	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Aufgabe 1: Referentielle Aktionen

[9 P.]

a) Welche Anforderung erfüllt ein (bzgl. der referentiellen Aktionen) sicheres Schema?

[1 P.]

Lösungsvorschlag:


Bei einem sicheren Schema ist das Ergebnis einer Änderungsoperation unabhängig von der Reihenfolge, in der die referentiellen Aktionen ausgeführt werden; es treten also keine reihenfolgeabhängige Ergebnisse auf.

b) Gegeben sei folgende Datendefinition:

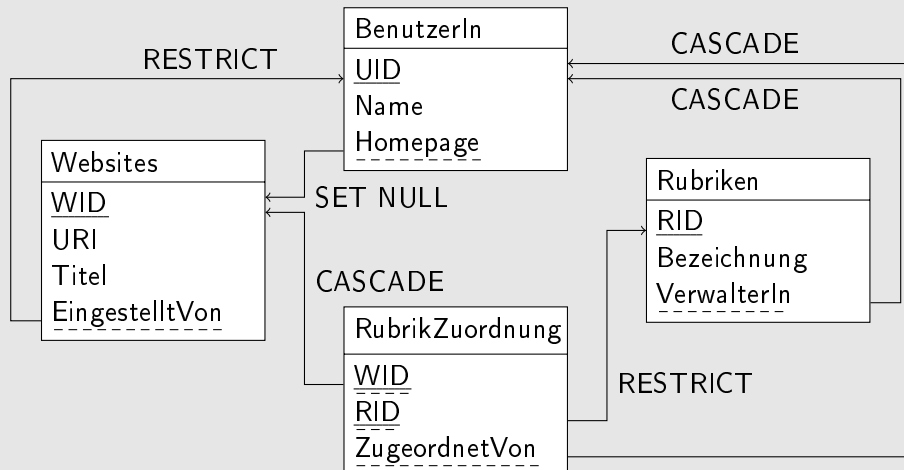
```
CREATE TABLE Websites(
    WID          INT          PRIMARY KEY,
    URI          VARCHAR(200) NOT NULL,
    Titel        VARCHAR(50)  NOT NULL,
    EingestelltVon INT        NOT NULL
);
CREATE TABLE BenutzerIn(
    UID          INT          PRIMARY KEY,
    Name         VARCHAR(20)  NOT NULL,
    Homepage     INT          REFERENCES Websites (WID) ON DELETE SET NULL
);
ALTER TABLE Websites ADD FOREIGN KEY (EingestelltVon)
    REFERENCES BenutzerIn (UID) ON DELETE RESTRICT;
CREATE TABLE Rubriken(
    RID          INT          PRIMARY KEY,
    Bezeichnung  VARCHAR(30)  NOT NULL,
    VerwalterIn  INT          NOT NULL
    REFERENCES BenutzerIn (UID) ON DELETE CASCADE
);
CREATE TABLE RubrikZuordnung(
    WID          INT          REFERENCES Websites (WID) ON DELETE CASCADE,
    RID          INT          REFERENCES Rubriken (RID) ON DELETE RESTRICT,
    ZugeordnetVon INT NOT NULL REFERENCES BenutzerIn (UID) ON DELETE CASCADE,
    PRIMARY KEY (WID, RID)
);
```

Zeichnen Sie den zugehörigen Referenzgraphen und beschriften Sie alle Kanten mit den entsprechenden referentiellen Aktionen.

[4 P.]

	Lehrveranstaltung	Grundlagen von Datenbanken		WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)		
	Gesamtpunktzahl	40		
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016

Lösungsvorschlag:



c) Erläutern Sie, in welchen Fällen das vorliegende Schema unsicher bezüglich referenzieller Aktionen ist. [2 P.]


