



**Hasso
Plattner
Institut**

IT Systems Engineering | Universität Potsdam

Übung Datenbanksysteme I
Besprechung

Thorsten Papenbrock



2

- Dirty Read (Write-Read Konflikt)
 - „Lesen eines falschen Wertes“
 - T1 schreibt Objekt A, das von T2 gelesen wird bevor T1 committed wurde; das Lesen ist dirty, weil T1 Objekt A erneut schreiben oder zurückgerollt werden kann.
 - Die Anomalie tritt nur ein, falls letzteres auch geschieht; andernfalls kann ein Dirty Read auch unbemerkt auftreten.
 - $w_1(A) \mathbf{r_2(A)} w_1(A)$ oder $w_1(A) \mathbf{r_2(A)} \mathbf{ABORT_1}$
 - $w_1(A) \mathbf{COMMIT_1} r_2(A)$ oder $w_1(A) \mathbf{ABORT_1} r_2(A)$

3

- Non-Repeatable Read (Read-Write Konflikt)
 - „Lesen eines veralteten Wertes“
 - T1 liest Objekt A, das von T2 geschrieben wird bevor T1 committed wurde; das Lesen ist dirty, weil T1 Objekt A erneut lesen könnte und dann einen anderen, inkonsistenten Wert erhält.
 - Die Anomalie tritt nur ein, falls letzteres auch geschieht; andernfalls kann ein Non-Repeatable Read auch unbemerkt auftreten.
 - $r_1(A) w_2(A) r_1(A)$
 - $r_1(A) COMMIT_1 w_2(A)$

- Lost Update (Write-Write Konflikt)
 - „Verlorenes Schreiben eines Wertes“
 - T1 schreibt Objekt A, das von T2 erneut geschrieben wird bevor T1 committed wurde; das Schreiben ist lost, weil T1 Objekt A erneut lesen könnte oder mit seinem Wert von A ein anderes Objekt inkonsistent ändern könnte.
 - Die Anomalie tritt nur ein, falls letzteres auch geschieht; andernfalls kann ein Lost Update auch unbemerkt auftreten.
 - $w_1(A) w_2(A) r_1(A)$ oder $w_1(A) w_2(A) w_1(B)$
 - $w_1(A) COMMIT_1 w_2(A)$

5

- Phantom Read (Read-Write und Write-Read Konflikt)
 - „Lesen inkonsistenter Werte“
 - Dirty Read + Non-Repeatable Read
 - T1 liest mehrere Objekte A und B, die während des Lesens von T2 beschrieben werden; das Lesen liefert Phantom Werte, wenn Lesen und Schreiben asynchron verläuft.
 - Die Anomalie tritt nur ein, falls letzteres auch geschieht; andernfalls kann ein Phantom Read auch unbemerkt auftreten.
 - $r_1(A) w_2(B) r_1(B) w_2(A)$
 - $w_2(A) r_1(A) w_2(B) r_1(B)$

6

- Gegeben:

- Relation: $R(A,B,C)$

- Transaktionen:

- $T_1: r_1(C) w_1(C)$

- $T_2: r_2(A) w_2(A) r_2(C) r_2(A)$

- $T_3: w_3(B) r_3(B) r_3(A)$

- (zufälliger) Schedule:

$r_2(A) w_3(B) r_3(B) w_2(A) r_3(A) r_2(C) r_1(C) w_1(C) r_2(A)$

- **Aufgabe:**

- Füge si-locks, xl-locks und unlocks 2PL-konform ein!**

7

- Gegeben:

- Relation: $R(A,B,C)$

- Transaktionen:

T_1 : $sl_1(C)$ $r_1(C)$ $xl_1(C)$ $w_1(C)$ $u_1(C)$

T_2 : $sl_2(A)$ $r_2(A)$ $xl_2(A)$ $w_2(A)$ $sl_2(C)$ $r_2(C)$ $r_2(A)$ $u_2(A)$ $u_2(C)$

T_3 : $xl_3(B)$ $w_3(B)$ $r_3(B)$ $sl_3(A)$ $r_3(A)$ $u_3(B)$ $u_3(A)$

- (zufälliger) Schedule:

$sl_2(A)$ $r_2(A)$ $xl_3(B)$ $w_3(B)$ $r_3(B)$ $xl_2(A)$ $w_2(A)$ $sl_3(A)$ $r_3(A)$ $u_3(B)$
 $u_3(A)$ $sl_2(C)$ $r_2(C)$ $sl_1(C)$ $r_1(C)$ $xl_1(C)$ $w_1(C)$ $u_1(C)$ $r_2(A)$ $u_2(A)$
 $u_2(C)$

- Aufgabe:**

Führe den Schedule im Scheduler aus!

