Z)			R(Z)	T3: U(Y)			W(Y)
T1: L(X)	R(X)			T3: U(Z)			W(Z)
T1: U(X)	W(X)			T2: L(Z)		R(Z)	
T3: U(Y)			W(Y)	T1: L(Y)	R(Y)		
T3: U(Z)			W(Z)	T1: U(Y)	W(Y)		
T2: L(X)		R(X)		T2: L(Y)		R(Y)	
T1: L(Y)	R(Y)			T2: U(Y)		W(Y)	
T1: U(Y)	W(Y)			T2: L(X)		R(X)	
T2: U(X)		W(X)		T2: U(X)		W(X)	

c) Die kritischen Phasen im S1 sind zwischen dem Lesezugriff auf Z durch T2 und durch T3. T2 hebt die Sperre quasi gar nicht auf. Im S2 gibt es nur eine Irritation durch den Readlock von Z durch T2, der auch hier nicht aufgehoben wird.

d) Hier könnte es in beiden Schedules Probleme geben, weil die Zugriffe verschachtelt sind. T2 sorgt in beiden Schedules für Probleme durch das Sperren und Nichtwiederfreigeben von Z. In S1 muss T3 wegen der Sperrung von Y in T2 auf das Ende von T2 warten. In S3 läuft alles glatt.

e) Gleiche Situation: In S1 nimmt sich T2 X, Y und Z und sorgt so für Verzögerungen, die erst T3 und danach T1 ablaufen lassen. In S2 werden T3, T1 und T2 nacheinander durchlaufen.

Aufgabe 23 (ORAC !):

a) Setzen Sie den Transaktionsmodus in Oracle auf und prüfen Sie, dass das System die Einhaltung dieser Vereinbarung kontrolliert.

b) Setzen Sie den Isolationslevel in Oracle auf und versuchen Sie die Probleme des Non-Repeatable Read und der Phantome praktisch zu produzieren.

c) Stellen Sie fest, wie das Datenbankmanagementsystem ORACLE mit Deadlocks umgeht.

Hinweis: Sie müssen zur Lösung des Aufgabenteils c) versuchen, einen Deadlock zu produzieren und dann beobachten, was das System unternimmt. Um einen Deadlock zu verursachen, benötigen Sie mindestens zwei parallel geöffnete SQL-Plus Sitzungen, die sich dann gegenseitig behindern.