Lösungsvorschlag:

Es können reihenfolgeabhängige Ergebnisse auftreten, wenn eine BenutzerIn gelöscht wird, die zwar keine Websites erstellt hat, die aber Rubriken verwaltet, denen nur diese BenutzerIn Websites zugeordnet hat. Die Löschoperation kann auf zwei Pfaden von Benutzer zu RubrikZuordnung kaskadieren:

- Pfad 1: BenutzerIn → Rubriken → RubrikZuordnung
- Pfad 2: BenutzerIn → RubrikZuordnung

Es gibt zwei mögliche Auswertungsreihenfolgen:

- Pfad 1 zuerst:
 - Alle von der gelöschten BenutzerIn verwalteten Rubriken-Tupel werden gelöscht.
 - Die Löschung der zugehörigen RubrikZuordnung-Tupel wird zurückgewiesen.
 - ⇒ Der Löschvorgang ist nicht erfolgreich.
- Pfad 2 zuerst:
 - Die RubrikZuordnung-Tupel der BenutzerIn werden gelöscht.
 - Alle von der gelöschten BenutzerIn verwalteten Rubriken-Tupel werden gelöscht.
 - ⇒ Der Löschvorgang ist erfolgreich.

	Lehrveranstaltung	Grundlagen von Datenbanken WS 2015/16		
	Aufgabenzettel	5 (Lösungsvorschläge)		
	Gesamtpunktzahl	40		
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016


- d) Beschreiben Sie, welche Vorkehrungen (Änderungen der referenziellen Aktionen) getroffen werden könnten, um diesen Missstand zu beheben.

[2 P.]

Lösungsvorschlag:

Das Problem kann durch veränderte Referenzdefinitionen behoben werden. Dabei gibt es mehrere Möglichkeiten. Drei davon werden hier kurz skizziert.

- Wird die Referenz RubrikZuordnung.RID → Rubriken.RID als ON DELETE NO ACTION deklariert, können keine reihenfolgeabhängigen Ergebnisse mehr auftreten. Effektiv wird zunächst die referentielle Aktion ON DELETE CASCADE für die Referenz RubrikZuordnung.ZugeordnetVon → BenutzerIn.UID ausgeführt, bevor schließlich die Referenz RubrikZuordnung.RID → Rubriken.RID überprüft wird.
- Wird die Referenz RubrikZuordnung.RID → Rubriken.RID als ON DELETE CASCADE deklariert, können keine reihenfolgeabhängigen Ergebnisse mehr auftreten. Es werden nun automatisch alle Websites aus den Rubriken entfernt, die von der zu löschenden BenutzerIn verwaltet werden.
- Wird die Referenz Rubriken.Verwalter → BenutzerIn.UID als ON DELETE RESTRICT deklariert, können keine reihenfolgeabhängigen Ergebnisse mehr auftreten. Es können nun jedoch keine BenutzerInnen mehr gelöscht werden, die noch Rubriken verwalten.

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Aufgabe 2: Änderbarkeit von Sichten

[8 P.]

a) Gegeben seien folgende Basisrelationen:

Raumschiffe(RNr, Name, Fraktion, Typ, Geschwindigkeit, Baujahr)
Besatzungsmitglieder(BNr, Name, Rang, Schiff → *Raumschiffe.RNr*)

Geben Sie die SQL-Anweisungen an, die notwendig sind, um die folgenden Sichten zu erstellen. Geben Sie zu jeder dieser Sichten an, ob sie Änderungsoperationen auf den in ihr enthaltenen Tupeln erlaubt. Begründen Sie Ihre Antwort, falls dies nicht der Fall ist.

[3 P.]

- i) *EnterpriseCrew*: BNr, Name und Rang aller Besatzungsmitglieder, die zu einem Schiff mit Namen 'Enterprise' gehören.

Lösungsvorschlag:

```
CREATE VIEW EnterpriseCrew
AS SELECT b.BNr, b.Name, b.Rang
FROM Raumschiffe r, Besatzungsmitglieder b
WHERE b.Schiff = r.RNr
AND r.Name = 'Enterprise';
```


Die Sicht ist aufgrund der Verbundoperation nicht änderbar.