$sl_1(A)$ $sl_2(A)$ $xl_1(A)$ ⚡
 $xl_1(A)$ $sl_2(A)$ ⚡

8

$sl_2(A)$ $r_2(A)$ $xl_3(B)$ $w_3(B)$ $r_3(B)$ $xl_2(A)$ $w_2(A)$ $sl_3(A)$ $r_3(A)$ $u_3(B)$
 $u_3(A)$ $sl_2(C)$ $r_2(C)$ $sl_1(C)$ $r_1(C)$ $xl_1(C)$ $w_1(C)$ $u_1(C)$ $r_2(A)$ $u_2(A)$ $u_2(C)$

T_1	T_2	T_3

$sl_2(A)$ $r_2(A)$ $xl_3(B)$ $w_3(B)$ $r_3(B)$ $xl_2(A)$ $w_2(A)$ $sl_3(A)$ $r_3(A)$ $u_3(B)$
 $u_3(A)$ $sl_2(C)$ $r_2(C)$ $sl_1(C)$ $r_1(C)$ $xl_1(C)$ $w_1(C)$ $u_1(C)$ $r_2(A)$ $u_2(A)$ $u_2(C)$

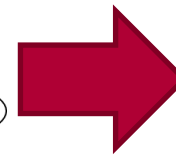
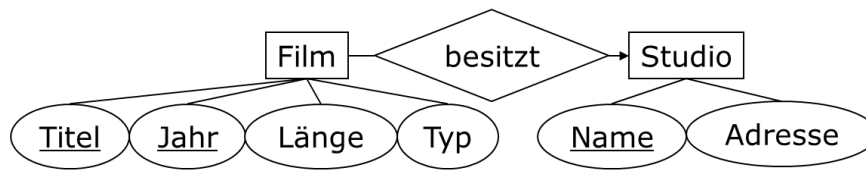
T_1	T_2	T_3
	$sl_2(A)$ $r_2(A)$	
		$xl_3(B)$ $w_3(B)$ $r_3(B)$
	$xl_2(A)$ $w_2(A)$	
		$sl_3(A)$ ⚡
	$sl_2(C)$ $r_2(C)$	
$sl_1(C)$ $r_1(C)$	sl & sl	
$xl_1(C)$ ⚡	sl > xl	
	$r_2(A)$ $u_2(A)$ $u_2(C)$	
		$sl_3(A)$ $r_3(A)$ $u_3(B)$ $u_3(A)$
$xl_1(C)$ $w_1(C)$ $u_1(C)$		

Wiederholung: Themen der Übungen

10



1. ER-Diagramm → Relationales Schema



Film				besitzt		
Titel	Jahr	Länge	Typ	Titel	Jahr	studioName
Basic Instinct	1992	127	Farbe	Basic Instinct	1992	Fox
Total Recall	1990	113	Farbe	Total Recall	1990	Disney
Dead Man	1995	121	s/w	Dead Man	1995	Paramount

Studio	
Name	Adresse
Basic Instinct	Union-Street 16
Total Recall	Carrowstr. 1
Dead Man	Myway 48

2. Funktionale Abhängigkeiten und Schlüssel

Film			
Titel	Jahr	Länge	Typ
Basic Instinct	1992	127	Farbe
Total Recall	1990	113	Farbe
Dead Man	1995	121	s/w

Studio	
Name	Adresse
Basic Instinct	Union-Street 16
Total Recall	Carrowstr. 1
Dead Man	Myway 48

besitzt		
Titel	Jahr	studioName
Basic Instinct	1992	Fox
Total Recall	1990	Disney
Dead Man	1995	Paramount