- ii) *Captains*: Die Namen aller Besatzungsmitglieder mit dem Rang 'Captain'.

Lösungsvorschlag:

```
CREATE VIEW Captains
AS SELECT Name
FROM Besatzungsmitglieder
WHERE Rang = 'Captain';
```

Die Sicht ist nicht änderbar, da sie nicht den Primärschlüssel der Besatzungsmitglieder-Relation enthält.

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- iii) *WarpFed*: RNr, Fraktion und Baujahr für jedes Warp-fähige Raumschiff der Föderation.
Hinweis: Genau jedes Raumschiff mit Geschwindigkeit ≥ 1 ist Warp-fähig.

Lösungsvorschlag:

```
CREATE VIEW WarpFed
AS SELECT RNr, Fraktion, Baujahr
FROM Raumschiffe
WHERE Geschwindigkeit >= 1
AND Fraktion = 'Föderation';
```

Die Sicht ist änderbar.

- b) Auf der Raumschiffe-Basisrelation seien folgende Sichten definiert:

```
CREATE VIEW Föderationsschiffe
AS SELECT * FROM Raumschiffe
WHERE Fraktion = 'Föderation'
WITH CASCADED CHECK OPTION;

CREATE VIEW Forschungsschiffe
AS SELECT * FROM Föderationsschiffe
WHERE Typ = 'Forschungsschiff';


CREATE VIEW GalaxyKlasse
AS SELECT * FROM Forschungsschiffe
WHERE Geschwindigkeit = 9.8;

CREATE VIEW NebulaKlasse
AS SELECT * FROM Forschungsschiffe
WHERE Baujahr > 2365
WITH CASCADED CHECK OPTION;
```

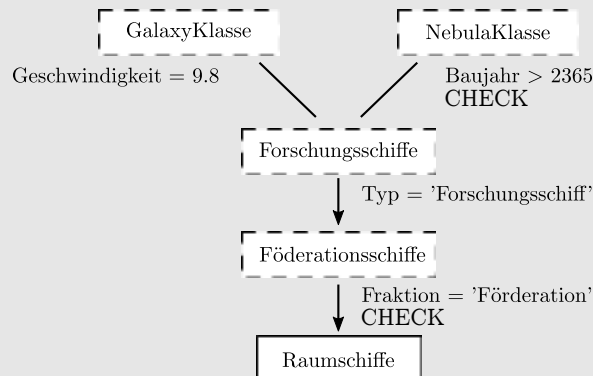
Es handelt sich bei allen obigen Sichtdefinitionen um änderbare Sichten. Bestimmen Sie, ob die folgenden SQL-Anweisungen auf diesen Sichtdefinitionen durchgeführt werden können. Für die Fälle, in denen die Änderung bzw. das Einfügen zulässig ist, geben Sie außerdem an, in welchen Sichten **auf jeden Fall alle** geänderten/eingefügten Tupel nach Abschluss der Operation sichtbar werden.

Hinweis: Im Falle von UPDATE-Operationen ist davon auszugehen, dass die zu ändernden Tupel vor der Änderung die die Sicht definierenden Prädikate erfüllen.

[5 P.]

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Lösungsvorschlag:



- i) **UPDATE** Föderationsschiffe
SET Geschwindigkeit = 9.8
WHERE Geschwindigkeit = 9.7
AND Typ = 'Kriegsschiff'
AND Baujahr = 2350;

Lösungsvorschlag:


Die Operation ist erlaubt. Die geänderten Tupel tauchen einzig und allein in der Föderationsschiffe-Sicht auf, da sie bereits nicht mehr die Bedingung Typ = 'Forschungsschiff' für Forschungsschiffe erfüllen und die anderen Sichten auf der Forschungsschiffe-Sicht aufbauen.

- ii) **INSERT INTO** GalaxyKlasse
VALUES (7, 'Sonnensegel', 'Bajoraner',
 'Forschungsschiff', 1, 1623);

Lösungsvorschlag:

Die Operation ist nicht erlaubt, da das einzufügende Tupel die Bedingung Fraktion = 'Föderation' für Föderationsschiffe verletzt.