3. BCNF und Dekomposition

Film					
Titel	Jahr	Länge	Typ	StudioName	SchauspName
Total Recall	1990	113	Farbe	Fox	Sharon Stone
Basic Instinct	1992	127	Farbe	Disney	Sharon Stone
Total Recall	1990	113	Farbe	Fox	Arnold
Dead Man	1995	121	s/w	Paramount	Johnny Depp

Film1				
Titel	Jahr	Länge	Typ	StudioName
Total Recall	1990	113	Farbe	Fox
Basic Instinct	1992	127	Farbe	Disney
Dead Man	1995	121	s/w	Paramount

Film2		
Titel	Jahr	SchauspName
Total Recall	1990	Sharon Stone
Basic Instinct	1992	Sharon Stone
Total Recall	1990	Arnold
Dead Man	1995	Johnny Depp

1. Relationale Algebra

Unäre Operatoren

Operator	Beschreibung
π (pi)	Projektion
σ (sigma)	Selektion
δ (delta)	Duplikateliminierung
ρ (rho)	Umbenennung
τ (tau)	Sortierung
γ (gamma)	Gruppierung

Binäre Operatoren

Operator	Beschreibung
\cap	Schnittmenge
\cup	Vereinigung
$-$	Differenz (auch „\“)
\times	Kreuzprodukt
\bowtie	Natürlicher Join
\bowtie_{θ}	Theta-Join
$ \bowtie$	Left outer Join
$\bowtie $	Right outer Join
$ \bowtie $	Full outer Join
\ltimes	Semijoin

$$\pi_{\text{model}}(\sigma_{\text{speed} \geq 1000}(\text{PC}))$$

$$\pi_{\text{maker}}(\sigma_{\text{hd} \geq 10}(\text{Product} \bowtie \text{Laptop}))$$

$$\delta(\pi_{\text{maker}}(\text{Product} \bowtie \text{Laptop})) - \delta(\pi_{\text{maker}}(\text{Product} \bowtie \text{PC}))$$

$$\pi_{\text{hd}}(\sigma_{\text{Anzahl} > 2}(\gamma_{\text{hd}, \text{count}(\text{model}) \rightarrow \text{Anzahl}}(\text{PC})))$$

1. SQL

SELECT <Attributliste>
FROM <Relationenliste>
WHERE <Bedingungen>
GROUP BY <Gruppierungsattribute>
HAVING <Bedingungen auf Gruppierungsattribute>
ORDER BY <Attributliste>;

Weitere Schlüsselwörter:

DISTINCT, AS, JOIN

AND, OR

MIN, MAX, AVG, SUM, COUNT

NOT, IN, LIKE, ANY, ALL, EXISTS

UNION, EXCEPT, INTERSECT

...

$\pi_{\text{model,price}}(\sigma_{\text{maker}='Apple'}(\text{Product} \bowtie (\pi_{\text{model,price}}(\text{PC}) \cup \pi_{\text{model,price}}(\text{Laptop}) \cup \pi_{\text{model,price}}(\text{Printer}))))$

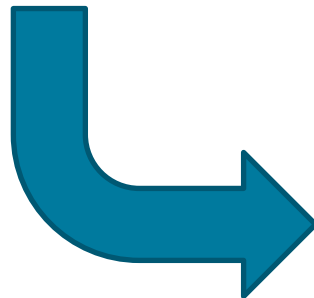
(**SELECT** Product.model, price
FROM Product **JOIN** PC **ON** Product.model = PC.model
WHERE maker = 'Apple')

UNION

(**SELECT** Product.model, price
FROM Product **JOIN** Laptop **ON** Product.model = Laptop.model
WHERE maker = 'Apple')

UNION

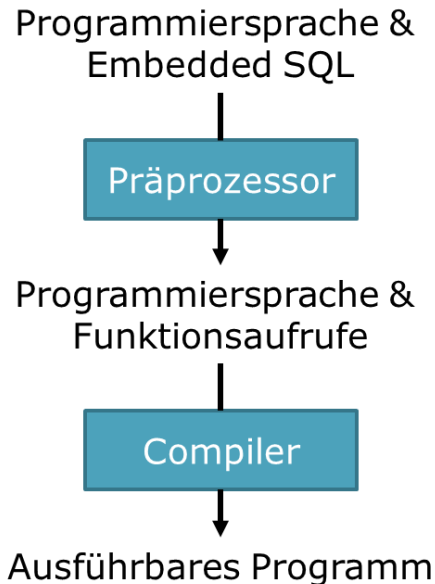
(**SELECT** Product.model, price
FROM Product **JOIN** Printer **ON** Product.model = Printer.model
WHERE maker = 'Apple');