Hinweis: In MySQL funktioniert diese Anweisung, da MySQL nur die CHECK Option der adressierten Sicht aber nicht diejenigen der darunterliegenden Sichten berücksichtigt und die Sicht GalaxyKlasse mit keiner CHECK Option versehen ist.

	Lehrveranstaltung	Grundlagen von Datenbanken WS 2015/16		
	Aufgabenzettel	5 (Lösungsvorschläge)		
	Gesamtpunktzahl	40		
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016

- iii) **UPDATE** Forschungsschiffe
SET Baujahr = 2360
WHERE Name = 'Enterprise'
AND Geschwindigkeit = 9.8

Lösungsvorschlag:

Die Änderungsoperation ist erlaubt. Die einzige Sicht, in der die geänderten Tupel nicht mehr auftauchen, wenn sie vor der Operation in ihr vorhanden waren, ist die Sicht NebulaKlasse, weil offensichtlich das Baujahr nicht der Bedingung Baujahr > 2365 entspricht. Folglich können geänderte Tupel in den Sichten Föderationsschiffe, Forschungsschiffe und GalaxyKlasse gefunden werden.

- iv) **UPDATE** NebulaKlasse
SET Baujahr = 2360
WHERE Geschwindigkeit = 9.6;


Lösungsvorschlag:

Geänderte Tupel wären nach der Operation nicht mehr in der Sicht NebulaKlasse zu finden. Da diese Sicht mit einer CHECK-Klausel definiert wurde, würde die Operation somit zurückgewiesen.

- v) **INSERT INTO** GalaxyKlasse
VALUES (88, 'Delta Flyer', 'Föderation',
'Forschungsschiff', 9.81, 2375);

Lösungsvorschlag:

Die Operation ist erlaubt, allerdings ist das einzufügende Tupel nicht in der Sicht GalaxyKlasse sichtbar, weil es die Bedingung Geschwindigkeit = 9.8 verletzt. In allen anderen Sichten taucht es auf.

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Aufgabe 3: Serialisierbarkeit und Anomalien

[15 P.]

Gegeben sind die folgenden Transaktionen $T_1 = r_1(B) \ r_1(A) \ w_1(A)$ und $T_2 = r_2(A) \ w_2(B) \ w_2(A)$.


T_1 liest den Wert von B, dann den Wert von A, erhöht den gelesenen Wert von A um 180, addiert den Wert von B dazu und schreibt den neuen Wert nach A zurück.

T_2 liest den Wert von A, schreibt den alten Wert von A nach B (ohne B vorher gelesen zu haben), erhöht den Wert von A um 110 und schreibt den neuen Wert nach A zurück. Der Anfangswert von A sei 5 und der von B sei 10. Allgemein bezeichnet $w_i(x)$ den Schreibzugriff der Transaktion i auf das Objekt x und $r_i(x)$ den Lesezugriff der Transaktion i auf x. Gegeben sind die folgenden Schedules:

$S_1 = r_1(B) \ r_1(A) \ w_1(A) \ r_2(A) \ w_2(B) \ w_2(A)$
 $S_2 = r_2(A) \ r_1(B) \ r_1(A) \ w_2(B) \ w_2(A) \ w_1(A)$
 $S_3 = r_2(A) \ w_2(B) \ r_1(B) \ w_2(A) \ r_1(A) \ w_1(A)$
 $S_4 = r_2(A) \ w_2(B) \ r_1(B) \ r_1(A) \ w_2(A) \ w_1(A)$
 $S_5 = r_2(A) \ r_1(B) \ r_1(A) \ w_1(A) \ w_2(B) \ w_2(A)$
 $S_6 = r_2(A) \ w_2(B) \ w_2(A) \ r_1(B) \ r_1(A) \ w_1(A)$