1. Impedence Mismatch

- Generisches Modell (Programmiersprachen)
 - Pointer, verschachtelte Strukturen und Objekte
 - Schleifen und Verzweigungen
 - i.d.R. Imperativ
- Relationales Model (DBMS)
 - Relationen und Attribute
 - Nebenbedingungen
 - Deklarativ

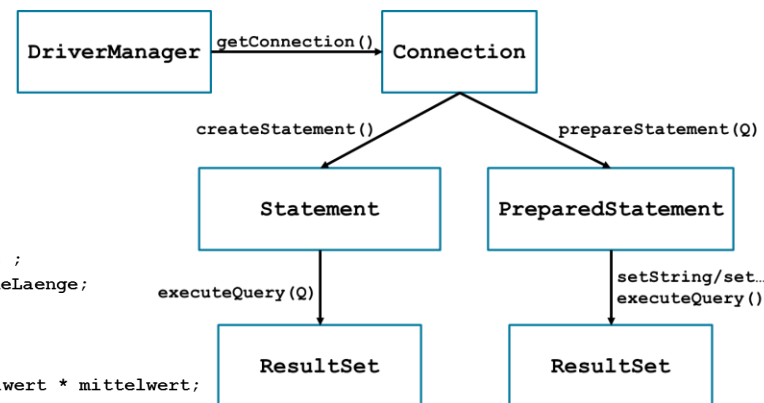
2. Embedded SQL, Stored Procedures und JDBC



```
CREATE PROCEDURE MeanVar(IN studioName CHAR[15],
                        OUT mittelwert REAL,
                        OUT varianz REAL)

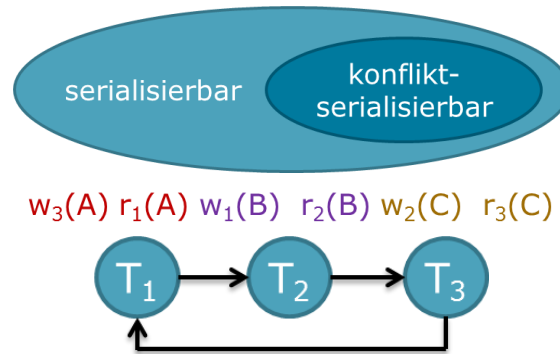
DECLARE Not_Found CONDITION FOR SQLSTATE '02000';
DECLARE FilmCursor CURSOR FOR
    SELECT Laenge FROM Filme WHERE StudioName = studioName;
DECLARE neueLaenge INTEGER;
DECLARE filmAnzahl INTEGER;

BEGIN
    SET mittelwert = 0.0;
    SET varianz = 0.0;
    SET filmAnzahl = 0;
    OPEN FilmCursor;
    FilmLoop: LOOP
        FETCH FilmCursor INTO neueLaenge;
        IF Not_Found THEN LEAVE FilmLoop END IF;
        SET filmAnzahl = filmAnzahl + 1;
        SET mittelwert = mittelwert + neueLaenge ;
        SET varianz = varianz + neueLaenge * neueLaenge;
    END LOOP;
    CLOSE FilmCursor;
    SET mittelwert = mittelwert / filmAnzahl;
    SET varianz = varianz / filmAnzahl - mittelwert * mittelwert;
END;
```



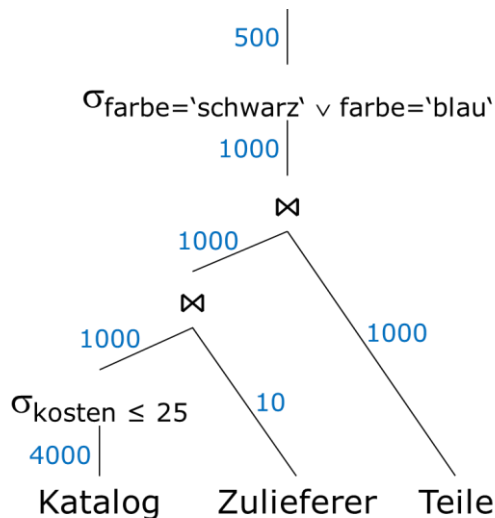
1. Transaktionen

ACID



T ₁	T ₂	T ₃
I ₁ (A) r ₁ (A)	I ₂ (B) r ₂ (B)	I ₃ (C) r ₃ (C)
I ₁ (B) ⚡	I ₂ (C) ⚡	I ₃ (D) w ₃ (D)
I ₁ (B) r ₁ (B) u ₁ (A) u ₁ (B)	I ₂ (C) r ₂ (C) u ₂ (B) u ₂ (C)	u ₃ (C) u ₃ (D)

2. Selektivität



3. XML

```

INSERT INTO CUSTOMER (cid, info)
VALUES (1000,
'<customerinfo cid="1000">
  <name>Kathy Smith</name>
  <addr country="Canada">
    <street>5 Rosewood</street>
    <city>Toronto</city>
    <prov-state>Ontario</prov-state>
    <pcode-zip>M6W 1E6</pcode-zip>
  </addr>
  <phone type="work">416-555-1358</phone>
</customerinfo>
');
    
```

Aufgabe 0: Matrikelnummern

Tragen Sie auf allen Blättern der Klausur Ihre Matrikelnummer ein. “Alle Blätter” umfasst das Deckblatt, die Aufgabenblätter und die Zusatzblätter. Tragen Sie die Matrikelnummer auch ein, falls Sie ein (Zusatz-)Blatt nicht verwenden.

Hinweis: Lösen Sie diese Aufgabe sofort.

1 Punkt

Wiederholung: Fragen zu Übungsinhalten

17

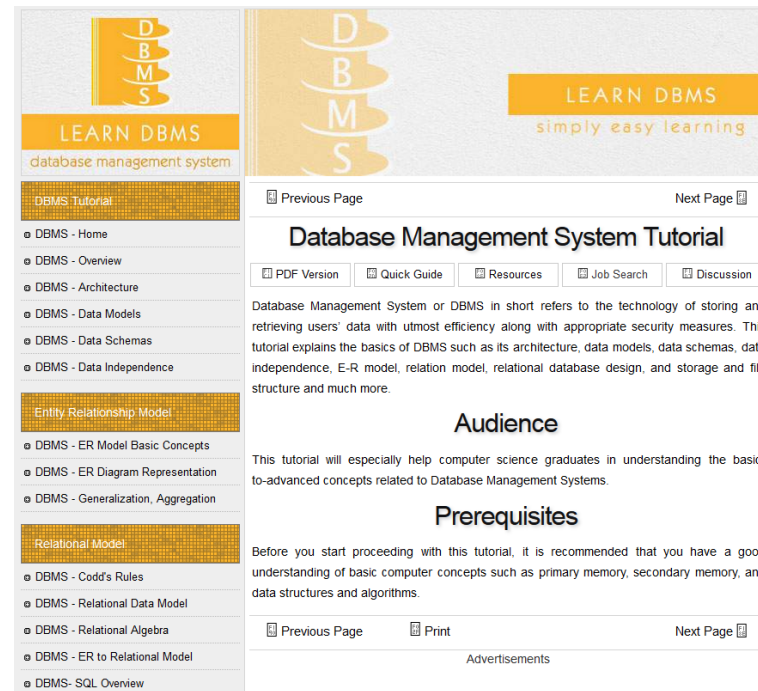


- Übungsaufgaben:
 - openHPI nutzen
 - Aufgaben ausdenken
 - Sonder- und Spezialfälle finden!
- Klausurinhalte:
 - Aufgaben der Übung
 - Inhalte der Vorlesung





<http://www.tutorialspoint.com/dbms/index.htm>



The screenshot shows the 'Database Management System Tutorial' page. It features a navigation menu on the left with categories like 'DBMS Tutorial', 'Entity Relationship Model', and 'Relational Model'. The main content area includes a title, a 'Previous Page' and 'Next Page' link, a 'PDF Version' button, and a brief introduction to DBMS. It also lists 'Audience' (computer science graduates) and 'Prerequisites' (basic computer concepts). At the bottom, there are 'Previous Page', 'Print', and 'Next Page' links, and an 'Advertisements' section.

Bachelor-Project

20



idealo



Flink