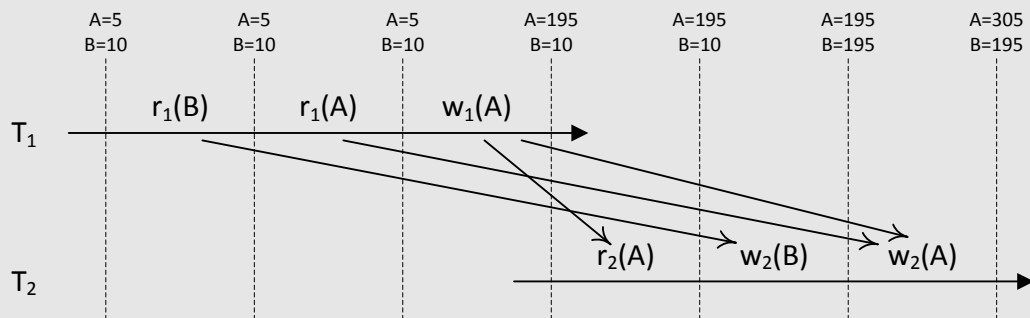
Beantworten Sie für jeden der Schedules die folgenden drei Fragestellungen:

- Wie lautet nach Ausführung des Schedules die Belegung für die Variablen A und B? [3 P.]
- Welche Abhängigkeiten existieren zwischen den Operationen der beiden Transaktionen innerhalb des Schedules? [6 P.]
- Ist der Schedule seriell, serialisierbar oder nicht serialisierbar? Erläutern sie zusätzlich bei einem nicht-serialisierbaren Schedule die auftretenden Datenanomalien. Begründen Sie die Antworten mit Hilfe der Abhängigkeiten! [6 P.]

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Lösungsvorschlag:

- Schedule $S_1 = r_1(B) \ r_1(A) \ w_1(A) \ r_2(A) \ w_2(B) \ w_2(A)$




a) Endwert für A ist 305, Endwert für B ist 195.

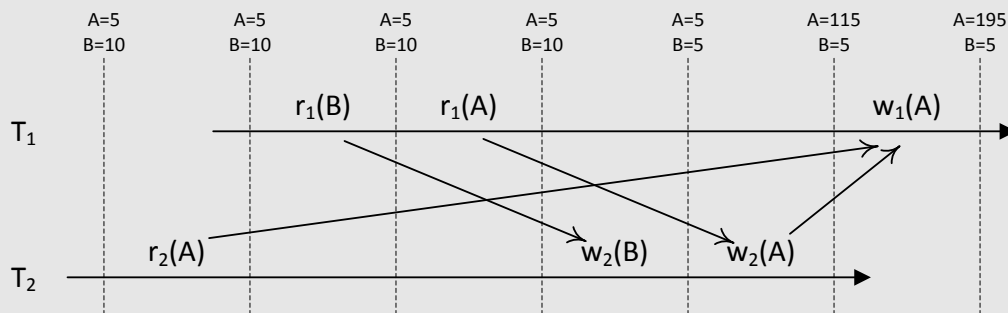
b) Abhängigkeiten

- $r_1(B) \rightarrow w_2(B)$
- $r_1(A) \rightarrow w_2(A)$
- $w_1(A) \rightarrow r_2(A)$
- $w_1(A) \rightarrow w_2(A)$

c) S_1 ist seriell.

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- Schedule $S_2 = r_2(A) \ r_1(B) \ r_1(A) \ w_2(B) \ w_2(A) \ w_1(A)$



a) Endwert für A ist 195, Endwert für B ist 5.


b) Abhängigkeiten

- $r_2(A) \rightarrow w_1(A)$
- $r_1(B) \rightarrow w_2(B)$
- $r_1(A) \rightarrow w_2(A)$
- $w_2(A) \rightarrow w_1(A)$

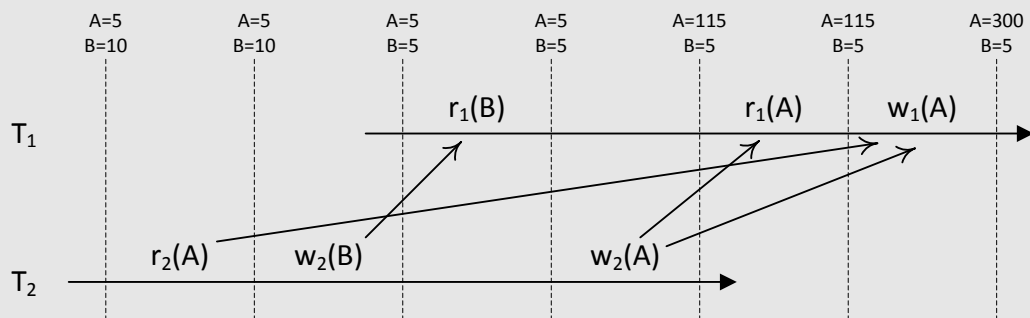
c) – Wegen der Abhängigkeiten $r_2(A) \rightarrow w_1(A)$ und $w_2(A) \rightarrow w_1(A)$ gilt: T_2 vor T_1 .

– Wegen der Abhängigkeiten $r_1(B) \rightarrow w_2(B)$ und $r_1(A) \rightarrow w_2(A)$ gilt: T_1 vor T_2 .

$\Rightarrow S_2$ ist nicht serialisierbar. Es gibt keine serielle Abfolge der beiden Transaktionen, die ein identisches Resultat für die Variablen A und B erzielt. Dabei überschreibt Transaktion T_1 alle Änderungen der Variable A, die von T_2 getätigt wurden (Lost-Update).

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- Schedule $S_3 = r_2(A) w_2(B) r_1(B) w_2(A) r_1(A) w_1(A)$




a) Endwert für A ist 300, Endwert für B ist 5.

b) Abhängigkeiten

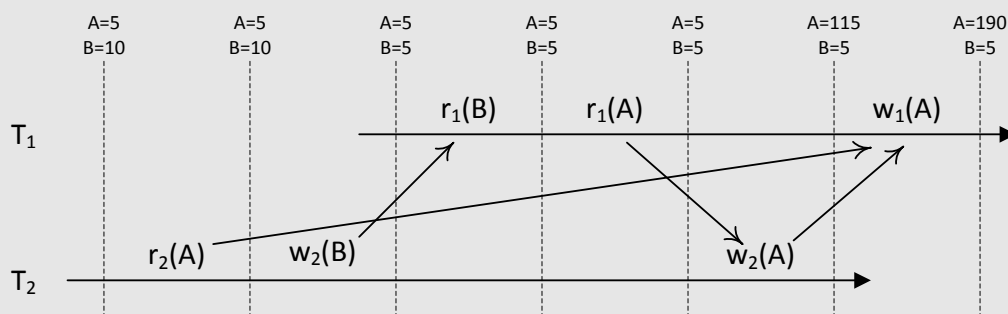
- $r_2(A) \rightarrow w_1(A)$
- $w_2(B) \rightarrow r_1(B)$
- $w_2(A) \rightarrow r_1(A)$
- $w_2(A) \rightarrow w_1(A)$

c) – Aufgrund aller Abhängigkeiten gilt: T_2 vor T_1 .

$\Rightarrow S_3$ ist serialisierbar (Schedule S_6 erzielt das identische Ergebnis für die Variablen A und B)

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- Schedule $S_4 = r_2(A) \ w_2(B) \ r_1(B) \ r_1(A) \ w_2(A) \ w_1(A)$




a) Endwert für A ist 190, Endwert für B ist 5.

b) Abhängigkeiten

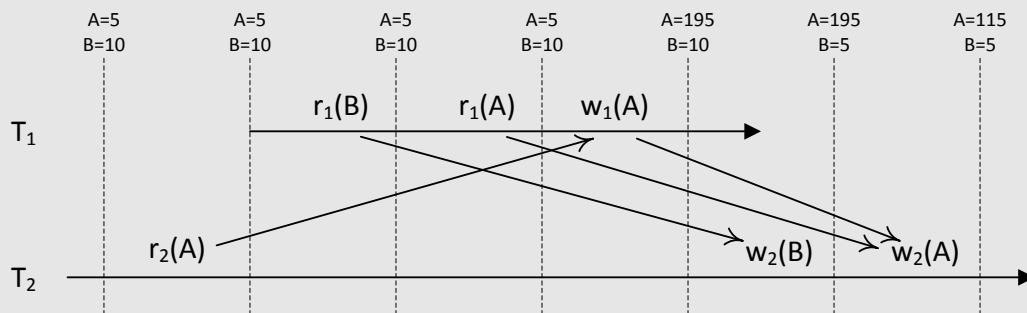
- $r_2(A) \rightarrow w_1(A)$
- $w_2(B) \rightarrow r_1(B)$
- $r_1(A) \rightarrow w_2(A)$
- $w_2(A) \rightarrow w_1(A)$

- c)
- Wegen der Abhängigkeiten $r_2(A) \rightarrow w_1(A)$, $w_2(B) \rightarrow r_1(B)$ und $w_2(A) \rightarrow w_1(A)$ gilt: T_2 vor T_1 .
 - Wegen der Abhängigkeit $r_1(A) \rightarrow w_2(A)$ gilt: T_1 vor T_2 .

$\Rightarrow S_4$ ist nicht serialisierbar. Dabei überschreibt Transaktion T_1 alle Änderungen der Variable A, die von T_2 getätigt wurden (Lost-Update).

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- Schedule $S_5 = r_2(A) \ r_1(B) \ r_1(A) \ w_1(A) \ w_2(B) \ w_2(A)$



a) Endwert für A ist 115, Endwert für B ist 5.


b) Abhängigkeiten

- $r_2(A) \rightarrow w_1(A)$
- $r_1(B) \rightarrow w_2(B)$
- $r_1(A) \rightarrow w_2(A)$
- $w_1(A) \rightarrow w_2(A)$

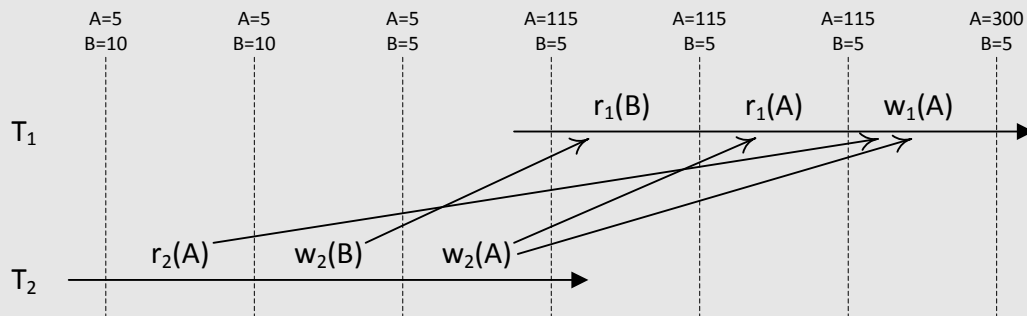
c) – Wegen der Abhängigkeit $r_2(A) \rightarrow w_1(A)$ gilt: T_2 vor T_1 .

– Wegen der Abhängigkeiten $r_1(B) \rightarrow w_2(B)$, $r_1(A) \rightarrow w_2(A)$ und $w_1(A) \rightarrow w_2(A)$ gilt: T_1 vor T_2 .

$\Rightarrow S_5$ ist nicht serialisierbar. Es gibt keine serielle Abfolge der beiden Transaktionen, die ein identisches Resultat für die Variablen A und B erzielt. Dabei überschreibt Transaktion T_2 alle Änderungen von T_1 (Lost-Update).

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

- Schedule $S_6 = r_2(A) w_2(B) w_2(A) r_1(B) r_1(A) w_1(A)$



- a) Endwert für A ist 300, Endwert für B ist 5.
- b) Abhängigkeiten
- $r_2(A) \rightarrow w_1(A)$
 - $w_2(B) \rightarrow r_1(B)$
 - $w_2(A) \rightarrow r_1(A)$
 - $w_2(A) \rightarrow w_1(A)$
- c) S_6 ist seriell.

Aufgabe 4: Transaktionen


[8 P.]

Gegeben sind die drei Objekte x, y und z, welche von den Transaktionen T_1 , T_2 und T_3 gelesen bzw. geschrieben werden. Dabei bezeichnet $w_i(x)$ den Schreibzugriff der Transaktion T_i auf das Objekt x und $r_i(x)$ den Lesezugriff der Transaktion T_i auf x. Die Operation c_i soll das Commit der Transaktion T_i darstellen.

Der Ablauf S zeigt an, in welcher Reihenfolge die Operationen der drei Transaktionen T_1 , T_2 , T_3 beim Scheduler eines Datenbanksystems eintreffen:

$$S = w_1(x) r_2(y) r_3(z) w_3(y) r_2(z) w_3(z) w_1(z) r_2(y) c_3 c_1 c_2$$

Bei der Ausführung von S soll das RX-Sperrverfahren mit 2PL zum Einsatz kommen. Vervollständigen Sie die unten angegebene Tabelle, indem Sie die Sperranforderungen und -freigaben (lock bzw. unlock) der Transaktionen, deren Lese- und Schreibzugriffe (read bzw. write) und Commits (commit) sowie die Sperrzustände der Datenobjekte (NL: no lock, R: read, X: exclusive) eintragen. Beachten Sie, dass eine Transaktion innerhalb eines Zeitschritts nur jeweils eine Operation durchführen kann. Nutzen Sie die Spalte „Bemerkungen“ für etwaige Wartebeziehungen und Benachrichtigungen an wartende Transaktionen.

	Lehrveranstaltung	Grundlagen von Datenbanken			WS 2015/16
	Aufgabenzettel	5 (Lösungsvorschläge)			
	Gesamtpunktzahl	40			
	Ausgabe	Mi. 09.12.2015	Abgabe	Fr. 08.01.2016	

Lösungsvorschlag:

$S = w_1(x) \ r_2(y) \ r_3(z) \ w_3(y) \ r_2(z) \ w_3(z) \ w_1(z) \ r_2(y) \ c_3 \ c_1 \ c_2$

Zeitschritt	T ₁	T ₂	T ₃	x	y	z	Bemerkung
0				NL	NL	NL	
1	lock(x,X)			X ₁	NL	NL	
2	write(x)	lock(y,R)		X ₁	R ₂	NL	
3		read(y)	lock(z,R)	X ₁	R ₂	R ₃	
4			read(z)	X ₁	R ₂	R ₃	
5			lock(y,X)	X ₁	R ₂	R ₃	T ₃ wartet auf Freigabe von y
6		lock(z,R)		X ₁	R ₂	R _{2,3}	
7	lock(z,X)	read(z)		X ₁	R ₂	R _{2,3}	T ₁ wartet auf Freigabe von z
8		read(y)		X ₁	R ₂	R _{2,3}	
9		unlock(y)		X ₁	X ₃	R _{2,3}	T ₃ wird benachrichtigt
10		unlock(z)	write(y)	X ₁	X ₃	R ₃	
11		commit	lock(z,X)	X ₁	X ₃	X ₃	
12			write(z)	X ₁	X ₃	X ₃	
13			unlock(y)	X ₁	NL	X ₃	
14			unlock(z)	X ₁	NL	X ₁	T ₁ wird benachrichtigt
15	write(z)		commit	X ₁	NL	X ₁	
16	unlock(x)			NL	NL	X ₁	
17	unlock(z)			NL	NL	NL	
18	commit			NL	NL	NL	